

Asynchronous Schwarz methods applied to constrained mechanical structures in grid environment



M. Chau^a, T. Garcia^{b,c}, P. Spiteri^{b,*}

^aAdvanced Solutions Accelerator, 199 rue de l'Oppidum, 34170 Castelnau le Lez, France

^bENSEEIH-IRIT, 2 rue Charles Camichel, 31071 Toulouse CEDEX, France

^cUVSQ-PRISM, 45 avenue des états-unis, 78035 Versailles Cedex, France

ARTICLE INFO

Article history:

Received 14 January 2013

Received in revised form 17 March 2014

Accepted 27 March 2014

Available online 8 May 2014

Keywords:

Obstacle problem

Parallel algorithm

Synchronous method

Asynchronous method

Complementary problem

Schwarz alternating method

Constrained mechanical structures

ABSTRACT

This paper deals with the parallel solution of the stationary obstacle problem with convection–diffusion operator. The obstacle problem can be formulated by various ways and in the present study it is formulated like a multivalued problem. Another formulation by complementary problem is also considered. Appropriate discretization schemes are considered for the numerical solution on decentralised memory machines by using parallel synchronous and asynchronous Schwarz alternating algorithms. The considered discretization schemes ensure the convergence of the parallel synchronous or asynchronous Schwarz alternating methods on one hand for the solution of the multivalued problem and on the other hand for the solution of the complementary problem. Finally the implementation of the algorithms is described and the results of parallel simulations are presented.

© 2014 Elsevier Ltd. All rights reserved.

Introduction

The behavior of mechanical structures is classically modelled by the equations governing the elasticity field [1,2]. The problem consists in studying the displacement of mechanical structure submitted to external forces. For physical reasons, the displacement of the structures may be submitted to some constraints, for example the displacement u cannot be less than a given threshold ϕ . The modelling of such constrained problem is very complex and leads to the solution of nonlinear boundary value problem, called in applied mathematics the obstacle problem. In fact this previous problem leads to the solution of a variational inequality. Variational inequalities arise from many physical and economical applications such as fluid flow in porous media, the behavior of elasto-plastic materials and lubrication phenomena in mechanics like free boundary problems and valuation of American options pricing in modern finance.

The obstacle problem has been studied in many contributions and by many authors. For example in [3] sequential methods are considered for the Hamilton–Jacobi–Bellman problem; in [4] a

study concerning the rate of convergence when multilevel domain decomposition and multigrid methods are considered for sequential solution of the obstacle problem. In [5] the linear convergence is proved for sequential multiplicative Schwarz method applied to the solution of variational inequalities and in [6] a geometric convergence rate is established for sequential additive Schwarz method for the same problem; in [7] the convergence rate analysis of domain decomposition methods is also studied in the case of obstacle problem solution. In [8] the block relaxation methods for algebraic obstacle problems with M-matrices are considered while in [9] multiplicative and additive Schwarz methods for obstacle problems with convection–diffusion operators, when two (possibly overlapping) subdomains are considered, one, where the solution is equal to a given obstacle function, and the other, where the solution satisfies a linear equation. In [10] the sequential convergence analysis of the generalized Schwarz method for solving the obstacle problems with T-monotone operator is studied. In the previous contributions, note that, mainly, the study of numerical solutions is considered by using sequential subdomain methods; moreover when the parallel solution is considered for the solution of the obstacle problem, it can be noticed that only parallel synchronous subdomain methods are studied. In [11] the rate of convergence of parallel asynchronous method is studied in the context of convex optimization; the considered work includes as particular cases parallel domain decomposition and multigrid

* Corresponding author. Tel.: +33 0534322166; fax: +33 0534322157.

E-mail addresses: mchau@asa-sas.com (M. Chau), Thierry.Garcia@enseeiht.fr, thierry.garcia@uvsq.fr (T. Garcia), Pierre.Spiteri@enseeiht.fr (P. Spiteri).

methods for solving elliptic partial differential equations and is closely related to relaxation methods for nonlinear network flow.

In the present study we will solve the model problem by the Schwarz alternating method. Thus, it is necessary to take into account the constraints and so to project the relaxed updates computed by the previous iterative algorithm on a convex set which defines such constraints. Thus, this projection has been achieved by various ways and in the proposed study we will consider two distinct ways allowing to project the relaxed updates. On other side, it can be noted that the discretization of boundary value problems arising from the formulation of the obstacle problem, leads to the solution of large scale algebraic systems. In order to compute efficiently the solution of such large algebraic systems, supercomputers can be used. The present study is devoted to the parallel solution by synchronous or asynchronous relaxation algorithms implemented on distributed architectures, mainly in Grid computing environment [12]. Recall that the asynchronous relaxation method corresponds to a general scheme of computation where the computations are performed in parallel without order nor synchronization among the processors [13]. Then idle times due to the synchronizations among the processors are suppressed; consequently, since the synchronizations are time consuming, the elapsed time of computation decreases when parallel asynchronous relaxation algorithms are performed.

In our previous works, the solution of the obstacle problem has been computed by various parallel synchronous or asynchronous algorithms implemented on various architectures. In [14–17] the parallel synchronous and asynchronous projected Richardson's method has been studied and implemented; since the projection operator is contracting, if appropriate additional assumptions hold, the convergence of the projected Richardson's method is ensured. Nevertheless, in the general model of the obstacle problem, when the gradient of the displacement u occurs, i.e. when a convection–diffusion operator models the studied problem, this previous algorithm cannot be applied directly in the context of the projected Richardson's method. Indeed the use of the Richardson's method needs a formulation of the obstacle problem as a constrained optimization problem on a convex set and is not directly suitable, since in such formulation the cost function must be defined thanks to a symmetric stationary operator. Moreover, the projected Richardson's method is not very efficient.

Besides domain decomposition methods are well suited to the parallel solution of boundary values problems. In [16,17], we have considered for the solution of the obstacle problem an iterative parallel asynchronous projected subdomain method without overlapping among the subdomains; the subdomains are here defined by gathering several adjacent blocks of the discretization matrix.

For both previous mathematical formulations, the convergence of parallel synchronous and asynchronous algorithms applied to the solution of the discretized problem, can be proved, either for every splitting of the problem to solve, by contracting techniques [18,19] or by partial ordering techniques [20,21]. The reader is referred to the previous works for more details.

The present study is based on various different formulations of the model problem taking into account the constraints on the expected solution. Indeed, we concentrate on the obstacle problem modelled with convection–diffusion operator. In a distinct mathematical context we consider in the present paper the use of parallel synchronous or asynchronous Schwarz alternating method for the solution of the obstacle problem. Recall that, in the considered Schwarz method the domain is decomposed into subdomains which overlap each other. So the present work is distinct from previous studies concerning the solution of the obstacle problem. A general approach is considered, linked to the numerical parallel solution of the obstacle problem, well adapted and very different than the ones considered from [3–11]. By using the asynchronous Schwarz

alternating method we hope to obtain the more multiplicative possible behavior of the domain decomposition method; recall that the additive and the multiplicative Schwarz alternating methods may be viewed respectively as Jacobi and Gauss–Seidel type methods applied to partial differential equations.

