Abstract

In this chapter, we present the interest of introducing parallelism in domain decomposition methods. In particular, we detail on an original approach: the flexible asynchronous iterations applied to the Schwarz alternating method. Application to various boundary value problems such as nonlinear convection-diffusion problem, Hamilton-Jacobi-Bellman problem and obstacle problem are presented. A coupled problem, i.e. the electrophoresis problem is also studied in details.

Keywords: Domain decomposition methods, Schwarz alternating method, parallel asynchronous iterations, boundary value problems, convection-diffusion problems, Hamilton-Jacobi-Bellman problem, obstacle problem, Navier-Stokes equations.

1 Introduction

The present contribution concerns the numerical solution of linear and nonlinear boundary value problems via flexible asynchronous Schwarz alternating methods. Various kinds of nonlinearities are considered: pertubation of a linear operator by a diagonal monotone increasing operator, nonlinear complementarity problems which occur in particular in mechanics, image processing and financial applications, Navier-Stokes equations in fluid mechanics which model flow problems and more generally coupled problems. Various boundary conditions are considered in the partial differential equations quoted above: Dirichlet, Neumann, Robin and mixed. The material presented in this chapter can be extended to the case of evolution problems solved via implicit, semi implicit or predictor corrector schemes since the solution of evolution problems leads to the solution of sequences of stationary problems.

Stationary problems considered here are discretized via discretization technics such as classical finite difference methods, finite elements, finite volume and variational fi-

nite difference methods. We note that under appropriate assumptions verified by the continuous problem, the above discretization technics lead to the solution of algebraic systems of equations involving M-functions in the nonlinear case and M-matrices in the linear case, respectively. These good properties permit one to insure the convergence of discrete asynchronous flexible Schwarz alternating methods. Finally, we note that the parallel asynchronous methods presented in this chapter are well suited to massive parallelism in high performance computing, grid computing and peer to peer computing.

Section 2 deals with continuous Schwarz alternating methods. Section 3 concerns the presentation of general discrete Schwarz alternating methods and more particularly, linear and nonlinear convection diffusion problems. In Section 4, we study complementarity problems: Hamilton Jacobi Bellman problems and obstacle problems. Section 5 is devoted to the analysis of a particularly interesting coupled problem: i.e. continuous electrophoresis problem. Finally, parallel implementation and significant computational results are presented and analyzed in Section 6.

2 The continuous Schwarz alternating method

Domain decomposition methods, such as the Schwarz alternating method introduced by P.L. LIONS [1, 2, 3] and M. DRYJA [4, 5, 6], are well suited to the parallel solution of boundary values problems (see [7]). For more details on the Schwarz alternating method see also [8, 9, 10, 11].

Let us first present the Schwarz alternating method in a very simple one-dimensional context. For this purpose, consider the Poisson equation with homogeneous Dirichlet boundary condition and defined in the domain $\Omega = [0, 1] \subset \mathbb{R}$

$$\begin{cases} -\frac{d^2u}{dx^2} = 0 \text{ sur } [0, 1], \\ u(0) = u(1) = 0. \end{cases} \tag{1}$$

Consider first the sequential context and assume that Ω is splitted into two overlapping subdomains Ω_1 and Ω_2, where $\Omega_1 = [0, \gamma_1^2], 0 < \gamma_1^2 < 1$ and $\Omega_2 = [\gamma_2^1, 1], 0 < \gamma_2^1 < \gamma_1^2 < 1$, where γ_1^2, γ_2^1 respectively are the right boundary of Ω_1 and the left boundary of Ω_2 respectively ; note that $\Omega = \Omega_1 \bigcup \Omega_2$ and $\Omega_1 \bigcap \Omega_2 \neq \emptyset$; and consider accordingly the decomposition of u into two subvectors u_1 and u_2. In order to solve equation (1) by the sequential Schwarz alternating method, let us define an initial guess $u^{(0)} = (u_1^{(0)}, u_2^{(0)})$; then the first component u_1 is computed on the subdomain Ω_1 using the boundary conditions $u_1(0) = 0$ and $u_1(\gamma_1^2) = \tilde{u_2}$, where $\tilde{u_2}$ is the restriction to γ_1^2 of the value of the subvector u_2 of u, computed on the other subdomain Ω_2. On the other hand, the component u_2 is computed symmetricaly on the subdomain Ω_2 using the boundary conditions $u_2(\gamma_2^1) = \tilde{u_1}$ and $u_2(1) = 0$, where $\tilde{u_1}$ is the restriction to γ_2^1 of the value of the subvector u_1. Then this iterative process is repeated alternatively. For choosing the values of $\tilde{u_1}$ and $\tilde{u_2}$ various strategies can be considered.

In the sequel let us denote by r the label of the Schwarz iteration. The case where

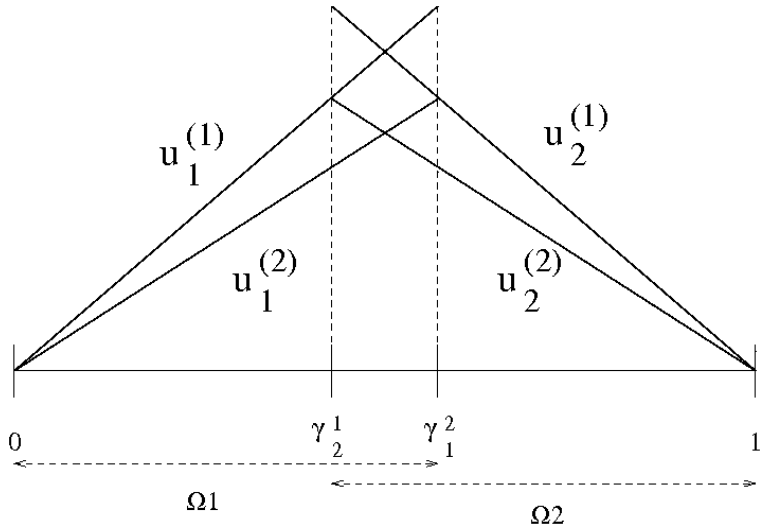

Figure 1: Additive Schwarz alternating method.

at each step $(r+1)$, we have $\tilde{u_1} = u_1^{(r)}(\gamma_1^2)$ and $\tilde{u_2} = u_2^{(r)}(\gamma_2^1)$ corresponds to a Jacobi like method, similar to an additive Schwarz alternating method (see Figure 1). On the other hand for the computation of $u^{(r+1)}$, we can also consider that $\tilde{u_1} = u_1^{(r+1)}(\gamma_1^2)$ and $\tilde{u_2} = u_2^{(r)}(\gamma_2^1)$, algorithm corresponding to a Gauss-Seidel like method, similar to a multiplicative Schwarz alternating method (see Figure 2).

More generally, in order to parallelize the computation, the domain Ω of a boundary value problem is splitted into rectangular subdomains in the 2D case or into parallelepiped subdomains in the 3D case. In order to compute a numerical solution of a boundary value problem, sequences of smaller subproblems will be solved using several processors of a parallel computer ; practically more accuracy is obtained.

Consider a boundary value problem $\mathcal{A}.u = f$ defined on the bounded domain Ω with boundary condition $\mathcal{B}.u = g$ on $\partial\Omega$, the boundary of Ω. For the sake of simplicity, we consider $\Omega \subset \mathbb{R}^2$; furthermore we consider also a decomposition in two overlapping subdomains denoted by Ω_1 and Ω_2 respectively (see Figure 3).

The parallel asynchronous solution with two processors of the above boundary value problem using the continuous Schwarz alternating method consists in solving simultaneously at each iteration :

$$\begin{cases} \mathcal{A}_1.u_1^{(r+1)} = f_1 \text{ on } \Omega_1 \\ \mathcal{B}_1.u_1^{(r+1)} = g_1 \text{ on } \partial\Omega \cap \Omega_1 \\ \quad u_1^{(r+1)} = \tilde{u}_2^{(r)} \text{ on } \gamma_1^2 = \partial\Omega_1 \cap \Omega_2 \end{cases} \quad \text{and} \quad \begin{cases} \mathcal{A}_2.u_2^{(r+1)} = f_2 \text{ on } \Omega_2 \\ \mathcal{B}_2.u_2^{(r+1)} = g_2 \text{ on } \partial\Omega \cap \Omega_2 \\ \quad u_2^{(r+1)} = \tilde{u}_1^{(r)} \text{ on } \gamma_2^1 = \partial\Omega_2 \cap \Omega_1 \end{cases}$$

where $\mathcal{A}_i$ and $\mathcal{B}_i, i = 1, 2$ respectively, denote the restriction of $\mathcal{A}$ and $\mathcal{B}$, to the subdomain Ω_i, and $\tilde{u}_1^{(r)}$ and $\tilde{u}_2^{(r)}$ denote the available values of the components of the iterate vector (u_1, u_2) at the current iteration ; more precisely,$\tilde{u}_1^{(r)}$ and $\tilde{u}_2^{(r)}$ are the restriction

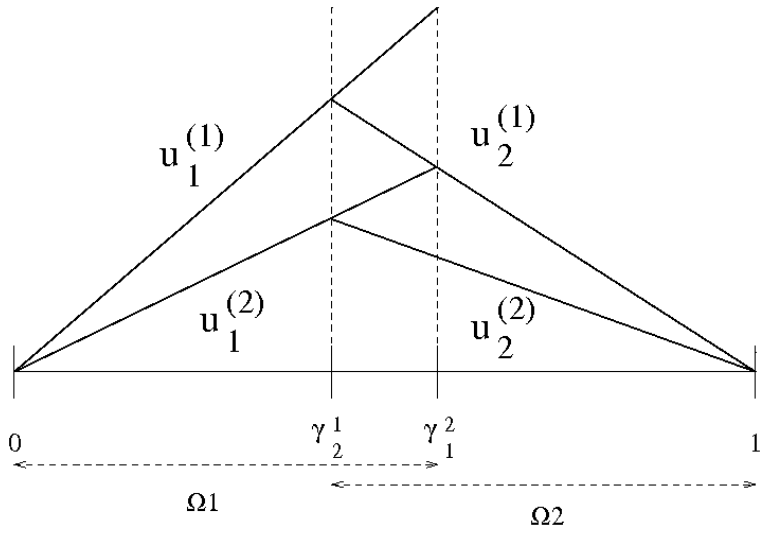

Figure 2: Multiplicative Schwarz alternating method.

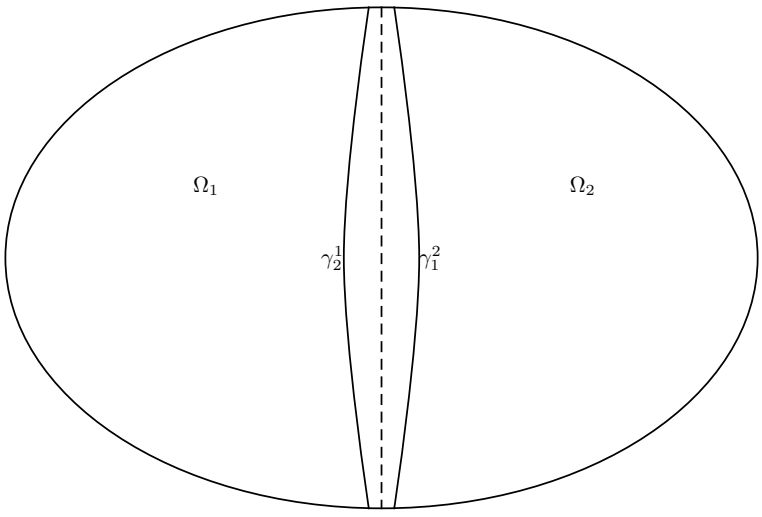

Figure 3: Decomposition of the domain Ω into 2 subdomains

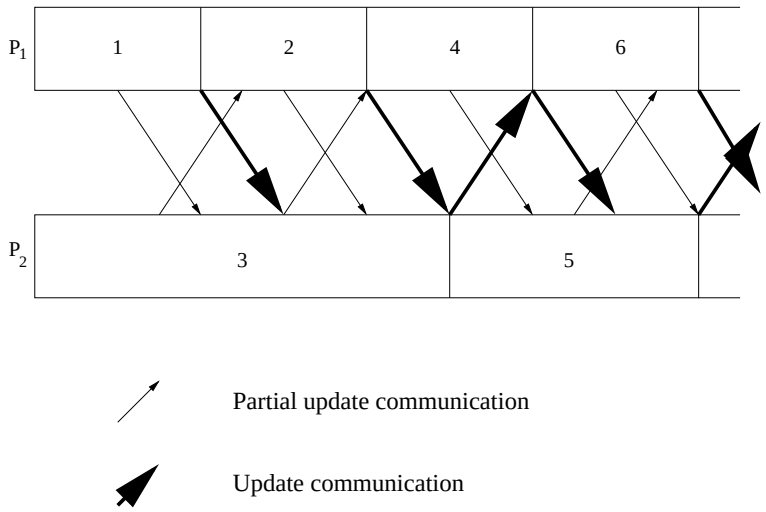

Figure 4: An example of asynchronous algorithm with flexible communications.

of the the computed components (u_1, u_2) on the boundaries γ_1^2 and γ_2^1. Flexible parallel asynchronous Schwarz alternating method (see [12, 13, 14]) are a general class of parallel iterative methods whereby computations can be made using the current values of each component of the iterate vector. The main feature of this class of parallel iterative methods is to allow flexible data exchange between the processors. In this iterative process $\tilde{u}_i^{(r)}$ are not necessarily associated to components that are labelled by an iteration number as data exchanges may occur at any time. Then, in this class of method, partial updates can be used at any time in the computation ; thus value of the components of the iterate vector which is used in an updating phase may come from updates which are still in progress.

Figure 4 displays the typical behavior of parallel asynchronous iterations with flexible communication in the simple case where two processors, denoted by P_1 and P_2, respectively, exchange data. In Figure 4, boxes and arrows, respectively, represent updating phases and communications, respectively. Thus, flexible data exchanges between processors are allowed; as a consequence, the coupling between communication and computation is improved.

This general method extends classical asynchronous Schwarz alternating method (see [15] to [19]), whereby computations are performed in parallel by several processors without any order nor synchronization by using values of the block component of the iterate vector produced at the end of each updating phase. This last kind of method generalize also the classical parallel synchronous scheme such as parallel Jacobi method, parallel Gauss - Seidel method ; note also that classical sequential Jacobi and Gauss - Seidel method can be infered from the synchronous scheme (see [20] to [31]).