A main property ensuring the convergence of parallel asynchronous iterative method is related to the fact that after appropriate discretization, the spatial part of the operator leads to a discretization matrix which is an M-matrix [22]. On the other hand, the projection on a convex set can be also formulated by the perturbation of the continuous convection–diffusion operator by a multivalued increasing diagonal operator [23]; so, the convergence of the considered method can also be analyzed by combining the monotony of the diagonal operator with the main property of the discretization matrix (see [19]). Another distinct theoretical approach consists in considering the obstacle problem written by a complementary formulation of this problem (which in fact corresponds to the projection of the intermediate values on the convex set which defines the constraints) and in solving the linearized algebraic system obtained by the Howard process, by the parallel asynchronous Schwarz alternating method. Note also that the present study is distinct to the one considered in [24], since in this previous work we have considered the parallel asynchronous Schwarz alternating method with flexible communications analyzed by partial ordering techniques, this last kind of communication mode among the processors which, possibly, consumes time when the rate of convergence is low; here, the convergence of the parallel asynchronous method follows from the fact that the matrix of the discretized and linearized problem is an M-matrix. Moreover, in order to obtain good efficiencies, in the presented work we consider only parallel experiments with the classical asynchronous schemes. Then we have to solve either a multivalued problem either a linearized problem by using the Howard process. So we obtain very interesting properties of discretized problems which ensure easily the convergence of parallel synchronous or asynchronous Schwarz alternating algorithms (see [18,20,21,25]).

Implementation of the considered algorithms is carried out on Grid'5000, the French national grid (see [12]). This architecture allows us to study the behavior of parallel algorithms on distant and heterogeneous clusters. Communications are managed with MPI facilities. Asynchronous and synchronous efficiencies of the studied parallel algorithms are compared. Nevertheless, in such experimental context of distant and heterogeneous clusters, the classical notions of speed-up and efficiency are not relevant. Thus, we define a new parameter which permit us to measure the real efficiency of the studied asynchronous parallel algorithms. Moreover, in the parallel experiments, we have also considered the impact of the communications on the elapsed time of the iterative parallel algorithms for both synchronous and asynchronous mode.

The present study is organized as follows. In Section 'The obstacle problem' several useful formulations of the obstacle problem are presented. In Section 'Numerical solution of the model problem' appropriate discretization schemes ensuring the convergence of the parallel synchronous or asynchronous Schwarz alternating method are considered for both formulations of the model problem. In Section 'Parallel simulation' the implementation of the algorithms and the results of the parallel experiments are presented. Finally some concluding remarks are given.

The obstacle problem

The classical obstacle problem [1–11] occurs in many applications such as mechanics, free boundary problems and financial derivatives. In the present study we concentrate mainly on the obstacle problem occurring in civil engineering. The obstacle

problem consists in computing a function u , whose characterization is given by the following time dependent nonlinear problem

$$\begin{cases} \frac{\partial u}{\partial t} - b \cdot \Delta u + c \cdot u - f \geq 0, & u \geq \phi, \text{ everywhere in } [0, T] \times \Omega, \\ (\frac{\partial u}{\partial t} - b \cdot \Delta u + c \cdot u - f)(u - \phi) = 0, & \text{everywhere in } [0, T] \times \Omega, \\ u(0, x, y, z) = u_0(x, y, z), \\ \text{B.C. for } u(t, x, y, z) \text{ defined on } \partial\Omega, \end{cases} \quad b > 0, \quad c \geq 0, \quad (1)$$

where u_0 is the initial condition, b is a physical parameter, T is the final time, $u = u(t, x, y, z)$ is the displacement of the constrained mechanical structure submitted to external strains, f is the external force imposed to the structure, B.C. describes the boundary conditions on the boundary $\partial\Omega$ of the domain Ω and ϕ models a constraint imposed to u . Practically, the Dirichlet condition (where u is fixed on $\partial\Omega$) or the Neumann condition (where the normal derivative of u is fixed on $\partial\Omega$) are classically considered.

Remark 1. When the materials constituting the structure are not homogeneous, then the coefficients arising in the mathematical model are not constant. Consequently, we can also consider the general formulation of the obstacle problem in which, after appropriate derivation with respect to the spatial variables, some terms of convection may appear. Then, we consider hereafter a simplified formulation of the obstacle problem, using a convection–diffusion operator.

$$\begin{cases} \frac{\partial u}{\partial t} + a^t \cdot \nabla u - b \cdot \Delta u + c \cdot u - f \geq 0, & u \geq \phi, \text{ everywhere in } [0, T] \times \Omega, \quad b > 0, \quad c \geq 0, \\ (\frac{\partial u}{\partial t} + a^t \cdot \nabla u - b \cdot \Delta u + c \cdot u - f)(u - \phi) = 0, & \text{everywhere in } [0, T] \times \Omega, \\ u(0, x, y, z) = u_0(x, y, z), \\ \text{B.C. for } u(t, x, y, z) \text{ defined on } \partial\Omega, \end{cases} \quad (2)$$

where a is a vector describing physical parameters. Consequently, we consider two situations in which on one hand the diffusion operator appears like that in (1) and, on the other hand, the convection–diffusion operator is considered in the formulation of the problem. This last situation is more general and our study can take into account the two situations.

For the sake of generality, we consider, in what follows the more general formulation of the obstacle problem, expressed with the convection term. The previous time dependent problem is solved numerically by considering an implicit or semi-implicit time marching scheme, where at each time step the following stationary nonlinear problem is solved.

$$\begin{cases} a^t \cdot \nabla u - b \cdot \Delta u + (c + \delta) \cdot u - g \geq 0, & u \geq \phi, \text{ everywhere in } [0, T] \times \Omega, \quad b > 0, \\ (a^t \cdot \nabla u - b \cdot \Delta u + (c + \delta) \cdot u - g)(u - \phi) = 0, & \text{everywhere in } [0, T] \times \Omega, \\ \text{B.C. for } u(t, x, y, z) \text{ defined on } \partial\Omega, \end{cases} \quad (3)$$

where δ is the inverse of the time step and $g = f + \delta u^{\text{prec}}$, where u^{prec} is the solution obtained at the previous time step.

Remark 2. In Fig. 1 we can see a mechanical interpretation of the obstacle problem. Indeed, in Fig. 1(a), a thread is fixed at its own extremities and is subjected to an external force; due to their effect, the thread has a displacement submitted to some constraints. So we can see that there exists a contact area \mathcal{O} in which the value of the displacement u is identical to the value of the

constraint ϕ whereas the displacement is free outside of this contact area. Indeed, classically, if $a^t \cdot \nabla u - b \cdot \Delta u + (c + \delta) \cdot$

$u - g < 0$, then $u = \phi$; on the contrary if $a^t \cdot \nabla u - b \cdot \Delta u + (c + \delta) \cdot u - g \geq 0$, then the constraints are not saturated and the solution of (3) is free. So, classically, due to the lack of constraints, in this last case where $u > \phi$ and the operator $a^t \cdot \nabla u - b \cdot \Delta u + (c + \delta) \cdot u - g$ is null; thus the physical problem is well described. From an industrial point of view, the Fig. 1(b) shows an example of application concerning the study of crash test arising in the self-propelling construction.

For each stationary problem many equivalent formulations of the obstacle problem exist. In the present study, we concentrate on the variational formulation associated with problem (3); classically the corresponding variational formulation is given as follows

$$\begin{cases} \text{Find } u \in K \text{ such that} \\ a(u, w - u) \geq L(w - u), \quad \forall w \in K, \end{cases} \quad (4)$$

where

$$a(u, w) = \int_{\Omega} (b \cdot \nabla u \cdot \nabla w + a^t \cdot \nabla u \cdot w + (c + \delta) \cdot u \cdot w) dx dy dz,$$

$$L(w) = (f, w) = \int_{\Omega} f \cdot w dx dy dz,$$

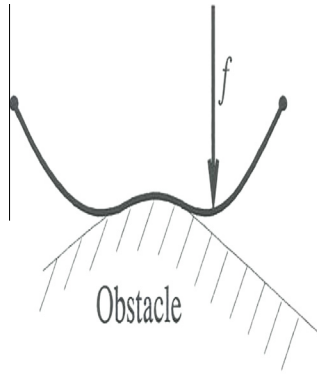
and

$$K = \{w | w \text{ given in } \Omega, \text{ such that } w(x, y, z) \geq \phi(x, y, z) \text{ everywhere on } \Omega\}.$$

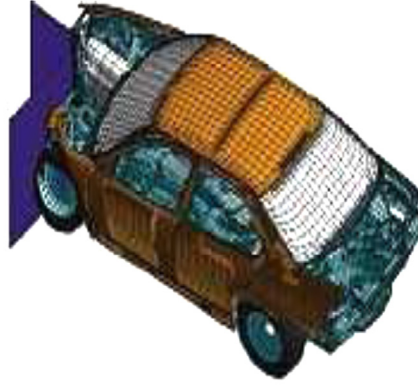
Let us consider the stationary variational inequality (4). Then, since classically

1. the space $\mathcal{H}^1(\Omega)$ normed by $\|w\|_{1,\Omega}^2 = \int_{\Omega} (\nabla w \cdot \nabla w + w \cdot w) dx dy dz$, is an Hilbert space and K is a closed convex set,
2. the mapping $(u, w) \rightarrow a(u, w)$ is obviously a bilinear, continuous and elliptic form,
3. the mapping $w \rightarrow L(w)$ is obviously a linear continuous form,

then, by applying the Stampacchia theorem, the variational inequality (4) has a unique solution.



(a) displacement of a thread



(b) crash test

Fig. 1. Mechanical interpretation of the obstacle problem for the constrained displacement of a thread (a) and of a crash-building of a self-propulsing vehicle (b).

In the case of the evolution problem (1) or (2), the reader is referred to [26] for a result of existence and uniqueness of the solution.

We can also apply the Riesz representative theorem. Let us denote by $E = \mathcal{H}^1(\Omega)$ the real vector space and by E' its dual space; in our case E' is identified with E , since $\mathcal{H}^1(\Omega)$ is an Hilbert space. Then, there exists a unique element denoted by $\bar{A}u \in E'$, where \bar{A} is a continuous linear mapping from $E \rightarrow E'$, such that

$$a(u, w) = \langle w, \bar{A}u \rangle_{E \times E'}, \quad \forall w \in \mathcal{H}^1(\Omega);$$

similarly, there exists also a unique element, denoted by $\bar{g} \in E'$ such that

$$L(w) = \langle w, \bar{g} \rangle_{E \times E'}, \quad \forall w \in \mathcal{H}^1(\Omega);$$

then the stationary variational inequality (4) can be written as follows

$$\begin{cases} \text{Find } u \in K \text{ such that} \\ \langle w - u, \bar{A}u - \bar{g} \rangle_{E \times E'} \geq 0, \quad \forall w \in K. \end{cases} \quad (5)$$

Remark 3. From a practical point of view, note that for the studied stationary problem (4) $\bar{A}u(x, y, z) = -b \cdot \Delta u(x, y, z) + a^t \cdot \nabla u + (c + \delta) \cdot u(x, y, z)$ and $\bar{g} = g(x, y, z)$.

Sub-differential mapping

The notion of sub-differential mapping will play a major role in the following study. So we recall hereafter this notion and the main properties associated (see [1,23]).

Definition 1. Let be χ a given convex function on E and a point $u \in E$; we denote by $\partial\chi(u)$ the set of all $u' \in E'$ such that

$$\chi(v) \geq \chi(u) + \langle v - u, u' \rangle_{E \times E'}, \quad \text{for every } v \in E, \quad (6)$$

where $\langle \cdot, \cdot \rangle_{E \times E'}$ denotes the pairing between E and E' . Such element u' is called sub-gradient of χ at u , and $\partial\chi(u)$ is called the sub-differential of χ at u .

Remark 4. Recall that the pairing between E and E' is a bilinear form, from $E \times E'$ onto \mathbb{R} (or possibly \mathbb{C}). If E is an Hilbert space, then the pairing is the inner product of E .

Remark 5. Let χ be a Gateaux differentiable (or Frechet differentiable) convex mapping at u . Then $\partial\chi(u)$ consists of a single element, namely the Gateaux (or Frechet) differential of χ at u (see

[23]). From (6), it is obvious that $\partial\chi(u)$ is a closed convex set (possibly empty, see [23]).

In the sequel, we will use a multivalued formulation of the model problem (4) which plays a major role for the studied non-linear obstacle problem.

Lemma 1. Let $u \in E$ be such that $\chi(u) = \min_{v \in E}(\chi(v))$; then u is a minimum if and only if $0 \in \partial\chi(u)$.

Proof. Indeed, let $u \in E$ such that $\chi(u) \leq \chi(v)$, $\forall v \in E$; then we have

$$\chi(v) \geq \chi(u) + \langle v - u, 0 \rangle_{E \times E'},$$

and then $0 \in \partial\chi(u)$. \square

Lemma 2. The sub-differential $\partial\chi(u)$ is a monotone operator (in general multivalued) from E to E' .

Proof. Let $w' \in \partial\chi(w)$; then $\chi(v) \geq \chi(w) + \langle v - w, w' \rangle_{E \times E'}$, $\forall v \in E$. Let also $u' \in \partial\chi(u)$; then $\chi(v) \geq \chi(u) + \langle v - u, u' \rangle_{E \times E'}$, $\forall v \in E$. Let us consider the first inequality for $v = u$ and the second for $v = w$; then by adding, we obtain $\langle w - u, w' - u' \rangle_{E \times E'} \geq 0$, and the proof is achieved. \square

The indicator function of the convex subset K will also play an important role in the sequel and is defined as follows.