3 General discrete Schwarz alternating method

3.1 A linear model convection-diffusion problem

Consider the following linear convection-diffusion problem

$$\begin{cases} -\nu\Delta u + a\frac{\partial u}{\partial x} + b\frac{\partial u}{\partial y} + cu = f, \text{ everywhere in } \Omega, \\ u = 0, \text{ on } \partial\Omega, \end{cases} \tag{2}$$

where $c \geq 0$, $\nu > 0$, Ω is a bounded domain, f is a given function of $\mathcal{L}^2(\Omega)$ and $\partial\Omega$ denotes the boundary of Ω. For the sake of simplicity, we assume that the discretization grid of the domain Ω is uniform. In the sequel h will denote the discretization step-size. We assume that the columns of the discretization grid are numbered naturally. The discretization of the operators which occur in problem (2) is made according to the following rules: the Laplacian is discretized via the classical five points scheme and the first derivatives are discretized as follows according to the sign of a and b

$$\frac{\partial u}{\partial x} = \begin{cases} \frac{u(x,y)-u(x-h,y)}{h} + \mathcal{O}(h), \text{ if } a > 0, \\ \frac{u(x+h,y)-u(x,y)}{h} + \mathcal{O}(h), \text{ if } a < 0. \end{cases} \tag{3}$$

Let A denote the discretization matrix of problem (2) ; let us also denote by F the corresponding right hand side of the discretized system. If c is strictly positive, then regardless the sign of a and b, it follows from (3) that the off-diagonal entries of matrix A are non-positive and the diagonal entries of A are positive. Moreover, the matrix A is strictly diagonally dominant; thus, A is a nonsingular M-matrix, i.e. $a_{ij} \leq 0$ for all $i \neq j$ and $A^{-1} \geq 0$ (see[32], [33]).

If c=0, then we can show that the matrix A is diagonally dominant. Moreover, by using the characterization of irreducible matrices (see [33]) we can verify that the matrix A is irreducibly diagonally dominant. Thus, in this case A is also an M-matrix(see [33]).

Consider now a red-black ordering of the columns of the grid and let $\hat{A}$ be the corresponding discretization matrix derived from A by a permutation which preserves the sign of the entries. We consider the former discretization scheme; if c is strictly positive, then, the matrix $\hat{A}$ is strictly diagonally dominant; if $c = 0$, then we can show analogously that the matrix $\hat{A}$ is irreducibly diagonally dominant. Thus, in both cases $\hat{A}$ is an M-matrix.

Let us now consider the parallel Schwarz alternating method. For the sake of clarity and simplicity, domain decomposition is presented in the 2D case. The effectiveness of domain decomposition methods is well known for boundary value problems. These methods are also well suited to parallel computing (see [7]). We concentrate here on parallel Schwarz alternating methods, which are based on overlapping subdomains.

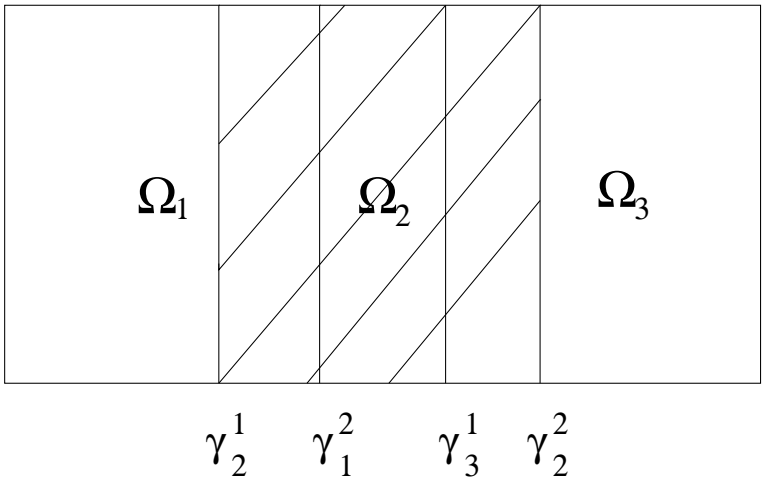

Figure 5: Example of decomposition of Ω with 3 subdomains.

Problem (2) can be decomposed into α sub-problems as follows. For $i = 1, ..., \alpha$,

$$\begin{cases} -\nu\Delta u_i + a\frac{\partial u_i}{\partial x} + b\frac{\partial u_i}{\partial y} + cu_i = f_i, \text{ everywhere in } \Omega_i, \\ u_{i_{/\Gamma_i}} = 0, \\ u_{i_{/\gamma_i^1}} = \tilde{u}_{i-1_{/\gamma_i^1}} \text{ for } 2 \leq i \leq \alpha, \\ u_{i_{/\gamma_i^2}} = \tilde{u}_{i+1_{/\gamma_i^2}} \text{ for } 1 \leq i \leq \alpha - 1, \end{cases} \tag{4}$$

where u_i and f_i, respectively, are the restriction of u and f, respectively, to Ω_i, $\Omega = \bigcup_{i=1}^{\alpha} \Omega_i$, $\Omega_i \bigcap \Omega_{i+1} \neq \emptyset, i \in \{1, \ldots, \alpha - 1\}$, $\gamma_i^1 = \partial\Omega_i \bigcap \Omega_{i-1}, i \in \{2, \ldots, \alpha\}$, $\gamma_i^2 = \partial\Omega_i \bigcap \Omega_{i+1}, i \in \{1, \ldots, \alpha - 1\}$, $\Gamma_i = \partial\Omega_i \bigcap \partial\Omega, i \in \{1, \ldots, \alpha\}$ and $\tilde{u}_j,\ j = i \pm 1$ are the restriction of the available values of the iterate vector on the overlapping boudaries $\gamma_i^k,\ k = 1, 2$ (see Figure 5).

The decomposition (4) corresponds to an overlapping subdomain decomposition, whereby u_i is computed using the restriction of u_{i-1} and u_{i+1}, respectively, on γ_i^1 and γ_i^2, respectively. In the sequential case, the scheme of computation corresponds exactly to a multiplicative Schwarz scheme. In the parallel case, the Schwarz alternating method can be combined with an asynchronous iterative scheme of computation with flexible communication in order to be as close as possible to a multiplicative scheme.

If we solve the linear simultaneous equations $AU = F$, via the Schwarz alternating method, then the augmentation process of the Schwarz alternating method transforms the M-matrix A (respectively the right hand side F) into an M-matrix $\widetilde{A}$ (respectively a vector $\widetilde{F}$)(see [34] and [12]). Thus, for the solution of the resulting linear system $\widetilde{A}\widetilde{U} = \widetilde{F}$ we are in the convergence analysis framework considered in [12] for the study of the parallel Schwarz alternating method with flexible communications ; so, for any initial guess $U^{(0)}$ satisfying $AU^{(0)} - F \geq 0$, the asynchronous Schwarz alternating method converge and according to [12] the convergence is monotone, i.e. $U^{(0)} \geq U^{(1)} \geq \ldots \geq U^{(r)} \geq \ldots U$, where $AU = F$.

Note that the initial guess $V^{(0)}$ can be also choosen such that $AV^{(0)} - F \leq 0$; in such

a case, according to the monotone convergence results we have $V^{(0)} \leq V^{(1)} \leq \ldots \leq V^{(r)} \leq \ldots U$. By combining both previous initializations for the iterative process, we can derive an effective stopping criterion of the iteration process ; indeed at each step the following inequality $V^{(r^{'})} \leq \cdots \leq U \leq \ldots \leq U^{(r)}$ holds and we can stop the iterative algorithm when $U^{(r)} - V^{(r^{'})}$ is sufficiently small.

More generally, we note that under realistic hypothesis, the finite element P_1 or P_2 discretization matrices occurring in the following analogous linear partial differential equations

$$\left\{ \begin{array}{l} -\frac{\partial}{\partial x}(p(x,y)\frac{\partial u}{\partial x}) - \frac{\partial}{\partial y}(p(x,y)\frac{\partial u}{\partial y}) + q(x,y)u = f, \text{ everywhere in } \Omega, \\ u = 0, \text{ on } \Gamma_0, \\ \frac{\partial u}{\partial n} + \sigma u = g, \text{ on } \Gamma_1 \end{array} \right. \tag{5}$$

where $\partial\Omega = \Gamma_0 \bigcup \Gamma_1$, $p, q \in C(\overline{\Omega})$ and $\sigma \in C(\Gamma_1)$, with

$$0 < p_0 \leq p(x,y) \leq p_1, \forall (x,y) \in \overline{\Omega},$$

$$0 \leq q(x,y) \leq q_1, \forall (x,y) \in \overline{\Omega},$$

$$0 < \sigma_0 \leq \sigma(x,y) \leq \sigma_1, \forall (x,y) \in \overline{\Gamma_1}.$$

is also an M-matrix (see [35]).

3.2 Nonlinear problems

In this subsection, let us consider various situations of nonlinear problems derived from the previous linear model problem by perturbation of the linear operator. We consider a first situation where nonlinearities are defined in the domain Ω. The general model can be given as follows

$$\left\{ \begin{array}{l} -\nu\Delta u + a\frac{\partial u}{\partial x} + b\frac{\partial u}{\partial y} + cu + \varphi(u) = f, \text{ in } \Omega, \\ B.C., \end{array} \right. \tag{6}$$

where $c \geq 0$, $f \in \mathcal{L}^2(\Omega)$, $\varphi : \mathbb{R} \rightarrow \mathbb{R}$ is a continuous, nondecreasing function and B.C. represents a classical boundary condition, i.e. Dirichlet, Neumann, Robin or mixed. The following nonlinear increasing functions occur in various nonlinear convection-diffusion problems : $\varphi(u) = e^{\alpha u}$, with $\alpha > 0$ and also $\varphi(u) = Log(\beta + \delta u)$, with $\delta > 0$ and a suitable sign for β, can be considered (see [36]). The discretization techniques lead to the solution of the following algebraic problem

$$\mathcal{A}(U) = AU + \Phi(U) - F = 0, \tag{7}$$

where A is the discretization matrix associated with the linear part of the equations, Φ is a diagonal operator derived from the discretization of the diagonal operator φ and $(F, U) \in \mathbb{R}^{dim(A)} \times \mathbb{R}^{dim(A)}$. According to the properties of the operator φ then Φ is a monotone increasing mapping. Assume that the same discretization techniques as

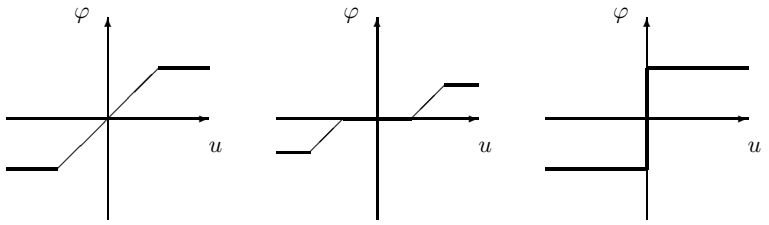

Figure 6: Different graphs for ϕ.

the one quoted in the previous subsection are used, then A is an M-matrix. Thus $\mathcal{A}$ is an M-function, according to Theorem 13.5.6 in [33] and [37], i.e. $\mathcal{A}$ is off-diagonally monotone decreasing and inverse monotone increasing ; note that, in the linear case, $\mathcal{A}(U) = AU - F = 0$ and we are in the framework of the previous subsectin since A is an M-matrix. If we solve the nonlinear simultaneous equations $\mathcal{A}(U) = 0$, using the Schwarz alternating method with flexible communications described in [12], then the augmentation process of the Schwarz alternating method transforms the M-matrix A into an M-matrix $\widetilde{A}$ and the monotone increasing mapping Φ into the monotone increasing mapping $\widetilde{\Phi}$ (see [34], [12]). Thus, the resulting nonlinear mapping $\widetilde{\mathcal{A}}$ is a surjective M-function and, for such asynchronous Schwarz alternating method with flexible communications we are in the monotone convergence framework considered in [12].

We can also consider nonlinear convection-diffusion problems where nonlinearities arise on the boundary of the domain (see [38]). This kind of problem occurs, for example, in the following boundary temperature control problem

$$\begin{cases} -\nu\Delta u + a\frac{\partial u}{\partial x} + b\frac{\partial u}{\partial y} + cu = f \text{ everywhere in } \Omega, \\ \frac{\partial u}{\partial n} + \varphi(u) = 0 \text{ on } \Gamma_d \text{ and } u = 0 \text{ on } \partial\Omega - \Gamma_d, \end{cases} \tag{8}$$

where $\Omega \subset \mathbb{R}^2$, $c \geq 0$, $\Gamma_d \subset \partial\Omega$, $f \in \mathcal{L}^2(\Omega)$ and $\varphi : \mathbb{R} \rightarrow \mathbb{R}$ is a continuous, nondecreasing, nonlinear function. Figure 6 displays some examples of graphs for function φ. In particular, the graphs (a) and (b) model saturation phenomena and the graph (c) models a multi-valued function corresponding to the boundary condition: $\frac{\partial u}{\partial n} + \varphi(u) \ni 0$.

The discretization techniques presented in the previous subsection can be also used for the interior points of domain Ω. For all points in Γ_d, the discretization of the Neumann condition leads to the solution of the following discrete equations

$$\frac{u_j - u_{j-1}}{h} + \varphi(u_j) = 0. \tag{9}$$

Thus, we have to solve the problem

$$\mathcal{A}(U) = AU + \Phi(U) - F = 0, \tag{10}$$

where A is the discretization matrix associated with the linear part of the equations, Φ is a diagonal, nondecreasing operator and $(F, U) \in \mathbb{R}^{dim(A)} \times \mathbb{R}^{dim(A)}$.

It follows from (9) that the j-th component of Φ is equal to $h\varphi(u_j)$ if j is the index of a point which belongs to Γ_d otherwise is null. If $c > 0$, then the matrix A is a strictly diagonally dominant matrix. Thus, A is an M-matrix. In the case where $c = 0$, we can verify by a similar argument that the matrix A is irreducibly diagonally dominant, regardless the sign of a and b; thus, A is an M-matrix. Since A is an M-matrix and Φ is a continuous, nondecreasing, diagonal mapping, $\mathcal{A}$ is a surjective M-function, according to Theorem 13.5.6 in [33].Thus, once again, the asynchronous Schwarz alternating method with flexible communications converges monotonically.

Note that the particular case of convection-diffusion problems with Neumann conditions defined everywhere on $\partial\Omega$ can also be considered. Then, the above analysis still holds when the condition $c > 0$ is satisfied.

Note also that the results of this subsection can also be extended to the case where a red-black ordering of the grid points is considered ; in such a case, the parallel asynchronous Schwarz alternating method with flexible communications converge also monotonically .

4 Complementarity problems

In the present section, we will study two kinds of complementarity problems, i.e. the discretized and linearized Hamilton-Jacobi-Bellman problem on the one hand and the obstacle problem on the other hand.

4.1 The discretized and linearized Hamilton-Jacobi-Bellman problem

The Hamilton-Jacobi-Bellman problem occurs in many fields such as stochastic control, management, economy, mechanics and image processing. We recall briefly the formulation of the Hamilton-Jacobi-Bellman problem with Dirichlet boundary condition

$$\begin{cases} \text{Find u solution of} \\ \sup_{v \in V}(A(v)u - f(v)) = 0, \ everywhere \ in \ \Omega, \\ u = 0, \ on \ \partial\Omega, \end{cases} \tag{11}$$

where $\Omega \subset \mathbb{R}^\eta$ is a bounded domain, $\partial\Omega$ is the boundary of Ω, V is a convex set of controls and $A(v)$ is an elliptic operator defined by

$$A(v) = -\sum_{i,j} a_{ij}(x,v)\frac{\partial^2}{\partial x_i \partial x_j} + \sum_i b_i(x,v)\frac{\partial}{\partial x_i} + c(x,v),$$

where $a_{ij}(x,v),\ b_i(x,v)$ and $c(x,v), 1 \leq i,j \leq \eta$ are bounded nonnegative functions of $\overline{\Omega} \times V \rightarrow \mathbb{R}$ and $a_{ij}(x,v), b_i(x,v)$ and $c(x,v) \in C^2(\Omega)$.

P.L. LIONS in [39, 40, 41] has proved that, under appropriate assumptions, particularly regularity of coefficients and ellipticity of mappings, the previous problem has a unique solution ; moreover problem (11) can be formulated as the following equivalent problem which is more easy to solve

$$\begin{cases} \text{Find u solution of} \\ \max\limits_{1\leq k\leq m}(A^k u - f^k) = 0, \ everywhere\ in\ \Omega \\ u = 0, \ on\ \partial\Omega, \end{cases}$$

where m is a fixed integer and for all $k \in \{1, 2, \ldots, m\}$, A^k and f^k are defined by

$$A^k(v) = -\sum_{i,j} a_{ij}^k(x,v)\frac{\partial^2}{\partial x_i \partial x_j} + \sum_i b_i^k(x,v)\frac{\partial}{\partial x_i} + c^k(x,v), \text{ and } f^k = f(v_k).$$

In the sequel we will restrict the study to the case $m = 2$. Thus, we consider the following problem

$$\begin{cases} \text{Find } u \text{ such that} \\ \max\{\mathbf{A}^1 u - f^1, \mathbf{A}^2 u - f^2\} = 0, \text{ everywhere in } \Omega, \\ u = 0, \text{ on } \partial\Omega, \end{cases} \tag{12}$$

where $\mathbf{A}_1$ and $\mathbf{A}_2$ are two elliptic operators of the second order satisfying the Maximum Principle, if previous appropriate assumptions are verified by $a_{ij}(x,v), c(x,v), 1 \leq i,j \leq \eta$ and f_1, f_2 are element of $L^2(\Omega)$.

If we consider appropriate discretization of problem (12) by finite differences, then we obtain the following discretized problem

$$\begin{cases} \text{Find } U \text{ solution of} \\ \max(A^1 U - F^1, A^2 U - F^2) = 0, \end{cases} \tag{13}$$

where, F^1, $F^2 \in \mathbb{R}^n$, and A^1, A^2 are matrices of size $n \times n$ with entries a_{ij}^1, a_{ij}^2, respectively, which satisfy

$$a_{ii}^k > 0, a_{ij}^k \leq 0, i,j = 1,...,n, j \neq i, k = 1,2, \tag{14}$$

$$\sum_j a_{ij}^k \geq 0, i = 1,...,n, k = 1,2, \tag{15}$$

and n denotes the number of grid points inside the domain Ω; furtheremore assume that

$$\text{there exists at least one } i \text{ such that } \sum_j a_{ij}^1 > 0 \text{ and } \sum_j a_{ij}^2 > 0, \tag{16}$$

$$\text{the matrices } A^1 \text{ and } A^2 \text{ are irreducible.} \tag{17}$$

Note that the matrices A^1 and A^2 are diagonally dominant. Under the above assumptions A^1 and A^2 are also M-matrices (see [32]). The problem (13) can be linearized as follows:

$$\mathcal{A}(U) = C(U).U - F(U) = 0,$$

where $F(U) \in \mathbb{R}^n$ and $C(U)$ is a matrix of size $n \times n$. In fact the linearization process is defined by the Howard method as follows :
If $(A^1U - F^1)_i$ is greater than $(A^2U - F^2)_i$, then the i-th row of matrix $C(U)$ is equal to the i-th row of matrix A^1 otherwise it is equal to the i-th row of matrix A^2. The vector $F(U)$ is defined analogously.

It follows from the above assumptions that the matrix $C(U)$ is an irreducible diagonally dominant matrix; thus $C(U)$ is an M-matrix. Thus, $\mathcal{A}$ is a continuous surjective M-function (see [12]).

Then, consider now the use of Schwarz alternating method for the solution of the linearized and discretized Hamilton-Jacobi-Bellman equations. Taking into account of the previous properties, particularly the fact that the mapping $\mathcal{A}$ is a continuous surjective M-function, then the Schwarz augmentation process leads to solve the augmented system $\tilde{\mathcal{A}}(\tilde{U}) = 0$, where $\tilde{\mathcal{A}}(\tilde{U}) = \tilde{C}(\tilde{U}).\tilde{U} - \tilde{F}(\tilde{U})$. Thus, according to results of subsection 3.2 (see also [12]) , the mapping $\tilde{\mathcal{A}}$ is a continuous surjective M-function and the parallel asynchronous Schwarz alternating method with flexible communications converge monotonously.

4.2 The obstacle problem

The obstacle problem occurs in many fields such as mechanics and finance. In financial applications there exists various mathematical models. We consider first the european option derivatives which is modelled by a linear diffusion boundary value problem or more generally by a linear convection-diffusion boundary value problem defined in a normed vectorial space ; such partial differential equation, similar to (2) can be solved by flexible parallel asynchronous Schwarz alternating method. For the study of convergence of the parallel asynchronous Schwarz alternating with flexible communications, the reader is refered to subsection 3.1.

We consider now american option derivative modelled by a linear diffusion boundary value problem or more generally by a linear convection-diffusion boundary value problem defined in a closed convex set $\mathcal{K}$. This mopdel occurs also in mechanics. In order to introduce this problem we will consider first an american option derivative derived from a diffusion boundary value problem. Indeed, let us define the symmetric bilinear form

$$a(u, v) = \int_\Omega (\nabla u \nabla v + duv) dx,$$

where $\Omega \subset \mathbb{R}^\eta$ is a bounded domain and d is a real positive bounded function ; let us also define the linear form

$$L(v) = \int_\Omega fv dx,$$

where f is a given function of $\mathcal{L}^2(\Omega)$. Consider now the following convex optimization problem

$$\begin{cases} \text{Find } u \in \mathcal{K} \text{ such that} \\ J(u) \leq J(v), \ \forall v \in \mathcal{K}, \end{cases}$$

where $J(v)$ is defined by

$$J(v) = \frac{1}{2}a(v,v) - L(v),$$

$\mathcal{K}$ is a closed convex set defined by

$$\mathcal{K} = \{v \in H_0^1(\Omega), v \geq \psi \text{ everywhere in } \Omega\},$$

and $H_0^1(\Omega)$ denotes classically the set of functions belonging to $\mathcal{L}^2(\Omega)$ whith gradients belonging to the same space and $\psi \in \mathcal{L}^2(\Omega)$. Classicaly, a characterization of the solution of the previous convex optimization problem (see [42], [43]) is given by solving the following problem

$$\begin{cases} (-\Delta u + du - f)(u - \psi) = 0 \text{ in } \Omega, \\ -\Delta u + du \leq f \text{ and } u \leq \psi \text{ in } \Omega, \\ u = 0, \text{ on } \partial\Omega. \end{cases}$$

An other characterization of the solution of such problem can be given by:

$$\begin{cases} \text{Sup } (-\Delta u + du - f, u - \psi) = 0, \text{ everywhere in } \Omega, \\ u = 0, \text{ on } \partial\Omega. \end{cases}$$

More generally, assume that d is a real positive bounded function and consider now the american financial derivative option associated with a convection-diffusion boundary value problem

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

A characterization of the solution of the previous problem can also be given by:

$$\begin{cases} \text{Sup } (-\Delta u + b\frac{\partial u}{\partial x} + c\frac{\partial u}{\partial y} + du - f, u - \psi) = 0, \text{ everywhere in } \Omega, \\ u = 0, \text{ on } \partial\Omega. \end{cases} \tag{18}$$

With appropriate discretization of the obstacle problem (18) by finite difference methods, we obtain the following discretized complementarity problem

$$\begin{cases} \text{Find } U \in \mathbb{R}^n \text{ solution of} \\ \text{Max } \Big(AU - F, U - \Psi\Big) = 0. \end{cases} \tag{19}$$

where A the discretization matrix of the convection-diffusion operator associated with Dirichlet boundary condition, satisfies the assumption

$$A \text{ is a strictly diagonally dominant } M\text{-matrix}, \tag{20}$$

and F and Ψ are two vectors derived from f and ψ which result from the discretization process. Since d is a nonnegative function, in the case of the convection-diffusion

operator, the assumption (20) is well satisfied if the first derivative is discretized via appropriate backward or forward decentered schemes according to the sign of the convection coefficients b and c (see (3)) and if the second derivative is discretized by the classical five points scheme ; classically, in the case of the diffusion operator assumption (20) is also satisfied. The complementarity problem (19) is a particular form of the following discrete Hamilton-Jacobi-Bellman problem

$$\max(A^1U - F^1, A^2U - F^2) = 0,$$

with $A^2 = I$ (identity matrix) and $F^2 = \Psi$.

The linearization of the previous discretized problem (19) is accomplished by the Howard-Mosco-Scarpini method, similar to the one considered in subsection 4.1 for the linearization of the Hamilton-Jacobi-Bellman problem. The linearized system is defined by

$$\mathcal{A}(U) = C(U).U - G(U) = 0, \tag{21}$$

where $C(U)$ and $G(U)$ are defined as follows
- if the first argument $AU - F$ is dominant in (19), then the i-th line of the matrix C(U) is equal to the i-th line of the matrix A and the i-th component of the vector G is equal to F_i, i-th component of the vector F,
- otherwise if the second argument $U - \Psi$ is dominant in (19), then the i-th line of the matrix C(U) is zero, except for the diagonal entry c_{ii} which is equal to one and the i-th component of the vector G which is equal to Ψ_i, i-th component of the vector Ψ.

It follows from (20) and Theorem 3.12 in [32], that $C(U)$ is an M-matrix. Then, $\mathcal{A}$ is a continuous surjective M-function. Thus, the parallel asynchronous Schwarz alternating method with flexible communications converge monotonously. The reader is refered to [44] for more details.

Note that the stationary obstacle problem associated with a second order elliptic operator satisfying the maximum principle can be classically written as an Hamilton-Jacobi-Bellman problem and the parallel asynchronous algorithms with flexible communications presented here can be used for the numerical solution of this problem. Nevertheless, it can be noted that the convergence analysis of parallel asynchronous iterative methods derived from the linearization process for the obstacle problem and for the Hamilton-Jacobi-Bellman problem does not follow from the same arguments since the discretization matrix can be reducible in the first case, whereas it is irreducible in the general latter case.

5 Continuous 3D flow electrophoresis problem

Now we present in details a practical application in order to illustrate our approach.

5.1 Principle of continuous flow electrophoresis

Continuous flow electrophoresis is a process for separating protein mixtures. This process is currently used for analysis in biology. Its resolution is determined by the migration distance at the collection plane and by the scale of the filament occuped by each protein species. Set of proteins, i.e. filaments undergo spreading due to a number of different phenomena, among which electrokinetics and electrohydrodynamics are known to be important. In the first of these phenomena, differences in migration velocity between the ionic species give rise to local variations in electrical conductivity near the protein filament. In the second phenomenum, the local change in electrical conductivity distorts the electrical field, thus including shear stress in the liquid and creating a local flow pattern. More precisely, density coupling phenomena involving thermal and solutal connection induce strong instability effect and numerical simulations can bring useful informations concerning the nature of expected effects. A physical model has been developed (see [45], [46] and [47]) in order to describe these phenomena when two or several proteins are being separated. This model consists in coupling three evolutive boundary value problems defined on a bounded domain Ω included in the three dimensional space ; taking into account the classical shape of the electrophoresis chamber, in the sequel we will consider that Ω is a parallelepiped. So the coupled equations describing the considered physical phenomena are

- the Navier-Stokes equations with mixed boundary conditions (the Dirichlet boundary conditions being preponderant), which describe the flow,
- an evolutive equation with mixed boundary conditions, more precisely the Dirichlet boundary conditions arising on three faces, which describes the transport of protein
- a potential equation which corresponds to a generalized Laplacian with Dirichlet boundary conditions, which describe the the electrical phenomena.

This process takes place in a very long parallelepipedic cell; a solution strains with low speed through this cell (see Figure 7). The solution constituted by the mixture to separate is injected in this flow by the face C of the cell as a sharp liquid filament. An electrical field is created through the cell by two electrodes located on both sides of the cell, on the faces E and F, respectively.

The proteins are transfered by the flow along the cell; furthermore due to the effect of the electrical field they migrate. The various species of protein having different electrical mobilities, they can be collected separatly on the face D.

In the sequel, the flow is assumed to be isothermal and without chemical reaction; consequently the various physical coefficients arising in the phenomenom are constant.