Definition 2. Let K be a closed convex subset of E . The indicator function of the convex subset K denoted ψ_K is then defined by

$$\psi_K(v) = \begin{cases} 0 & \text{if } v \in K \\ +\infty & \text{otherwise.} \end{cases}$$

Clearly, $\psi_K(v)$ is convex. By definition of the sub-differential we have (see [23])

$$\partial\psi_K(v) = \{v' \in E' \mid \langle v - w, v' \rangle_{E \times E'} \geq 0, \text{ for every } w \in K\}.$$

This shows that the domain of the subdifferential is $D(\partial\psi_K) = D(\psi_K) = K$ and $\partial\psi_K(v) = \{0\}$ for each $v \in \text{int}(K)$. Moreover, if v lies on the boundary of K , then $\partial\psi_K(v)$ coincides with the cone of normal to K at point v . So we can summarize by giving bellow the expression of the sub-differential indicator function of the convex subset K

$$\partial\psi_K(v) = \begin{cases} \emptyset & \text{if } v < \phi, \\] -\infty, 0] & \text{if } v = \phi, \\ 0 & \text{if } v \in K, \end{cases}$$

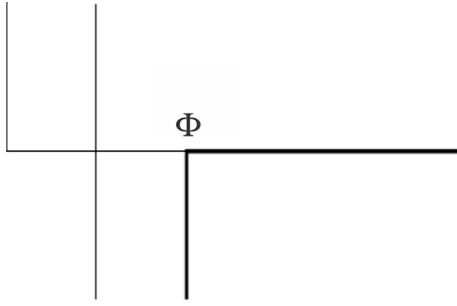


Fig. 2. Graph of the sub-differential of the indicator function.

and the corresponding graph presented on the following Fig. 2.

Equivalent formulation of the obstacle problem

Then, thanks to the notions presented in the previous subsection, the formulation (5) of the continuous stationary variational inequality (4) is equivalent to the following multivalued problem

$$\begin{cases} \text{Find } u \in E \text{ such that} \\ \bar{A}u - \bar{g} + \partial\psi_K(u) \ni 0, \end{cases} \quad (7)$$

where $\partial\psi_K(u)$ is the sub-differential of the indicator function of the convex subset K . Note that, in (7), the projection on the convex set K can be also formulated by the perturbation of the continuous convection–diffusion operator by a multivalued increasing diagonal operator. Indeed, the definition of the sub-differential of the indicator function implies

$$\partial\psi_K(u) = \{\omega' \in E' \mid \psi_K(w) - \psi_K(u) \geq \langle w - u, \omega' \rangle_{E \times E'}, \forall w \in K\}.$$

Let us assume that (5) holds and let us verify that the following inequality is true

$$\psi_K(w) - \psi_K(u) \geq \langle w - u, \bar{g} - \bar{A}u \rangle_{E \times E'}, \quad \forall w \in K. \quad (8)$$

If $w \in K$, then $\psi_K(w) = 0$ and since $\psi_K(u) = 0$, then we obtain

$$0 \geq \langle w - u, \bar{g} - \bar{A}u \rangle_{E \times E'},$$

which is valid by considering (5).

If $w \notin K$, then $\psi_K(w) = +\infty$; since, $\psi_K(u) = 0$, then we obtain

$$+\infty - 0 \geq \langle w - u, \bar{g} - \bar{A}u \rangle_{E \times E'},$$

and the obtained inequality is always true.

Conversely, consider two distinct situations:

1. if there exists $u \in E$ such that $\bar{g} - \bar{A}u \in \partial\psi_K(u)$ then we have to verify that there exists $u \in E$ such that (8) holds for all $w \in K$. If $u \in K$ then $\psi_K(u) = 0$ and (8) can be written as follows

$$\psi_K(w) \geq \langle w - u, \bar{g} - \bar{A}u \rangle_{E \times E'}, \quad \forall w \in K.$$

Since $w \in K$ then $\psi_K(w) = 0$ and (8) can be written as follows

$$0 \geq \langle w - u, \bar{g} - \bar{A}u \rangle_{E \times E'}, \quad \forall w \in K,$$

inequality well verified by (5);

2. besides, let us show that the assertion $u \notin K$ is impossible. Indeed, in this case $\psi_K(u) = +\infty$, so that

$$\psi_K(w) - \infty \geq \langle w - u, \bar{g} - \bar{A}u \rangle_{E \times E'}, \quad \forall w \in K,$$

which involves, since $\psi_K(w) = 0$, that

$$-\infty \geq \langle w - u, \bar{g} - \bar{A}u \rangle_{E \times E'}, \quad \forall w \in K,$$

inequality never verified.

In conclusion, the problems (7) and (8) are equivalent. So the numerical solution of the obstacle problem leads to the solution of the multivalued problem (7).

Numerical solution of the model problem

Discretization

Let us consider the classical stationary linear convection–diffusion operator similar to the one arising in (2) and defined in a bounded domain Ω included in the 3D space. For the sake of simplicity, let us assume that the discretization grid of the domain Ω is uniform. In the sequel h will denote the discretization step-size. The discretization of the operators is made according to the following rules: the Laplacian is discretized with the classical seven points scheme; the first derivatives, for example the derivative with respect to x , is discretized as follows according to the sign of the component a_1 of the vector a

$$\frac{\partial u}{\partial x} = \begin{cases} \frac{u(x,y,z) - u(x-h,y,z)}{h} + \mathcal{O}(h), & \text{if the coefficient } a_1 \text{ is strictly positive,} \\ \frac{u(x+h,y,z) - u(x,y,z)}{h} + \mathcal{O}(h), & \text{if the coefficient } a_1 \text{ is strictly negative,} \end{cases} \quad (9)$$

and accordingly for the other first derivatives.

Let A denote the discretization matrix of the considered convection–diffusion operator; let us also denote by \bar{G} the corresponding right hand side of the discretized system. Since $c + \delta$ is strictly positive, then regardless the sign of the components of the vector a , it follows from (9) that the off-diagonal entries of the matrix A are nonpositive and the diagonal entries of the matrix A are positive. Thus, the matrix A is strictly (and irreducibly by using the characterization of irreducible matrices (see [22])) diagonally dominant; thus, A is a nonsingular M-matrix and $A^{-1} \geq 0$ i.e. all the entries of the inverse of the matrix A are nonnegative (see [22]).

Let us now consider the discretization of the global problem (7); taking into account the nonlinear part of the problem due to the perturbation of the linear algebraic system by the multivalued sub-differential operator $\partial\psi_K$ of the indicator function ψ_K , then the discretized problem can be written as follows

$$AU - G + \partial\Psi_K(U) \ni 0, \quad (10)$$

where $\partial\Psi_K(U)$ is the discretized sub-differential operator of the indicator function.