The physical phenomena related to the present study concerns

- the mass conservation,

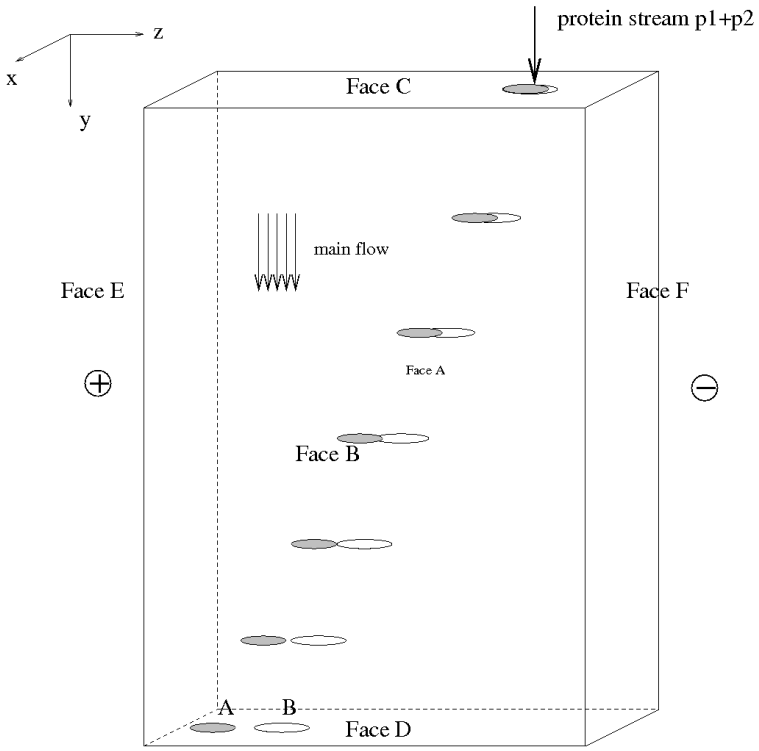

Figure 7: The principle of continuous flow electrophoresis.

- the main flow of the fluid in the three dimensional space, described by the velocity $\vec{V} = (u, v, w)$, which belongs on the one hand to the pressure p and on the other hand to the electrokinetic effect,
- the transport and the migration of the proteins and more specially the concentration c of protein, which belongs to $\vec{V}$,
- the electrokinetic effect, connected to the spatial changes of the potential in terms of the concentration of the various ionic species.

5.2 The physical problem

We present in the sequel the physical parameters which govern the electrophoresis flow, in the physical model developped by M.J. CLIFTON, V. SANCHEZ et all (see [45], [46] and [47]) ; the parameters are given at each point $M = (x, y, z)$ of the bounded domain Ω included in the three dimensional space

- the velocity field $\vec{V} = (u, v, w)$,
- the pressure p,
- the electrical field $\vec{E} = (E_x, E_y, E_z)$,

- the concentration c_m, of each protein m,
- the potential Φ,
- the temperature T,
- the kinematic viscosity of the fluid ν,
- the volumetric mass of the fluid ρ,
- the dielectric permittivity of the fluid ϵ,
- the diffusion coefficient of the protein m D_m,
- the electrical conductivity of the fluid K,
- the mean ionic conductivity of the protein m λ_m,
- the electrophoretic mobility of the protein m μ_m.

In the actual problem, we consider the flow of an incompressible viscous fluid in the domain Ω; the volumetric mass of the fluid ρ is then a constant and the following mass conservation law

$$\frac{\partial \rho}{\partial t} + \text{ div}(\rho \vec{V}) = 0$$

is reduced to

$$div(\vec{V}) = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0. \tag{22}$$

The main flow is described by the following 3D Navier-Stokes equations which take into account the external strength field

$$\frac{\partial \vec{V}}{\partial t} + \overline{\overline{\nabla \vec{V}}}.\vec{V} = \nu \Delta \vec{V} - \frac{1}{\rho}\nabla p + S \tag{23}$$

where, for $\xi = x, y, z$, $S = (S_u, S_v, S_w)$ is the source term defined by

$$S_\eta = \epsilon \text{ div}(E_\xi.\vec{E}) = \epsilon(\frac{\partial}{\partial x}E_\xi E_x + \frac{\partial}{\partial y}E_\xi E_y + \frac{\partial}{\partial z}E_\xi E_z), \text{ for } \eta = u, v, w,$$

and

$$\overline{\overline{\nabla \vec{V}}} = \begin{pmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} & \frac{\partial u}{\partial z} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} & \frac{\partial v}{\partial z} \\ \frac{\partial w}{\partial x} & \frac{\partial w}{\partial y} & \frac{\partial w}{\partial z} \end{pmatrix}.$$

The transport equation for a protein m is modelled by the following evolution convection - diffusion evolution equation

$$\frac{\partial c_m}{\partial t} + u\frac{\partial c_m}{\partial x} + v\frac{\partial c_m}{\partial y} + w\frac{\partial c_m}{\partial z} - D_m \Delta c_m \quad = \quad \varphi. \tag{24}$$

where φ is the source term.

The potential Φ is governed by a generalized Poisson equation

$$-\text{ div }(K\text{ grad }\Phi) = \Delta Q, \tag{25}$$

which can also be written as follows

$$-\frac{\partial}{\partial x}(K\frac{\partial \Phi}{\partial x}) - \frac{\partial}{\partial y}(K\frac{\partial \Phi}{\partial y}) - \frac{\partial}{\partial z}(K\frac{\partial \Phi}{\partial z}) = \Delta Q, \tag{26}$$

with $Q = Q_0 + RT\sum \mu_m c_m$, where R is the constant arising in the law of the perfect gas, and $K = K_0 + \sum \lambda_m c_m$, where $\forall m,\ \lambda_m > 0$ and $\sum_{m=1}^{n_p} \lambda_m = 1$. The equation governing the flow (23) is coupled with the above relation by

$$\vec{E} = -\vec{\text{grad}}\ \Phi \tag{27}$$

The above partial differential equations (23) to (25) must be completed by the definition of boundary values induced by physical considerations.

The fluid comes in the cell by the upper face C and comes out by the lower face D. We consider that the velocity fulfils non homogeneous Dirichlet boundary conditions on the face A and homogeneous Neumann boundary conditions on the face D. Furthermore the velocity is equal to zero on the other four faces ; so the velocity fulfils homogeneous Dirichlet boundary conditions on these last faces. In brief the boundary conditions for the velocity can be written as follows

$$\begin{cases} u_{/A} = v_{/A} = w_{/A} = u_{/B} = v_{/B} = w_{/B} = 0, \\ u_{/C} = w_{/C} = 0, v_{/C} = V_C \\ \frac{\partial u}{\partial n}_{/D} = \frac{\partial v}{\partial n}_{/D} = \frac{\partial w}{\partial n}_{/D} = 0, \\ u_{/E} = v_{/E} = w_{/E} = u_{/F} = v_{/F} = w_{/F} = 0 \end{cases} \tag{28}$$

Concerning the transport equation (24), the proteins come in the cell by the face C ; so on this upper face the concentration is known and the concentration fulfils non homogeneous Dirichlet boundary condition on the face C. Furthermore the concentration is unfixed on the other five faces of the cell; so we can consider that on these five faces the concentration fulfils homogeneous Neumann boundary conditions. Nevertheless we assume that the proteins do not reach the electrodes on the faces E and F. Consequently, we consider that the concentration fulfils homogeneous Dirichlet boundary conditions on the faces E and F. In brief the boundary conditions for the concentration can be summarized as follows

$$\begin{cases} \frac{\partial c_m}{\partial n}_{/A} = \frac{\partial c_m}{\partial n}_{/B} = \frac{\partial c_m}{\partial n}_{/D} = 0, \\ c_{m/C} = c_{Jet} \\ c_{m/E} = c_{m/F} = 0 \end{cases} \tag{29}$$

The potential is known and constant at every points of the electrodes, i.e. on the two lateral faces E and F; thus the potential fulfils non homogeneous Dirichlet boundary conditions on the faces E and F. Furthermore the two other vertical faces A and B are electricaly isolated ; consequently the potential fulfils non homogeneous Dirichlet boundary conditions on these two faces obtained by a linear interpolation between the values of the potential defined on the electrodes. On the horizontal faces C and D, the potential fulfils non homogeneous Dirichlet boundary conditions obtained by the solution of the potential equation restricted to each upper and lower face ; these boundary conditions are preliminary computed. Taking into account that the concentration on the face C is constant, the potential on this face is computed once only and the boundary condition on the face C does not change. On the other hand the concentration on the face D changes during the time; then, in order to obtain the missing boundary condition on the face D, at each time step the potential must be computed on this face. In brief, the boundary conditions for the potential can be summarized as follows

$$\begin{cases} \Phi_{/A} = \frac{(|Z|-z)\Phi_E + z\Phi_F}{|Z|}, \Phi_{/B} = \frac{(|Z|-z)\Phi_E + z\Phi_F}{|Z|} \\ \Phi_{/C} = \Phi_C, \Phi_{/D} = \Phi_D(t), \Phi_{/E} = \Phi_E, \Phi_{/F} = \Phi_F \end{cases} \tag{30}$$

where $|Z|$ denotes the width of the cell.

In order to solve numerically the coupled boundary value equations, we will consider in the sequel various well adapted discretization technics and we will also establish usefull properties verified by the discrete operators ; these assumptions allows to analyze the behaviour of the parallel asynchronous Schwarz alternating method used in order to solve the large algebraic systems derived from the discretization.

5.3 Discretization of the Navier-Stokes equations

The Navier-Stokes equations are solved by the PISO algorithm (Pressure Implicit with Split of Operators), a predictor-corrector method, introduced by R.I. Issa (see [48]) coupled with a discretization by the standart finite volume method (see [49]). In the case of incompressible flow, the PISO method is an implicit algorithm corresponding to a time marching predictor-corrector method based upon the splitting of the solution of velocity equations and pressure equations. The principle of the PISO method allows to deal with the coupling of the variables $(\vec{V}, p)$ by dividing each time step into three sub-time steps

The predictor step : starting from $\vec{V}^n = (U^n, V^n, W^n)$ and $P^{(n)}$, this step allows to compute by an implicit way $\vec{V}^{n+\frac{1}{3}} = (U^{n+\frac{1}{3}}, V^{n+\frac{1}{3}}, W^{n+\frac{1}{3}})$, satisfying the momentum equation when the pressure is fixed at its value of the previous time step $P^{(n)}$. Note that, in this first step, the mass conservation equation (22) is not verified. Then the velocity components are solution of the three systems like the following one written in order to obtain $U^{n+\frac{1}{3}}$

$$D_u.U^{n+\frac{1}{3}} = H_u^{'}(U^{n+\frac{1}{3}}) - \frac{1}{\rho}\Delta_x P^n + S_u + \frac{1}{\delta t}U^n, \tag{31}$$

where H_u is the discretization matrix of the convection-diffusion terms given by $H_u = A_0^u + H_u^{'}$ and A_0^u is the diagonal part of H_u, S_u is the term source equal to ϵ.div $(E_x.E)$, $D_u = (\frac{I}{\delta t} - A_0^u)$ is a diagonal matrix (D_v and D_w being defined accordingly) and Δ_ξ is the difference operator approximation of the first space derivative $\frac{\partial}{\partial \xi}$ ($\xi = x$ or $\xi = y$ or $\xi = z$). Note that the predictor step consists in the solution of three uncoupled algebraic linear sytems each system allowing to obtain a component of the velocity ; so these systems can be solved independantly by a parallel way.

First corrector step : this first explicit step compute the field velocity $\vec{V}^{n+\frac{2}{3}}$ associated with the pressure $P^{(n+\frac{1}{2})}$, verifying the discrete approximate Navier-Stokes equations and the approximate mass conservation equation; this step consists, for example, in finding $U^{n+\frac{2}{3}}$ such that

$$\begin{cases} D_u U^{n+\frac{2}{3}} = H_u^{'}(U^{n+\frac{1}{3}}) - \frac{1}{\rho}\Delta_x P^{(n+\frac{1}{2})} + S_u + \frac{1}{\delta t}U^n \\ \text{div}\vec{V}^{n+\frac{2}{3}} = 0 \end{cases}$$

Note that the approximation consists in maintaining unchanged the term $H_u^{'}(U^{n+\frac{1}{3}})$. By subtracting the previous equation to the equation (31), we obtain

$$D_u(U^{n+\frac{2}{3}} - U^{n+\frac{1}{3}}) = -\frac{1}{\rho}\Delta_x(P^{(n+\frac{1}{2})} - P^{(n)});$$

then

$$U^{n+\frac{2}{3}} = U^{n+\frac{1}{3}} - \frac{1}{\rho}D_u^{-1}\Delta_x(P^{(n+\frac{1}{2})} - P^{(n)}).$$

The continuity equation $\text{div}\vec{V}^{n+\frac{2}{3}} = 0$ can be written in a discrete form as follows

$$\Delta_x(U^{n+\frac{2}{3}}) + \Delta_y(V^{n+\frac{2}{3}}) + \Delta_z(W^{n+\frac{2}{3}}) = 0.$$

Finally, the first corrector step consists in computing a pressure correction $P^c = P^{(n+\frac{1}{2})} - P^{(n)}$ solution of the following linear system

$$-\frac{1}{\rho}(\Delta_x D_u^{-1}\Delta_x + \Delta_y D_v^{-1}\Delta_y + \Delta_z D_w^{-1}\Delta_z)P^c = G^{n+\frac{1}{3}} \tag{32}$$

where

$$G^{n+\frac{1}{3}} = -(\Delta_x U^{n+\frac{1}{3}} + \Delta_y V^{n+\frac{1}{3}} + \Delta_z W^{n+\frac{1}{3}}).$$

In order to obtain $P^{(n+\frac{1}{2})}$ and $U^{n+\frac{2}{3}}, V^{n+\frac{2}{3}}, W^{n+\frac{2}{3}}$, it is sufficient to compute $P^{(n+\frac{1}{2})} = P^{(n)} + P^c$ and for example $U^{n+\frac{2}{3}} = U^{n+\frac{1}{3}} - \frac{1}{\rho}D_u^{-1}\Delta_x P^c$, and analogously for $V^{n+\frac{2}{3}}$ and $W^{n+\frac{2}{3}}$.

Second corrector step : using the same procedure, a second corrector step is set up and allows to improve the approximation of the pressure and the velocity fields ; starting from the fields $\vec{V}^{n+\frac{2}{3}} = (U^{n+\frac{2}{3}}, V^{n+\frac{2}{3}}, W^{n+\frac{2}{3}})$ and $P^{(n+\frac{1}{2})}$, we compute the

fields $\vec{V}^{n+1} = (U^{n+1}, V^{n+1}, W^{n+1})$ and $P^{(n+1)}$ which are taken as the approximation at time step $(n+1)$; the continuity equation is taken into account by considering the discrete Navier-Stokes equations under a more complete form by performing evolution the term $H'_\eta(\vec{V}^{n+\frac{2}{3}}), \eta = u, v, w$. So, in order to find for example U^{n+1}, we consider the momentum equation written as follows

$$D_u.U^{n+1} = H'_u(U^{n+\frac{2}{3}}) - \frac{1}{\rho}\Delta_x P^{(n+1)} + S_u + \frac{1}{\delta t}U^n. \tag{33}$$

By subtracting (33) to (31), we obtain

$$U^{n+1} = U^{n+\frac{2}{3}} + D_u^{-1}(H'_u(U^{n+\frac{2}{3}} - U^{n+\frac{1}{3}}) - \frac{1}{\rho}\Delta_x(P^{(n+1)} - P^{(n+\frac{1}{2})})).$$

Then

$$\text{div}\vec{V}^{n+1} = \text{div}\vec{V}^{n+\frac{2}{3}} - \frac{1}{\rho}(\Delta_x D_u^{-1}\Delta_x + \Delta_y D_v^{-1}\Delta_y + \Delta_z D_w^{-1}\Delta_w)P^{cc} + G^{n+\frac{1}{3}}$$

where $P^{cc} = P^{(n+1)} - P^{(n+\frac{1}{2})}$,

$$G^{n+\frac{1}{3}} = -(\Delta_x D_u^{-1}.H'_u.U^* + \Delta_y D_v^{-1}.H'_v.V^* + \Delta_z D_w^{-1}.H'_w.W^*),$$

with $U^* = U^{n+\frac{2}{3}} - U^{n+\frac{1}{3}}$, $V^* = V^{n+\frac{2}{3}} - V^{n+\frac{1}{3}}$ and $W^* = W^{n+\frac{2}{3}} - W^{n+\frac{1}{3}}$. Taking into account that $\text{div}\vec{V}^{n+\frac{2}{3}} = \text{div}\vec{V}^{n+1} = 0$, the pressure correction P^{cc} is obtained by solving the following linear system

$$-\frac{1}{\rho}(\Delta_x D_u^{-1}\Delta_x + \Delta_y D_v^{-1}\Delta_y + \Delta_z D_w^{-1}\Delta_w)P^{cc} = G^{n+\frac{1}{3}}. \tag{34}$$

It can be noted that the matrix of the second corrector step is the same as the one obtained after the first corrector step ; in fact the equations of the first corrector step and of the second corrector step are different only by their second member.

Isaa [48] has shown that two correctors steps are sufficient to obtain a suitable accuracy compatible with the discretization scheme and round of error propagation; furthermore, the previous time marching scheme, based on the Euler's scheme, is unconditionally stable.

The finite volume method : in order to achieve the discretization of the Navier-Stokes equation the PISO method must be coupled with a spatial discretization. Among the spatial discretization technics, the finite volume method [49] is the most appropriate. The finite volume method needs classically the use of four staggered meshes denoted by $\Omega_u, \Omega_v, \Omega_w$ (where the discrete components u, v and w of the velocity are approximated) and Ω_p (where the discretization of the pressure p is achieved). It is not possible to know the pressure and the velocity in any point of the mesh ; so, in order to obtain both the value of the pressure and of the velocity at the same point, extrapolation of their values are necessary. In Figure 8 to 10

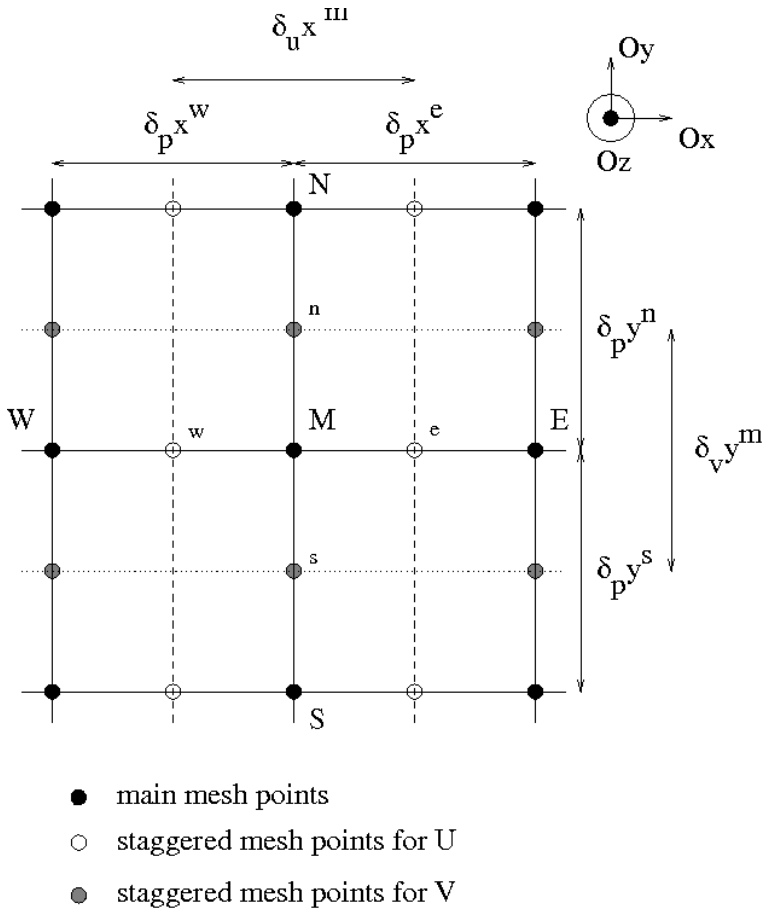

Figure 8: Section of the mesh by a parallel plane to xOy.

- the pressure is computed at the points M, E, W, N, S, H, B.
- the component U of the velocity is computed at the points **e** and **w**.
- the component V of the velocity is computed at the points **n** and **s**.
- the component W of the velocity is computed at the points **h** and **b**.
- the size of the volume control round the point M is $\delta_u x^m \times \delta_v y^m \times \delta_w z^m$.
- the size of the volume control round the point w for the computation of the component U of the velocity is $\delta_p x^w \times \delta_v y^m \times \delta_w z^m$.

Discretization of the flow equations for the predictor step. For the predictor step, the complete discretization of the Navier-Stokes equations leads to three uncoupled algebraic linear systems $A^U U = b^U$, $A^V V = b^V$, $A^W W = b^W$ obtained from the momentum equations ; note that the three matrices A^U, A^V and A^W are obtained by

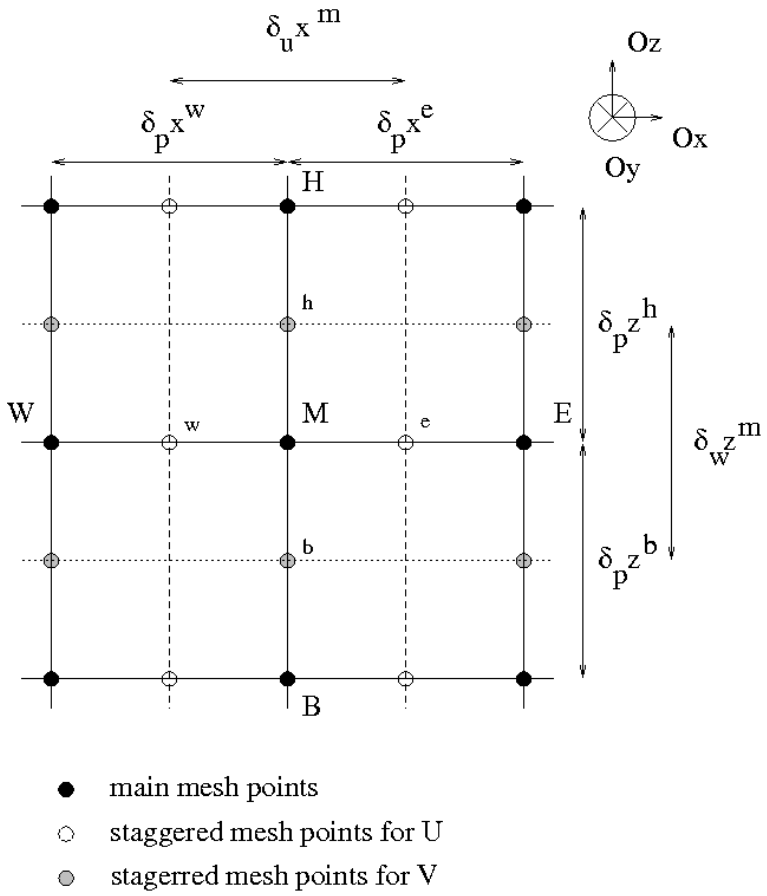

Figure 9: Section of the mesh by a parallel plane to xOz.

the same way. So, let us denote by Θ any component of the velocity (i.e. $\Theta = u$ or $\Theta = v$ or $\Theta = w$) ; then we have to integrate on a volume $Vol = \Delta x\,\Delta y\,\Delta z$ the following equation

$$\frac{\partial(\Theta)}{\partial t} + \sum_j \frac{\partial}{\partial x_j}[u_j\Theta - \frac{\partial(\nu\Theta)}{\partial x_j}] = B_\Theta \tag{35}$$

round a point of the staggered mesh where the component Θ of the velocity is defined. Then, after elementary computations [49], we finally obtain the following discretized equation

$$-a_B^\Theta\Theta_B - a_S^\Theta\Theta_S - a_W^\Theta\Theta_W + a_M^\Theta\Theta_M - a_E^\Theta\Theta_E - a_N^\Theta\Theta_N - a_H^\Theta\Theta_H = b^\Theta \tag{36}$$

where

$$b^\Theta = \int\int\int_{Vol} B_\Theta\,dx\,dy\,dz + \frac{\Delta x\,\Delta y\,\Delta z}{\delta t}\Theta_M^{(n)},$$

and where the coefficients a_*^Θ are the entries of the matrix A^U if $\Theta = U$, of the matrix A^V if $\Theta = V$ or of the matrix A^W if $\Theta = W$. The values of the entries of the previous

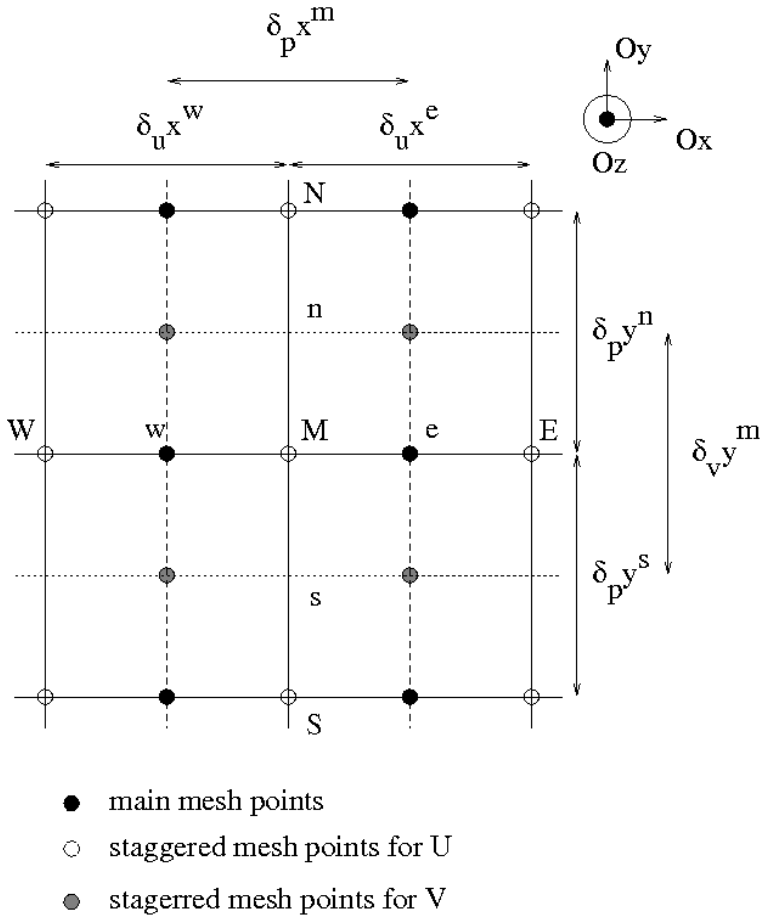

Figure 10: Section of the mesh by a parallel plane to xOy of the control volume around a point of the staggered mesh for U.

matrices are very useful for the study of the behaviour of the parallel asynchronous algorithms for the solution of the uncoupled linear systems $A^{\Theta}\Theta = b^{\Theta}$. Let us denote by $\mathcal{P}_i$ the Peclet number ([49]) on the face i ; so, for the component U of the velocity, we obtain

- lower face of the control volume $\delta_p x^w \times \delta_v y^m \times \delta_w z^m$

$$a_B^U = D_b\, \alpha(|\mathcal{P}_b|) + \max(0, F_b),$$
$$\text{where } D_b = \frac{\nu\, \delta_p x^m \delta_v y^m}{\delta_p z^b},\ F_b = W_b\, \delta_w z^m \text{ and } \mathcal{P}_b = \frac{F_b}{D_b};$$

- south face of the control volume $\delta_p x^w \times \delta_v y^m \times \delta_w z^m$:

$$a_S^U = D_s\, \alpha(|\mathcal{P}_s|) + \max(0, F_s),$$
$$\text{where } D_s = \frac{\nu\, \delta_p x^m \delta_w z^m}{\delta_p y^s},\ F_s = V_s\, \delta_v y^m \text{ and } \mathcal{P}_s = \frac{F_s}{D_s};$$

- west face of the control volume $\delta_p x^w \times \delta_v y^m \times \delta_w z^m$

$$a_W^U = D_w\, \alpha(|\mathcal{P}_w|) + \max(0, F_w),$$
$$\text{where } D_w = \tfrac{\nu\, \delta_v y^m \delta_w z^m}{\delta_u x^w},\ F_w = U_w\, \delta_p x^m \text{ and } \mathcal{P}_w = \tfrac{F_w}{D_w};$$

- east face of the control volume $\delta_p x^w \times \delta_v y^m \times \delta_w z^m$

$$a_E^U = D_e\, \alpha(|\mathcal{P}_e|) + \max(0, F_e),$$
$$\text{where } D_e = \tfrac{\nu\, \delta_v y^m \delta_w z^m}{\delta_u x^e},\ F_e = U_e\, \delta_p x^m \text{ and } \mathcal{P}_e = \tfrac{F_e}{D_e};$$

- north face of the control volume $\delta_p x^w \times \delta_v y^m \times \delta_w z^m$

$$a_N^U = D_n\, \alpha(|\mathcal{P}_n|) + \max(0, F_n),$$
$$\text{where } D_n = \tfrac{\nu\, \delta_p x^m \delta_w z^m}{\delta_p y^n},\ F_n = V_n\, \delta_v y^m \text{ and } \mathcal{P}_n = \tfrac{F_s}{D_n};$$

- upper face of the control volume $\delta_p x^w \times \delta_v y^m \times \delta_w z^m$

$$a_H^U = D_h\, \alpha(|\mathcal{P}_h|) + \max(0, F_h),$$
$$\text{where } D_h = \tfrac{\nu\, \delta_p x^m \delta_v y^m}{\delta_p z^h},\ F_h = W_h\, \delta_w z^m \text{ and } \mathcal{P}_h = \tfrac{F_h}{D_h};$$

- and lastly for the diagonal entry

$$a_M^\Theta = a_B^\Theta + a_S^\Theta + a_W^\Theta + a_E^\Theta + a_N^\Theta + a_H^\Theta + \frac{\delta_p x^m\, \delta_v y^m\, \delta_w z^m}{\delta t}.$$

In the previous relations the mappings $\alpha(|\mathcal{P}_i|)$ are defined in the table 1 in order to define many kinds of discretization scheme according to [49].