Remark 6. Since the sub-differential is a diagonal operator, the components of $\partial\Psi_K(U)$ are constituted by all the discretized sub-gradients belonging to the sub-differential of the indicator function.

The Schwarz alternating method

The use of a purely implicit time marching scheme and the considered discretization of the spatial part of the operators lead, at each time step, to the numerical solution of very large algebraic systems. Owing to the great size and also of the sparsity of such a system, iterative algorithm may be used in order to reduce the computation time at each time step. Moreover the parallelization of such iterative algorithm can reduce again the elapsed time of computation.

In the present study, we concentrate on the implementation of the iterative Schwarz alternating algorithm for the solution of the stationary obstacle problem. In such method, at each time step, we consider the classical projected Schwarz alternating method applied to the solution of each stationary variational inequality.

Note that the Schwarz alternating algorithm with overlapping is well suited for efficient parallel computation (see [27]). Note also that we can consider parallel synchronous and asynchronous Schwarz alternating algorithms. The interest of asynchronous algorithm, compared to the synchronous one, is the reduction of idle times due to synchronizations among the processors.

In this kind of method, in order to parallelize the computation, the domain Ω , is split into parallelepiped subdomains which overlap each other. Thus, for the computation of the solution of the global problem, smaller subproblems are solved on each processor of a parallel computer and then more accuracy can be obtained since the problem to be solved is ill-conditioned. For a considered subdomain, the restriction of the available values of the computed solution obtained on the neighboring subdomains are taken as Dirichlet boundary conditions. The synchronous algorithm consists in performing a block Jacobi like method. Besides, in the classical asynchronous algorithm, the components computed by the other processors may be delayed (see [18,20,25]) and the available values are used at the end of each relaxation.

More precisely, let us consider the system of algebraic Eq. (10). The numerical solution of (10) by the Schwarz alternating method leads to the solution of the following multivalued problem

$$\mathcal{A} \cdot \mathcal{U} - \mathcal{G} + \partial \bar{\Psi}_K(\mathcal{U}) \ni 0, \quad (11)$$

where \mathcal{A} , \mathcal{V} , \mathcal{G} and $\bar{\Psi}_K$ are derived from the augmentation process associated with the Schwarz alternating method [28]. This process is a theoretical model that represents the solution of the algebraic system (10) by a Schwarz domain decomposition method. In the implementation of the algorithms \mathcal{A} , \mathcal{G} and $\bar{\Psi}_K$ are not explicitly computed. According to a result of Evans and Deren [28], the matrix \mathcal{A} is also an M-matrix. So, the system (11) derived from the augmentation process, has the same property as the initial algebraic system (10), i.e. \mathcal{A} is an M-matrix and $\partial \bar{\Psi}_K(\mathcal{U})$ is a monotone operator.

Let us consider the system of algebraic Eq. (10) to solve. Let $\alpha \in \mathbb{N}$ be a positive integer and consider now the following block decomposition of problem (11) into α subproblems associated to the subdomain decomposition

$$\sum_{j=1}^{\alpha} \mathcal{A}_{ij} \cdot \mathcal{U}_j + \partial \bar{\Psi}_{K_i}(\mathcal{U}_i) - \mathcal{G}_i \ni 0, \quad \forall i \in \{1, \dots, \alpha\} \quad (12)$$

where $\mathcal{U}_i \in \mathbb{R}^{n_i}$, $\mathcal{G}_i \in \mathbb{R}^{n_i}$ and $\partial \bar{\Psi}_{K_i}(\mathcal{U}_i)$ is the i -th block of the sub-differential mapping, where n_i denotes the size of the i -th block of the previous vectors and $\mathcal{A} = (\mathcal{A}_{ij})_{1 \leq i, j \leq \alpha}$ is defined according to the associated block decomposition.

Let us now consider the solution of subproblems (12) by the classical asynchronous parallel iteration which can be written as follows

$$\begin{cases} \mathcal{A}_{ii} \cdot \mathcal{U}_i^{r+1} + \partial \bar{\Psi}_{K_i}(\mathcal{U}_i^{r+1}) \ni \mathcal{G}_i - \sum_{j \neq i} \mathcal{A}_{ij} \cdot \mathcal{W}_j, & \text{if } i \in s(r), \\ \mathcal{U}_i^{r+1} = \mathcal{U}_i^r, & \text{if } i \notin s(r), \end{cases} \quad (13)$$

where $\{\mathcal{W}_1, \mathcal{W}_2, \dots, \mathcal{W}_\alpha\}$ are the available values of the components $(\mathcal{U}_j)_{j \neq i}$, computed by the other processors and corresponding to an iterative method where the updates are performed at the end of any block relaxation by an asynchronous way, each component being defined by $\mathcal{W}_j = \mathcal{U}_j^{\rho_j(r)}$ and $\mathcal{S} = \{s(r)\}_{r \in \mathbb{N}}$ is a sequence of non-empty subsets of $\{1, 2, \dots, \alpha\}$. In other words, $s(r)$ is the subset of the subscripts of the components updated at the r -th relaxation. In the sequel, we have also to consider the vector

$$\mathcal{R} = \{\rho_1(r), \dots, \rho_\alpha(r)\}_{r \in \mathbb{N}},$$

which denotes a sequence of integer vectors from \mathbb{N}^α . In order to take into account the asynchronism among the processors, \mathcal{R}

models the delays between the parallel updates of each component, at the r -th relaxation. Furthermore \mathcal{S} and \mathcal{R} satisfy the following assumptions

$$\forall i \in \{1, 2, \dots, \alpha\}, \quad \text{the set } \{r \in \mathbb{N} | i \in s(r)\} \text{ is unbounded,}$$

$$\forall i \in \{1, 2, \dots, \alpha\}, \quad \forall r \in \mathbb{N}, \quad 0 \leq \rho_i(r) \leq r,$$

$$\forall i \in \{1, 2, \dots, \alpha\}, \quad \lim_{r \rightarrow \infty} \rho_i(r) = +\infty.$$

Remark 7. Classically, when $\rho_i(r) = r$ for all $i \in \{1, \dots, \alpha\}$ and for all $r \in \mathbb{N}$, then (13) models a synchronous Schwarz alternating method.

This formulation (10)–(12) by a multivalued sub-problem (12) takes some interest only from a theoretical point of view. Indeed, since the sub-differential of the indicator function is monotone and diagonal, the previous formulation allows to prove the convergence of the parallel synchronous and asynchronous iterations [18–21].

Recall that the matrix arising in the algebraic linear system, is an M-matrix; the corresponding algebraic linear system is perturbed by a monotone diagonal operator. Then, for any subdomain decomposition, according to theoretical results obtained in [18,19,25], since it can be proved that the uniform weighted norm of the error at the step r is decreasing, the numerical solution of problem (10) by the classical parallel asynchronous Schwarz alternating method associated with (13) and starting from any initial guess \mathcal{U}^0 , converges to the solution of the obstacle problem. Reference is made to these three last papers for more details concerning these results. Note also that the same result can also be obtained by partial ordering analysis (see [20,21]), since the mapping $\mathcal{A}\mathcal{U} - \mathcal{G} + \partial \bar{\Psi}_K(\mathcal{U})$ is obtained by the perturbation of a linear system with an M-matrix by a monotone diagonal operator, and then is an M-function, i.e. a mapping off-diagonally antitone and inverse isotone (see [22]).

Sometimes, the implementation of parallel algorithms requires the use of lexicographical ordering or red-black ordering of the columns of the mesh. The previous convergence results still hold, in both cases of natural and red-black ordering of the subdomains. Let us denote by A and \bar{A} the corresponding matrices associated with the two considered orderings. Let us assume that if A is an M-matrix, then \bar{A} is also an M-matrix since \bar{A} is obtained from A by a permutation matrix P which preserves the sign of the entries and particularly the sign of the diagonal entries. Furthermore we have $\bar{A} = P \cdot A \cdot P^t$; thus we have $\bar{A}^{-1} = P \cdot A^{-1} \cdot P^t$. The matrix A being an M-matrix, it follows that A^{-1} is a nonnegative matrix [22], i.e. the entries of A^{-1} are nonnegative. Thus \bar{A}^{-1} is also a nonnegative matrix obtained from A^{-1} by the same permutation as the one considered for A . Then \bar{A} is an M-matrix. Moreover when a red-black ordering of the blocks is considered, since the sub-differential mapping is a diagonal operator, this is again a monotone mapping. This remark proves that the asynchronous parallel methods converge also in the context of the red-black ordering of the subdomains.

Remark 8. For the solution of system (10), we can consider instead of the classical parallel asynchronous Schwarz alternating method, the parallel asynchronous Schwarz alternating method with flexible communications which corresponds to a more general model of parallel asynchronous iterations. This kind of method consists in using partial ordering techniques linked with the discrete maximum principle [21]. In the present context, the values of the

components of the iterate vector generated by the other process, can be accessed while the computations are still in progress. In such method the current value of any component of the iterate vector is not necessarily labelled by an outer iteration number as communication may occur at any time. Then, in this class of methods, partial updates can be used at any time in the computation. Thus, flexible data exchanges among processors are allowed; as a consequence, the coupling between communications and computations can be improved. This method consists in starting the iteration (13) with an initial guess U^0 such that

$$A \cdot U^0 + \Lambda(U^0) - G \geq 0, \quad \Lambda(U^0) \in \partial \bar{\Psi}_K(\mathcal{U}^0). \quad (14)$$

In practice U^0 is chosen such that $U^0 = \Phi$. In this case, for $r \geq 1$, the vector $\mathcal{W} = \{\mathcal{W}_1, \mathcal{W}_2, \dots, \mathcal{W}_x\}$ belongs to the order interval $(\mathcal{U}^r, \min(\mathcal{U}^{p(r)}, \mathcal{U}^q))$, where $\mathcal{U}^{p(r)}$ denotes the vector with block components $\mathcal{U}_j^{p_j(r)}$, $j \in \{1, 2, \dots, \alpha\}$ and $q = \max_{k \in K_{s(r)}^r}(k)$, where the set K_i^r contains all the iteration numbers lower than r associated with the computation of the i^{th} block component (see [21]). Note that the values of the components \mathcal{W}_j are also the available values of the iterate vector. In this case, if (14) is satisfied, it can be proved (see [21]) that the sequence of iterate vectors satisfies the maximum discrete principle; indeed $(U^r)_{r \geq 0}$ satisfies the following inequalities $U \leq \dots \leq U^r \leq \dots \leq U^0$ (see also [29] for an analysis by using contraction technics).

In conclusion for the first way of projection, since the discretization scheme applied to the numerical approximation of the boundary value problem modelling the obstacle problem, leads to the solution of an algebraic system constituted by an M-matrix perturbed by a monotone diagonal operator, then, the parallel synchronous and asynchronous alternating methods with or without flexible communications converge to the solution of the considered discretized boundary value problem for any initial guess U^0 and for any ordering of the subdomains. From a practical point of view, concerning the implementation associated to the model problem, we have:

1. first to compute an intermediate value of a component of the iterate vector by solving for example, one equation of the algebraic linear system,
2. and then to project this intermediate value of the component on the convex set; more precisely, if the constraint is satisfied, the intermediate value of the component is not changed and if the constraint is saturated, for example, if the component of U is less than the component of Φ ($U < \Phi$), then the intermediate value of the component is equal to Φ .

Linearization of the problem by the Howard method

Note also that the projection on the discretized convex set can be performed by using the Howard method; such mathematical approach avoids using the notion of sub-differential mapping. In fact this second method is a Newton like method allowing the linearization of the stationary problem associated with (3). Indeed, after appropriate discretization we have to solve at each time step $\max(AV - G, V - \Phi) = 0$

and we associate it with the linearized system

$$B(V) \cdot V = C(V), \quad (15)$$

constructed as follows:

1. if the i -th component of $(AV - G)$ is greater than the i -th component of $(V - \Phi)$, then the i -th line of $B(V)$ is the i -th line of A and the i -th component of $C(V)$ is the i -th component of G ,

2. else all the entries of the i -th line of $B(V)$ are null except the diagonal entry equal to one, and the i -th component of $C(V)$ is the i -th component of Φ .

Since the discretization matrix A is strictly (and irreducibly) diagonally dominant and regardless the sign of the entries of the matrix A , obviously the matrix $B(V)$ is strictly diagonally dominant. Note that, the matrix $B(V)$ is not necessarily irreducible, since in the linearization process some entries equal to zero can appear in all the off-diagonal entries of a same line; this happens when the previous case 2) occurs during the building of the linearized system. Nevertheless, at each step of the linearization process, the global matrix $B(V)$ has strictly positive diagonal entries and nonpositive off-diagonal entries; furthermore $B(V)$ is strictly diagonally dominant. Consequently, $B(V)$ is an M-matrix and at each step of the Howard method we have to solve a linear algebraic system where the matrix to be inverted is an M-matrix. So, for solving such linear system, we can use the parallel synchronous or asynchronous Schwarz alternating method described in subSection 'The Schwarz alternating method', and, due to the property of $B(V)$, this last algorithm converges for each step of the linearization process. We refer to [24] for a direct convergence analysis of the parallel asynchronous Howard algorithm by partial ordering analysis, including parallel asynchronous method with flexible communications.