Scheme	:	$\alpha(\lvert\mathcal{P}_i\rvert)$
Centered differences:		$1 - 0.5\lvert\mathcal{P}_i\rvert$
Upwind	:	1
Hybrid	:	$Max(0; 1 - 0.5\lvert\mathcal{P}_i\rvert)$
Power Law	:	$Max(0; (1 - 0.5\lvert\mathcal{P}_i\rvert)^5)$
Exponential	:	$\lvert\mathcal{P}_i\rvert/(exp\lvert\mathcal{P}_i\rvert - 1)$

Table 1: Definition of $\alpha(|\mathcal{P}_i|)$.

For the points in the neighbourhood of the boundaries, the corresponding entries are zero :

- near the lower boundary (resp. upper), a_B^Θ (resp. a_H^Θ) is zero,
- near the south boundary (resp. north), a_S^Θ (resp. a_N^Θ) is zero,
- near the west boundary (resp. east), a_W^Θ (resp. a_E^Θ) is zero.

Other Neumann boundary conditions are discretized according to standard technics (see [49]).

The entries of the matrices A^V and A^W are defined accordingly. Then, from the values of the entries of the three matrices A^U, A^V and A^W we can easily verify that these matrices are diagonally dominant. Moreover, A^U, A^V and A^W are both M-matrices for all discretization schemes of table 1 except for the central difference scheme. For the central difference scheme, if $|\mathcal{P}_i| \leq 2$, A^U, A^V and A^W are also M-matrices ; indeed the diagonal entries of the matrices A^U, A^V and A^W are strictly positive and their off-diagonal entries are non-positive. Furtheremore the matrices A^U, A^V and A^W are obviously irreducible (see [33]) ; since there was diagonally dominant, then the M-matrix property is proved (see [50]).

Discretization of the equations for the corrector step. For the two corrector steps the PISO method leads to the following system to solve

$$-\frac{1}{\rho}(\Delta_x D_u^{-1}\Delta_x + \Delta_y D_v^{-1}\Delta_y + \Delta_z D_w^{-1}\Delta_w)P^\gamma = G.$$

equivalent to the linear system

$$A^P P^\gamma = G, \tag{37}$$

where

$$P^\gamma = \begin{cases} P^c \text{ for the first corrector step,} \\ P^{cc} \text{ for the second corrector step.} \end{cases}$$

and

$$G = \begin{cases} G^{n+\frac{1}{3}} \text{ for the first corrector step,} \\ G^{n+\frac{2}{3}} \text{ for the second corrector step.} \end{cases}$$

After discretization we finally obtain for each point M of the main mesh the following discrete equation

$$-d_B p_B^\gamma - d_S p_S^\gamma - d_W p_W^\gamma + d_M p_M^\gamma - d_E p_E^\gamma - d_N p_N^\gamma - d_H p_H^\gamma = g_M, \tag{38}$$

where :

$$\begin{cases} d_B &= 1/(a_P^W(b)\ \delta_w z^m\ \delta_p z^b), \\ d_S &= 1/(a_P^V(s)\ \delta_v y^m\ \delta_p y^s), \\ d_W &= 1/(a_P^U(w)\ \delta_u x^m\ \delta_p x^w), \\ d_E &= 1/(a_P^U(e)\ \delta_u x^m\ \delta_p x^e), \\ d_N &= 1/(a_P^V(n)\ \delta_v y^m\ \delta_p y^n), \\ d_H &= 1/(a_P^W(h)\ \delta_w z^m\ \delta_p z^h), \\ d_M &= d_B + d_S + d_W + d_E + d_N + d_H. \end{cases}$$

Finally, the matrix A^P arising in the corrector step has properties analog to the matrices arising in the predictor step. Indeed, considering the boundary conditions ([49]), the matrix A^P is diagonally dominant and, since the previous matrix is irreducible, A^P is an M-matrix.

Note also that, in the case of incompressible flow computation , the parallel asynchronous Schwarz alternating method with flexible communications can be analyzed when the Navier-Stokes equation is formulated in vorticity stream function (see [51]).

5.4 Discretization of the transport equation for proteins

The transport equation (24) is in fact a convection-diffusion equation. In the sequel we will consider a finite difference discretization for this equation. The diffusion term is discretized using the classical seven points discretization scheme. The convection terms can be discretized using either the central difference discretization scheme or the one-sided finite difference scheme. For theoretical convenience and in order to satisfy always the convergence of the iterative algorithm, we will consider in the sequel only upwind finite difference schemes ; then according to the sign of the components $u,\ v$ and w we consider only backward or forward discretization scheme for the convection terms. For example, if we consider the term $u\frac{\partial c_m}{\partial x}$ of the equation (24), the discretization is as follows

$$u(x,y,z)\frac{\partial c_m(x,y,z)}{\partial x} = \begin{cases} u(x,y,z)\frac{c_m(x,y,z)-c_m(x-\delta x,y,z)}{\delta x} + O(\delta x), \text{ if } u > 0, \\ u(x,y,z)\frac{c_m(x+\delta,y,z)-c_m(x,y,z)}{\delta x} + O(\delta x), \text{ if } u < 0. \end{cases}$$

Let us denote by A^C the associated discretization matrix ; then, the matrix A^C is an M-matrix.

If we consider a central difference discretization scheme for the convection term, then, the matrix A^C is an M-matrix, in the case where the magnitude of the components of the velocity are small enough. More precisely, in the case where $\delta x = \delta y = \delta z = h$, if

$$|u| < \frac{2D_m}{h}, |v| < \frac{2D_m}{h}, |w| < \frac{2D_m}{h},$$

then, the matrix A^C is an M-matrix. Nevertheless, from a practical point of view, the previous conditions are not interesting for the considered electrophoresis problem, since the components of the velocity are unknown.

In the sequel of the study, we will consider the migration of only one protein ; so, we will denote by c the concentration of the protein.

5.5 Discretization of the potential equation

For the potential equation, we consider a finite difference discretization. The numerical scheme is the same for every term of the diffusion equation. This scheme is obtained by making the mean of two intermediate schemes. For example let us consider firstly the discretization of $-\frac{\partial}{\partial x}\left(K\frac{\partial \phi}{\partial x}\right)$ for $y = y_j$ and $z = z_k$ fixed. In order to simplify, we shall use the following notations :

- $h_i = x_i - x_{i-1}$,
- $K_i = \mathbf{K}(x_i, y_j, z_k) = \mathbf{K}_{ijk}$,
- $K_{i\pm1} = \mathbf{K}(x_{i\pm1}, y_j, z_k) = \mathbf{K}_{i\pm1,jk}$,
- $\Phi_i = \mathbf{\Phi}(x_i, y_j, z_k) = \mathbf{\Phi}_{ijk}$,
- $\Phi_{i\pm1} = \mathbf{\Phi}(x_{i\pm1}, y_j, z_k) = \mathbf{\Phi}_{i\pm1,jk}$.

Let us first consider the two following schemes

- forward-backward scheme

$$\begin{aligned} -\tfrac{\partial}{\partial x}\left(K\tfrac{\partial \phi}{\partial x}\right)_{x_i} &= -\tfrac{1}{h_{i+1}}\left[K_{i+1}\left(\tfrac{\partial \Phi}{\partial x}\right)_{i+1} - K_i\left(\tfrac{\partial \Phi}{\partial x}\right)_i\right] \\ &= -\tfrac{1}{h_{i+1}}\left[K_{i+1}\left(\tfrac{\Phi_{i+1}-\Phi_i}{h_{i+1}}\right) - K_i\left(\tfrac{\Phi_i-\Phi_{i-1}}{h_i}\right)\right] \\ &= -\tfrac{K_i}{h_i h_{i+1}}\Phi_{i-1} + \left(\tfrac{K_i}{h_i h_{i+1}} + \tfrac{K_{i+1}}{h_{i+1}^2}\right)\Phi_i - \tfrac{K_{i+1}}{h_{i+1}^2}\Phi_{i+1} \end{aligned}$$

- backward-forward scheme

$$\begin{aligned} -\tfrac{\partial}{\partial x}\left(K\tfrac{\partial \phi}{\partial x}\right)_{x_i} &= -\tfrac{1}{h_i}\left[K_i\left(\tfrac{\partial \Phi}{\partial x}\right)_i - K_{i-1}\left(\tfrac{\partial \Phi}{\partial x}\right)_{i-1}\right] \\ &= -\tfrac{1}{h_i}\left[K_i\left(\tfrac{\Phi_{i+1}-\Phi_i}{h_{i+1}}\right) - K_{i-1}\left(\tfrac{\Phi_i-\Phi_{i-1}}{h_i}\right)\right] \\ &= -\tfrac{K_{i-1}}{h_i^2}\Phi_{i-1} + \left(\tfrac{K_{i-1}}{h_i^2} + \tfrac{K_i}{h_i h_{i+1}}\right)\Phi_i - \tfrac{K_i}{h_i h_{i+1}}\Phi_{i+1} \end{aligned}$$

Then, the final discretization scheme is obtained by making the mean of each previous scheme ; for example the second derivative with respect to x is approximated by

$$\begin{aligned} -\frac{\partial}{\partial x}\left(K\frac{\partial \phi}{\partial x}\right)_{x_i,y_j,z_k} = \frac{1}{2}\Bigg[&-\left(\frac{K_{i-1}}{h_i^2} + \frac{K_i}{h_i h_{i+1}}\right)\Phi_{i-1} \\ &+ \left(\frac{K_{i-1}}{h_i^2} + \frac{2K_i}{h_i h_{i+1}} + \frac{K_{i+1}}{h_{i+1}^2}\right)\Phi_i \\ &- \left(\frac{K_i}{h_i h_{i+1}} + \frac{K_{i+1}}{h_{i+1}^2}\right)\Phi_{i+1}\Bigg] \end{aligned} \tag{39}$$

Analogously the other second derivative with respect to y and z are approximated with the method used to obtain the scheme (39) : on one hand, for x_i and z_k fixed,

$$
\begin{aligned}
-\frac{\partial}{\partial y}\left(K\frac{\partial \phi}{\partial y}\right)_{x_i,y_j,z_k} &= \frac{1}{2}\left[-\left(\frac{K_{j-1}}{h_j^2}+\frac{K_j}{h_j h_{j+1}}\right)\Phi_{j-1}\right. \\
&+\left(\frac{K_{j-1}}{h_j^2}+\frac{2K_j}{h_j h_{j+1}}+\frac{K_{j+1}}{h_{j+1}^2}\right)\Phi_j \\
&\left.-\left(\frac{K_j}{h_j h_{j+1}}+\frac{K_{j+1}}{h_{j+1}^2}\right)\Phi_{j+1}\right]
\end{aligned} \tag{40}
$$

and on the other hand, for x_i and y_j fixed

$$
\begin{aligned}
-\frac{\partial}{\partial z}\left(K\frac{\partial \phi}{\partial z}\right)_{x_i,y_j,z_k} &= \frac{1}{2}\left[-\left(\frac{K_{k-1}}{h_k^2}+\frac{K_k}{h_k h_{k+1}}\right)\Phi_{k-1}\right. \\
&+\left(\frac{K_{k-1}}{h_k^2}+\frac{2K_k}{h_k h_{k+1}}+\frac{K_{k+1}}{h_{k+1}^2}\right)\Phi_k \\
&\left.-\left(\frac{K_k}{h_k h_{k+1}}+\frac{K_{k+1}}{h_{k+1}^2}\right)\Phi_{k+1}\right]
\end{aligned} \tag{41}
$$

Finally, the discretization matrix P of the potential equation is a heptadiagonal matrix and the approximation of the potential is obtained by solving the following linear system

$$P\Phi = S_\Phi \tag{42}$$

where $S_\Phi = \Delta Q$; the electrical conductivity K being positive, then the matrix P is positive definite. Moreover the discretization error can be computed by very simple calculation and it is very easy to show that this quantity leads to zero with the step size discretization. Furthermore, we can verify easily that the matrix P is an M-matrix.

Finally, according to the result of section 3.1, since the matrices of the seven previous discretized systems associated with the Navier-Stokes equations, the transport equation of the proteins and the potential equation are all M-matrices, we can conclude that the solution of the previous algebraic linear systems by the parallel synchronous and flexible asynchronous Schwarz alternating method converge to the solution of the considered discretized boundary value problem for any initial guess and for any ordering of the subdomains.

6 Parallel experiments

The parallel asynchronous Schwarz alternating method with flexible communications has been implemented for solving many numerical problems and in particular the various boundary value prolems quoted above ; among them we can cite the parallel simulations applied to the solution of the electrophoresis problem [52], and also the parallel solution of the obstacle problem [44], of the Hamilton-Jacobi-Bellman problem [12],

of a nonlinear diffusion problem [36] and of the Navier-Stokes problem formulated in the classical way [50] or formulated in the vorticity stream function [51]. These parallel simulations have been carried out on various multiprocessors architectures such as shared memory machines, or distributed memory machines.