Remark 9. It can be noted that the Howard method can be used for a more general class of complementary problems. Indeed consider the Hamilton–Jacobi–Bellman equation arising for example in image processing. Such stationary problem can be formulated as follows

$$\begin{cases} \text{Find } v \in E \text{ such that} \\ \sup(\bar{A}_1 v - f_1, \bar{A}_2 v - f_2) = 0, & \text{everywhere in } \Omega, \\ v = 0, & \text{everywhere on } \partial\Omega, \end{cases} \quad (16)$$

where \bar{A}_i , $i = 1, 2$ are two elliptic operators; after appropriate discretization we have to solve the discrete complementary problem $\max(A_1 V - F_1, A_2 V - F_2) = 0$,

$$(17)$$

where A_i , $i = 1, 2$ are the discretization matrices. Then, for the previous problem, we can consider an associated linearized problem obtained by the Howard method. The linearization is the same as the one considered in (15), except the identity matrix is replaced by the matrix A_2 and Φ is replaced by F_2 in the second argument. Now, if A_i , $i = 1, 2$ are two M-matrices, the matrix $B(V)$ arising in the linearized system is also an M-matrix and consequently, the parallel synchronous or asynchronous Schwarz alternating methods converge.

In conclusion, in the proposed study, we present two distinct ways for achieving the projection on the convex set. The first way consists classically to perturb the linear algebraic systems by adding the sub-differential of the indicator function. Then we obtain a multivalued formulation of the problem to solve. This way corresponds to a theoretical way and allows to analyze the convergence of the parallel synchronous and asynchronous subdomain methods. In the implementation of the algorithm, we have to project the intermediate value of the iterate vector on the convex set as explained below.

The second way is distinct to the first one and is more global. Indeed we have to linearize the problem by the Howard process. Then we obtain a linearized system (15), and the convergence of the parallel synchronous and asynchronous subdomain methods follows from the fact that the matrix $B(V)$ issued from the Howard

process is an M-matrix. Then, the proposed paper presents two methods for the solution of the obstacle problem.

Parallel simulation

This section is devoted to the presentation of the parallel algorithms implementation and the results of parallel experiments.

Principle of implementation

The simulation algorithm is based on master/slave paradigm which is a commonly used approach for parallel and distributed applications. In our case, a master process controls the distribution of work to a set of slave processes.

Algorithm 1. General algorithm

Algorithm

```

if (master)
  compute discretization matrix and right hand side
  send input data to slaves
  compute SPMD parallel algorithm to solve the algebraic
  system
  receive output data from slaves
else
  receive input data from master
  compute SPMD parallel algorithm to solve the algebraic
  system
  send output data to master
end if

```

This process is used for the solution of the system that arises in the numerical simulation of the obstacle problem. Matrix and right hand side creation have been implemented sequentially since this part of computation is not very intensive. In this way, the master process can be seen as a matrix and vector filler that feeds a parallel solver. In other words, only intensive computations have been parallelized with MPI facilities.¹ Note that only the parallel synchronous and classical asynchronous Schwarz alternating methods have been implemented; the asynchronous Schwarz alternating method with flexible communications has not been considered in the present study. In addition, the principle of parallel asynchronous iterative subdomain algorithms implementation for on one hand the Schwarz method for the multivalued problem and on the other hand for the Schwarz method with Howard linearization can be summarized as follows:

Algorithm 2. Schwarz for multivalued problem

```

do until global convergence
  for each subdomain assigned to the processor do
    if local convergence is not reached then
      receive the latest boundary values
      solve with projection the subsystems
      send the boundary values to the neighbors
    end if
  end for
end do

```

Algorithm 3. Schwarz with Howard linearization

```

do until global convergence
  for each subdomain assigned to the processor do
    if local convergence is not reached then
      receive the latest boundary values
      solve the subsystems:
      – compute matrix and right hand side according to
        Section 'Linearization of the problem by the Howard
        method'
      – solve the linearized subsystem
      send the boundary values to the neighbors
    end if
  end for
end do

```

Subdomains are assigned to each processor following a red–black ordering. For the solution of the obstacle problem, note that in each subdomain we have considered either the implementation of the block Gauss–Seidel method with projection of the iterate vector on the convex set or the block Gauss–Seidel method applied to the solution of the linearized problem by the Howard process. For the first considered method, this kind of algorithm is faster than a point relaxation method with projection.

Parallel experiments

Parallel simulations for the solution of the obstacle problem have been performed on various distant clusters of the Grid'5000 platform. This French grid platform was composed of 2970 processors with a total 6906 cores distributed over 9 sites in France. In this network every site hosts clusters and all sites are connected by highspeed communication; nowadays the communication network is Gigabit Ethernet for local machines. Bandwidth among the different sites is about 10 Gbps. Most of Grid'5000 sites are composed of several clusters with different computing architectures and performances. Fig. 3 illustrates the sites of the Grid'5000 architecture.

The parallel experiments have been performed using an heterogeneous and dedicated machines running Linux 64 bits. The machines of the Grid'5000 platform used are located in Sophia-Antipolis, Rennes, Nancy and Toulouse sites. The characteristics of these machines are summarized in Table 1.

In the parallel experiments we have only considered the performances obtained by computing on one time step which in fact corresponds to the solution of a stationary obstacle problem. Indeed in a previous study [16,17] where a time-dependent obstacle problem has been solved by using projected Richardson method and subdomain without overlapping algorithm, we have shown that all time steps require similar execution time for the solution of each stationary problem. Thus, experiments with Schwarz alternating methods allow us to estimate the performances of the considered parallel algorithm for the solution of time dependent problem.

In the parallel experiments, we consider a benchmark where a regular mesh is defined on $\Omega = [0, 1] \times [0, 1] \times [0, 1]$; in numerical simulations, we have considered two meshes constituted on one hand of 64,000,000 mesh points² and on other hand of 32,768,000 mesh points³; due to utilization constraints of the grid platform in dedicated exploitation, the number of mesh points is limited to the previous values; for the same reason the number of cores is also

¹ Message Passing Interface.

² $400 \times 400 \times 400 = 64,000,000$ points.

³ $320 \times 320 \times 320 = 32,768,000$ points.