Flexible asynchronous schemes of computation have also been applied to the solution of nonlinear optimization problems (see [53, 54]).

6.1 Parallel implementation of Schwarz alternating method

The implementation principle of the parallel flexible asynchronous Schwarz alternating method is similar for all applications quoted previously. The main implementation effort concerns the solution of very large scale algebraic systems. In particular linear systems must be solved since linearization techniques such as Newton method, Howard method or PISO method are used. Experimentally we have noticed (see [12], [36], [44], [50, 52]) that the general behavior of parallel flexible asynchronous Schwarz alternating method is very similar for all studied applications. In the specific case of evolution problems, where series of stationary problems are solved, the synchronization of all the computations must occur before the begining of a solution of new stationary problem.

For sake of clarity, we present in the sequel implementation and experimental results for the solution of the 3D convection-diffusion problem (2).

The Schwarz alternating method can be combined with various schemes of computation. In each case an asynchronous iterative scheme with flexible communications and a synchronous one have been implemented for the parallel numerical experiments with 3D physical model. The domain Ω, where the boundary value problem is defined, is splitted into overlapping parallelepiped subdomains (see Figure 11). We have choosen the smallest subdomain overlapping, i.e. one mesh. Thus, sequences of smaller subproblems are solved on each processor of the parallel computer in order to compute a solution of the global problem ; practically more accuracy is obtained. Several subdomains, i.e. parallepipeds, are assigned to each processor in order to implement a strategy which is close to the multiplicative strategy [8]. To obtain a faster convergence of the parallel computations, each processor handles contiguous subdomains, numbered according to red-black ordering ; such ordering is more appropriate for parallel computations (see [58]) ; according to results of section 3.1, the convergence of the parallel iterative algorithms is then obtained.

Each processor updates the components of the iterate vector associated with its subdomains and computes the residual norm corresponding to the subdomains in order to participate to the convergence detection. A block relaxation method is used in order to solve each subproblem on each subdomain ; this kind of method allows to have very flexible communications between the processors. More precisely, all points of a subdomain are updated twice by the relaxation procedure first forward, then backward, by an SSOR scanning. Note also that a direct method could be also consider for this purpose (see [44, 36]) but combersome with 3D domains since this kind of algorithm

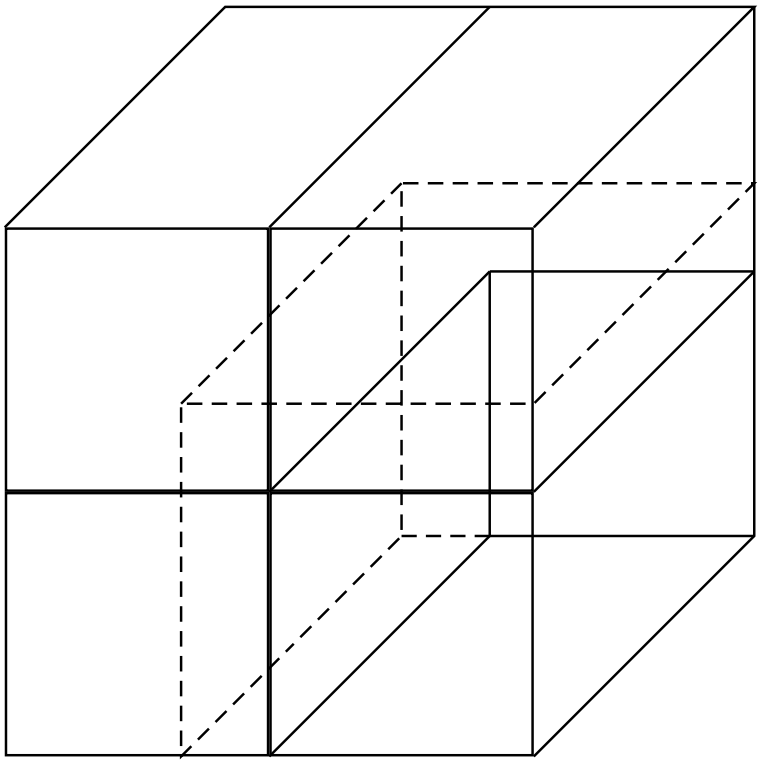

Figure 11: General 3D splitting with parallelepiped overlapping subdomains.

induce fill-in during the factorisation of the matrix.

Convergence detection of the parallel iterative process occurs when a given predicate on a global state is true ; an usual predicate corresponds to the fact that, on every subdomain, the norm of the local residual remains under a given threshold (see [55], page 580).

Various strategies of data exchange can be implemented (see [54]). We present here a strategy based on systematic communications between the processors with a given fixed exchange frequency. Note that the efficiency of parallel algorithms strongly depends on the communication frequency within the computations as communications increase the overhead. Point to point communications between two processes have been implemented using persistent communications request and MPI (Message Passing Interface) facilities in both version of Schwarz alternating methods. Message exchanges with the same argument list is repeatedly executed; it corresponds to data transmission of successive values of the components of the iterate vector associated with a subdomain frontier. That is the reason why persistent communication request has been used. A persistent communication request can be thought of as a one way channel. This approach permits one to reduce the communication overhead between the process and the communication controller.

For the sake of robustness, we have used a synchronous mode send operation since ready mode is unsafe and buffered mode may lead to overflow in the high commu-

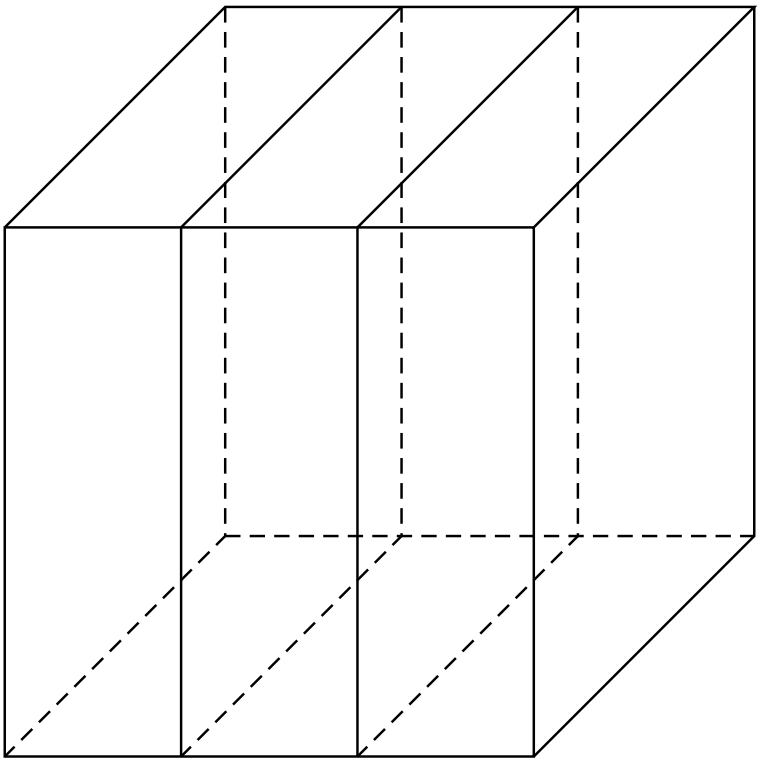

Figure 12: 3D splitting with slice overlapping subdomains.

nication frequency case. Note that the use of a synchronous mode send operation is not in contradiction with the implementation of asynchronous iterations since the implementation of communication layers and the type of implemented parallel iterative computation scheme are independent.

If global convergence is detected, then computations can be terminated and resources can be freed. All persistent communication requests are cancelled. Note that cancellation of send requests must occur before cancellation of receive requests; otherwise data exchange based on rendezvous mechanism may fail. For more details on the implementation of asynchronous iterative schemes of computation, the reader is referred to [56, 57, 58]. The principle of implementation of parallel asynchronous iterative algorithms with flexible communication can be summarized as follows

do until global convergence
  for each subdomain assigned to the processor **do**
   if local convergence is not reached **then**
    for i in 1 .. N **do**
     receive the latest frontier values
     relaxation
    end do
    send the frontier values to the neighbors

end if
end do
end do

Implementation of parallel synchronous iterative schemes of computation was based on the blocking reception of boundary values. The termination order of communications requests is totally handled with MPI facilities. It is not necessary to provide additional information about synchronous Schwarz alternating method, since its implementation and message passing issues between the processors are straightforward in this case. Reference is made to [57, 59] for implementation details concerning parallel synchronous iterative algorithms.

6.2 Numerical experiments

We present now the main computational results for a 3D convection diffusion-problem (2). For all experiments, we have considered 3,750,000 discretization points and 256 well balanced, cubic subdomains. We have tested several communication frequencies for data exchange. The tuning of the number of relaxations was made experimentally. We present here results in the case where data exchange occurs every two relaxations on each subdomain. Reception of boundary values occurs in the beginning of each updating phase. For sake of effectiveness, a different subdomain is considered after a communication. As previously said, the subdomains assigned to a processor are treated cyclically according to a red-black ordering. Experimentally, this strategy turned out to be the most efficient one.

Computational experiments were carried out using an IBM-SP4 series machines located at IDRIS computing centers in Paris. More precisely, the main support of our experiments was an IBM-SP4 with twelve SMP nodes of thirty-two P690+ processors (at 1.3 Ghz); nodes are connected via a Federation network (1.6 Gbits per seconds). The latency of Federation network is between 5 and 7 micro-seconds and its bandwith is 2 Gbits per second for each node (see http://www.arcade-eu.org/overview/); note that the bandwidth is good since there are few processors per node. We have used up to 128 processors.

Figures 13, display the elapsed time of parallel iterative algorithms for different values of the number of processors in the case of 3D linear problems with the following convection parameters: 0.5, 1.5 and -0.5, $c = 10,$ and $\nu = 1,$ where ν is the diffusion parameter.

Figures 14 show the efficiency of parallel iterative algorithms in function of the number of processors.

The number of relaxations is given in Figures 15.

Parallel experimental results are summarized in table 2 and 3

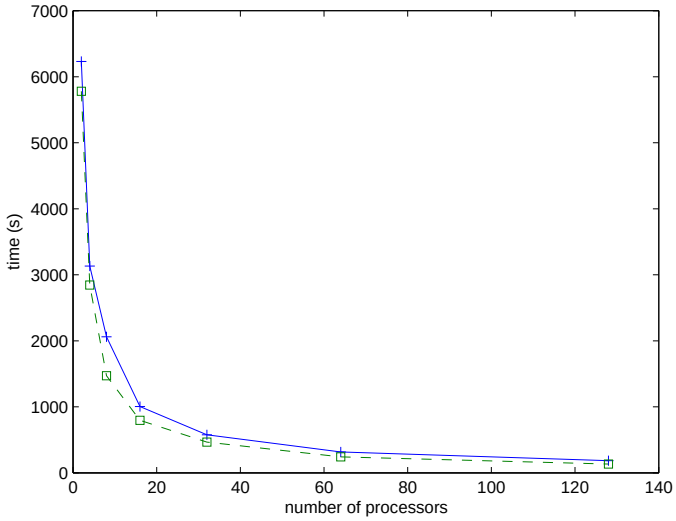

Figure 13: 3D problem, $\nu = 1$, 3,750,000 nodes, 256 subdomains, IBM-SP4 P690+, elapsed time of synchronous algorithms (solid) and asynchronous algorithms (dashed).

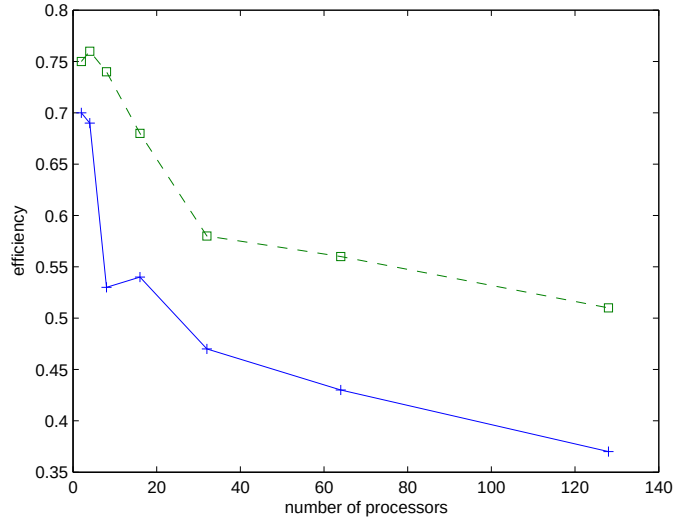

Figure 14: 3D problem, $\nu = 1$, 3,750,000 nodes, 256 subdomains, IBM-SP4 P690+, efficiency of synchronous algorithms (solid) and asynchronous algorithms (dashed).

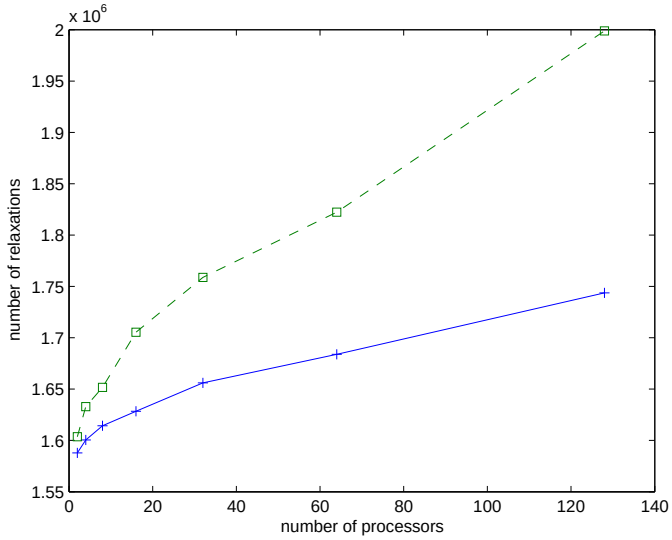

Figure 15: 3D problem, $\nu = 1$, 3,750,000 nodes, 256 subdomains, IBM-SP4 P690+, number of relaxations of synchronous algorithms (solid) and asynchronous algorithms (dashed).

procs.	time (sec.)	relaxations	speed-up	efficiency
Seq	8 681	1 573 740	-	-
As2	5 781	1 603 558	1.50	0.75
As4	2 846	1 632 888	3.05	0.76
As8	1 470	1 651 626	5.91	0.74
As16	796	1 705 320	10.91	0.68
As32	466	1 758 772	18.63	0.58
As64	241	1 822 306	36.02	0.56
As128	133	1 998 848	65.27	0.51

Table 2: Asynchronous Algorithm, $\nu = 1$

From Figures 15, we see that the number of relaxations increases with the number of processors. In the case of parallel synchronous schemes of computation, this phenomenon is mainly due to slight modifications in the order of treatment of the different subdomains; in the case of asynchronous schemes of computation, this fact is mainly due to the chaotic behavior of the algorithm. Note that asynchronous algorithms perform more relaxations than synchronous ones ; then, in asynchronous domain decomposition methods, boundary values of subdomains are exchanged with no order : thus, regarding the number of relaxations, convergence may be slower. We must note that despite higher numbers of relaxations, elapsed time of asynchronous parallel iterations are less than the elapsed time of synchronous ones. In other words,

procs.	time (sec.)	relaxations	speed-up	efficiency
Seq	8 681	1 573 740	-	-
Sy2	6 232	1 587 852	1.39	0.70
Sy4	3 132	1 600 460	2.77	0.69
Sy8	2 061	1 614 310	4.21	0.53
Sy16	1004	1 628 400	8.65	0.54
Sy32	577	1 656 002	15.05	0.47
Sy64	317	1 683 804	27.38	0.43
Sy128	185	1 743 758	46.92	0.37

Table 3: Synchronous Algorithm, $\nu = 1$

the withdrawal of synchronization can overcome slower convergence in number of relaxation. Asynchronism is an efficient way to deal with communication overhead and load unbalance, which are major issues in parallel computing. On the other hand, in the case of synchronous algorithm, as the number of subdomains is bounded to 256, the more processors is being used, the less subdomains are assigned to each one. The order in which boundary values are exchanged between the processors, varies as the assignment of the subdomains changes. This order does have a slight influence on the convergence speed of domain decomposition methods. Finally, we note that asynchronous algorithms with flexible communication are more efficient than synchronous algorithms. It turns out that the overhead generated by additional relaxations in the case of asynchronous algorithms is smaller than the synchronization overhead combined with processor idle time of parallel synchronous schemes of computation. Moreover, the efficiency of synchronous algorithms decreases faster than the efficiency of asynchronous algorithms when the number of processors increases.

7 Conclusion

In this chapter, we have studied the solution of linear and nonlinear boundary values problems via parallel Schwarz alternating method. We have shown the interest of introducing flexible asynchronous scheme of computation for various applications such as linear and nonlinear convection-diffusion problems, Hamilton-Jacobi-Bellman problem, obstacle problem, Navier-Stokes equations and coupled problems of continuous electrophoresis flow problem. We have also presented how to implement such parallel methods on a supercomputer. We have also shown that the use of persistent communication request with MPI library can lead to efficient implementation. Parallel synchronous and asynchronous iterative schemes of computation have been also compared. Computational results were displayed for test on IBM-SP series machines have clearly shown the benefits of using parallel algorithms and particularly the efficiency of parallel asynchronous Schwarz alternating method compared to the synchronous one.

References

[1] Lions, P. L., "On the Schwarz alternating method I", in "Proceedings of the First International Symposium on Domain Decomposition Methods for Partial Differential Equations", Glowinski, R., Golub, G. H., Meurant, G. A., Periaux, J., (Editors), SIAM, Philadelphia, 1-42, 1988.

[2] Lions, P. L., "On the Schwarz alternating method II", in "Proceedings of the Domain Decomposition Methods", Chan, T., Glowinski, R., Periaux, J., Widlund, O., (Editors), SIAM, Philadelphia, 47-70, 1989.

[3] Lions, P. L., " On the Schwarz alternating method III : a variant for nonoverlapping subdomains", in "Proceedings of Third International Symposium on Domain Decomposition Methods for Partial Differential Equations", Chan, T., Glowinski, R., Periaux, J., Widlund, O., (Editors), SIAM, Philadelphia, 6, 202-223, 1990.

[4] Dryja, M., "An additive Schwarz algorithm for two and three dimensional finite element elliptic problems", in "Proceedings of the third international symposium on domain decomposition methods for partial differential equations", Chan, T., Glowinski, R., Periaux, J., Widlund, O., (Editors), SIAM, Philadelphia, 147-153, 1989.

[5] Dryja, M., Widlund, O. B., "Some domain decomposition algorithms for elliptic problems", in "Proceedings of the conference on iterative methods for large linear systems", Hager, L., and al., (Editors), Academic Press, San-Diego, California, 273-291, 1989.

[6] Dryja, M., Widlund, O.B., "Toward a unified theory of domain decomposition algorithms for elliptic problems", in " Proceedings of the third international symposium on decomposition methods for partial differential equations", Chan, T., Glowinski, R., Periaux, J., Widlund, O., (Editors), SIAM, Philadelphia, 3-21, 1989.

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

[8] Chan, T., Mathew, T., *"Domain Decomposition algorithms"*, Acta Numerica, 61-143, 1994

[9] Quarteroni, A., Valli, A., " Domain Decomposition Methods for Partial Differential Equations", Oxford University Press, Oxford, 1999

[10] Smith, B., Bjoerstad, P., Gropp, W., "Domain Decomposition : parallel multilevel methods for elliptic partial differential equations", Cambridge University Press, 1996.

[11] Toselli, A., Widlund, O., "Domain Decomposition methods-Algorithms and Theory", Springer, 2005.

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

[13] El Baz, D., Frommer, A., Spiteri, P., *"Asynchronous iterations with flexible communication: contracting operators"*, Journal of Computational and Applied Mathematics, 176, 91-103, 2005.

[14] Frommer, A., Szyld, D., *"Asynchronous iterations with flexible communication for linear systems"*, Calculateurs Parallèles, Réseaux et Systèmes Répartis, 10, 421-429, 1998.

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

[16] Miellou, J. C., Spiteri, P., *" Un critère de convergence pour des méthodes générales de point fixe"*, R.A.I.R.O.- MMAN, 19, 645-669, 1985.

[17] Miellou, J. C., "Asynchronous iterations and order intervals", in " Proceedings of the international workshop on Parallel Algorithms and Architectures", Cosnard, M., (Editor), North Holland, 85-96, 1986.

[18] Spiteri, P., *"Parallel asynchronous algorithms for solving boundary value problems"*, in " Proceedings of the international workshop on Parallel Algorithms and Architectures", Cosnard, M., (Editor), North Holland, 73-84, 1986.

[19] Giraud, L., Spiteri, P., *"Résolution parallèle de problèmes aux limites non linéaires"*, M.2 A.N., 25, 73-100, 1991.

[20] Chazan, D., Miranker, W., *"Chaotic relaxation"*, Linear Algebra Appl., 2, 199-222, 1969.

[21] Miellou, J.C., *"Algorithmes de relaxation chaotique à retards"*, RAIRO Analyse numérique, 1, 55-82, 1975.

[22] Miellou, J.-C., *"Itérations chaotiques à retards, étude de la convergence dans le cas d'espaces partiellement ordonnés"*, C.R.A.S., Paris, 280, 233-236, 1975.
bibitemBau78 Baudet, G. M., *"Asynchronous iterative methods for multiprocessors"*, J.A.C.M., 25,226-244, 1978.

[23] El Tarazi, M. N., *"Some convergence results for asynchronous algorithms"*, Numerisch Mathematik, 39, 325-340, 1982.

[24] El Tarazi, M. N., *"Algorithmes mixtes asynchrones, étude de la convergence monotone"*, Numerisch Mathematik, 44, 363-369, 1984.

[25] Frommer, A., Schwandt, H., *"Asynchronous parallel methods for enclosing solutions of nonlinear equations"*, J. Comp. Appl. Math., 60, 47-62, 1995.

[26] Frommer, A., Schwandt, H., *" A Unified Representation and theory of Algebraic Additive Schwarz and Multisplitting Methods"*, SIAM J. Matrix Anal. Appl., 18, 893-912, 1997.

[27] Frommer, A., Schwandt, H., D. Szyld, D., *" Asynchronous weighted additive Schwarz methods*, Electron. Trans. Numer. Anal., 5, 48-61, 1997.

[28] Benzi, M., Szyld, D., *"Existence and uniqueness of splittings for stationary iterative methods with application to alternating methods"*, Numerische Mathematik, 76, 309-321, 1997.

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

[30] Frommer,A., Szyld, D., *"Weighted max norms, splittings and overlapping additive Schwarz methods""*, Num. Math, 83, 259-278, 1999.

[31] Frommer, A., Szyld, D., *"On asynchronous iterations"*, Journal of Computational and Applied Mathematics, 123, 201-216, 2000.

[32] Varga, R., "Matrix iterative analysis", Prentice Hall, Inc., 1962.
[33] Ortega, J. M., Rheinboldt, W. C., " Iterative Solution of Nonlinear Equations in Several Variables", Academic Press, New York, 1970.
[34] Evans, D.J., Deren, W., *"An asynchronous parallel algorithm for solving a class of nonlinear simultaneous equations"*, Parallel Computing, 17, 165-180, 1991.
[35] Axelson, O., Barker, V., "Finite Element Solution of Boundary Value Problems", Academic Press, Orlando, 1984.
[36] Spiteri, P., Miellou, J. C., El Baz, D., *"Parallel Schwarz method and multi-splitting methods for a nonlinear diffusion problem"*, Numerical Algorithms, 33, 461-474, 2003.
[37] Rheinboldt, W. C., *"On M-functions and their application to nonlinear Gauss-Seidel iterations and to network flows"*, J. Math. Anal. and Appl., 32, 274-307, 1970.
[38] Bernudez, A., *"Contrôle par feedback a priori de systèmes régis par des équations aux dérivées partielles"*, Rapport de Recherche INRIA, 288, 1978.
[39] Lions, P.L., *"Résolution des problèmes généraux de Bellman-Dirichlet"* , Comptes rendus à l'académie des Sciences, 287, sŕie A, 747-750, 1980.
[40] Lions, P.L., Mercier, B., *"Approximation numérique des équations d'Hamilton-Jacobi-Bellman"*, RAIRO Analyse numérique, 14, 369-393, 1980.
[41] Lions P.L., Mercier, B., *"Approximation numérique des équations de Hamilton-Jacobi-Bellman"*, R.A.I.R.O. Analyse numérique, 14, 369-393, 1980.
[42] Glowinski, R., Lions, J.L., Trḿolières, R., " Analyse numérique des inéquations variationnelles", Dunod, 1976.
[43] Duvaut, G., Lions, J.L., "Les inéquations en mécanique et en physique", Dunod, 1972.
[44] Spiteri, P., Miellou, J.C., El Baz, D., *"Asynchronous Schwarz alternating methods with flexible communicationfor the obstacle problem"*, Réseaux et systèmes répartis, 13, 47-66, 2001.
[45] Clifton, M.J., Roux-de-Balmann, H., Sanchez, V., *"Electrohydrodynamic Deformation Of The Sample Stream in Continuous-flow Electrophoresis With an AC Electric Field"*, The Canadian Journal Of Chemical Engineering, 70, 1055-1062, 1992.
[46] Clifton, M.J., Sanchez, V., *"Continuous-flow electrophoresis: numerical simulation of electrokinetics and electrohydrodynamics"*, in "43rd Congress of the International Astronautical Federation", Washington, DC, 1992 .
[47] Clifton, M.J., *"Numerical simulation of protein separation by continuous-flow electrophoresis"*, Electrophoresis, 14, 1284-1291, 1993.
[48] Issa, R.I., *"Solution of the implicitely discretised fluid flow equations by operator splitting"*, Journal of Computational Physics, 62, 40-65, 1986.
[49] Patankar, S.V., "Numerical heat transfer and fluid flow", Mc Graw-Hill, 1980.
[50] Spiteri, P., Boisson, H.C., *"A subdomain predictor-corrector algorithm for solving incompressible Navier-Stokes equation"*, in "Critical parameters and domain decomposition", Kaper, H.G., Garbey, M., (Editors), NATO ASI series, Kluwer academic publishers, 335-347, 1993.

[51] Guivarch, R., Spiteri, P., Boisson, H.C., Miellou, J.C., *"Schwarz alternating method parallel algorithm applied to incompressible flow computation in vorticity stream function formulation"*, Parallel Algorithm and Applications, 11, 205-225, 1997.

[52] Chau, M., Spiteri, P., Boisson, H.C., *"Parallel numerical simulation for the coupled problem of continuous flow electrophoresis"*, International Journal for Numerical Methods in Fluids, 54, ??, 2007.

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

[54] El Baz, D., Gazen, D., Miellou, J.C., Spiteri, P., *"Mise en oeuvre des méthodes itératives asynchrones avec communication flexible. Application à la résolution d'une classe de problèmes d'optimisation"*, Calculateurs Parallèles, 8-4, 393-411, 1996.

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

[56] Guivarch, R., Spiteri, P., *"Implémentation de méthodes de sous-domaines asynchrones avec PVM et MPI sur IBM SP2'*, Calculateurs Parallèles Réseaux et Systèmes Répartis, 10, 431-438, 1998.

[57] Chau, M., *"Algorithmes parallèles asynchrones pour la simulation numérique"*, Thèse de Doctorat, Institut National Polytechnique de Toulouse, Laboratoire d'Informatique et de Mathématiques Appliquées, IRIT / ENSEEIHT, 2005.

[58] Chau, M., El Baz, D., Guivarch, R., Spiteri, P., *"MPI implementation of parallel subdomain methods for linear and nonlinear convection-diffusion problems"*, Journal of Parallel and Distributed Computing, 67, 581-591, 2007.

[59] Guivarch, R., *"Résolution parallèle de problèmes aux limites couplés par des méthodes de sous-domaines synchrones et asynchrones"*, Thèse de Doctorat, Institut National Polytechnique de Toulouse, Laboratoire d'Informatique et de Mathématiques Appliquées, IRIT / ENSEEIHT, 1997.

[60] Miellou, J.C., Spiteri, P., El Baz, D., *"Stopping criteria, forward and backward errors for perturbed asynchronous linear fixed point methods in finite precision"*, IMA Journal of Numerical Analysis, 25, 429-442, 2005.

[61] Bertsekas, D. P., *"Distributed asynchronous computation of fixed points"*, Mathematical Programming, 27, 107-120, 1983.