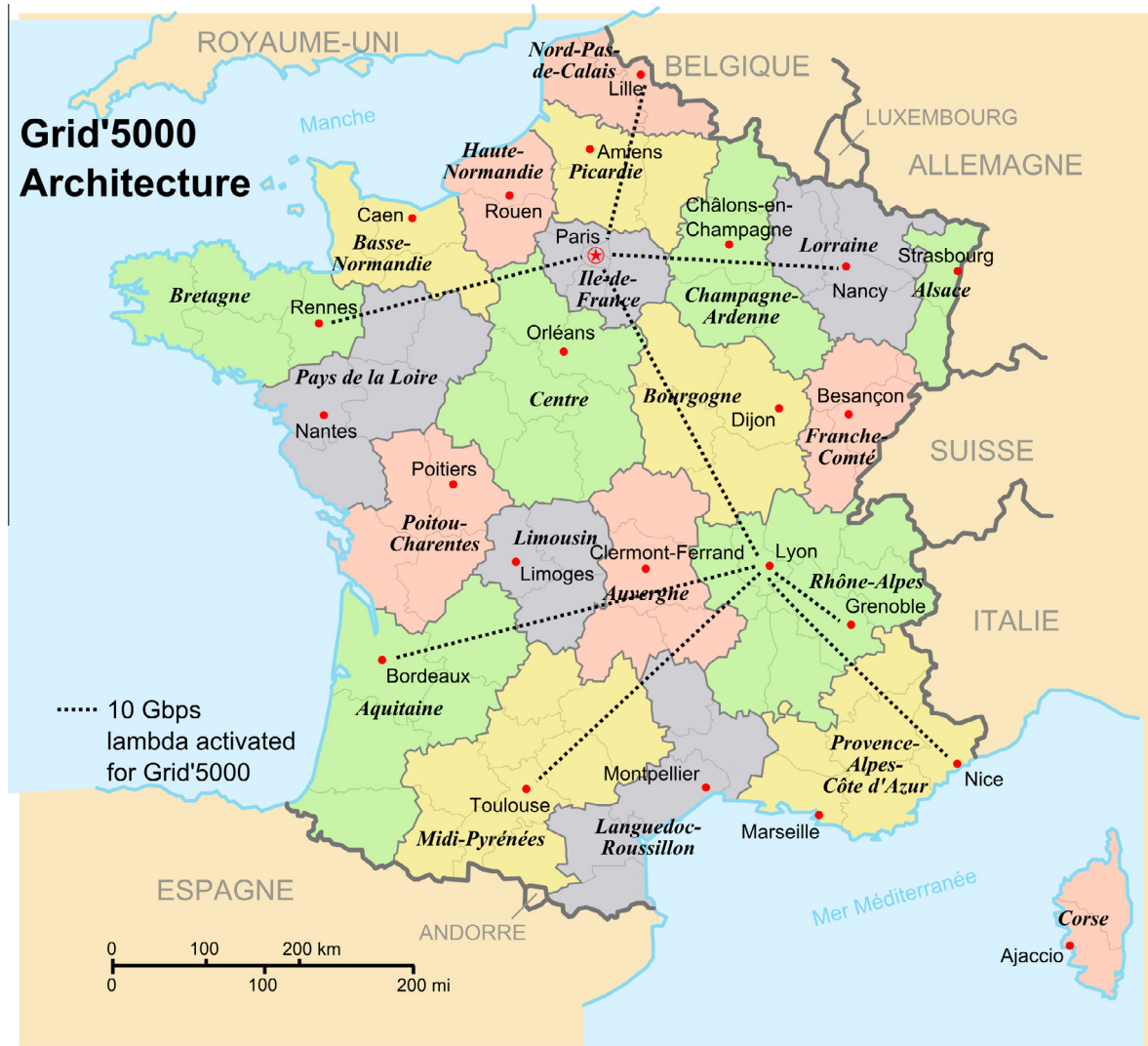


Fig. 3. The Grid'5000 architecture.

Table 1
Characteristics of machines on each site.

| Site | Cluster | Processors type | Speed (GHz) | cpu | Core | RAM size (GB) |
|----------|----------|-----------------------|-------------|-----|------|---------------|
| Rennes | Paradent | Intel Quad Xeon L5420 | 2.5 | 2 | 4 | 32 |
| Sophia | Suno | Intel Quad Xeon E5520 | 2.26 | 2 | 4 | 32 |
| Toulouse | Pastel | AMD Opteron 2218 | 2.6 | 2 | 2 | 8 |
| Nancy | Griffon | Intel Quad Xeon L5420 | 2.5 | 2 | 4 | 16 |

limited. The diffusion coefficient is set to $b = 0.2$ and the components of the convection vector are equal to -0.9 ; furthermore $c = 0$ and $\phi = 0$. The domain is decomposed into 256 overlapping subdomains numbered using a red-black ordering, well adapted to parallel computation. In the implementation of the Schwarz alternating methods, two or more subdomains are assigned to each processor; thus, on each processor, this assignation allows a behavior of the parallel algorithm as close as possible of a multiplicative Schwarz alternating method. From a practical point of view, parallel synchronous and asynchronous algorithms are more efficient on such subdomain distribution.

The numerical algorithms have been implemented using Fortran 90, BLAS subroutines, no other external libraries have been used; the remainder of the implementation is coded by ourself. In particular, a block relaxation method has been implemented for the solution of the obstacle problem. The parallelization is achieved using MPI facilities. Message passing are implemented with MPI persistent communication requests, in both synchronous and asynchronous cases. Thus, the proposed MPI implementation of the Schwarz alternating method in decentralized memory multiprocessor is more efficient. The implementations of message passing are described in Algorithms 4 and 5.

Algorithm 4. Implementation of synchronous message passing using MPI facilities.

```

Synchronous message passing:
!!!!!!!!!!!!!!!!!!!!!!
! SEND REQUESTS!
!!!!!!!!!!!!!!!!!!!!!!
! Wait for the completion of the former send
  requests
call MPI_WAITALL (mlpt%NSEND,sndt,starray,ierr)
! Pack the current messages
do i = 1, mlpt
  badr = mlpt%SND_BT (i)
  call MSGPK (PK_P,mlpt%SND_HD (i),
    handl%LOCFV,X,buf (badr))
end do
! Start sending messages
call MPI_WAITALL (mlpt%NRECV,rcvt,starray,ierr)
!!!!!!!!!!!!!!!!!!!!!!
! RECEIVE REQUESTS!
!!!!!!!!!!!!!!!!!!!!!!
! Wait for the completion of the former receive
  requests
call MPI_WAITALL (mlpt%NRECV,rcvt,starray,ierr)
! Unpack the current messages
do i = 1, mlpt%NRECV
  badr = mlpt%RCV_BT (i)
  call MSGPK (PK_U,mlpt%RCV_HD
    (i),handl%LOCFV,X,buf (badr))
end do
! Start receiving messages
call MPI_STARTALL (mlpt%NRECV,rcvt,ierr)

```

Algorithm 5. Implementation of asynchronous message passing using MPI facilities.

```

Asynchronous message passing:
!!!!!!!!!!!!!!!!!!!!!!
! SEND REQUESTS!
!!!!!!!!!!!!!!!!!!!!!!
! Nonblocking check for the completion
! of the former send requests
call MPI_TESTSOME
  (mlpt%NSEND,sndt,nout,outarray,starray,ierr)
do i = 1, nout
  ! Pack the current messages for completed
    requests only
  mpos = outarray (i)
  badr = mlpt%SND_BT (mpos)
  call MSGPK (PK_P,mlpt%SND_HD
    (mpos),handl%LOCFV,X,buf (badr))
  ! Start sending messages
  call MPI_START (sndt (mpos),ierr)
end do
!!!!!!!!!!!!!!!!!!!!!!
! RECEIVE REQUESTS!
!!!!!!!!!!!!!!!!!!!!!!
! Nonblocking check for the completion
! of the former receive requests
call MPI_TESTSOME
  (mlpt%NRECV,rcvt,nout,outarray,starray,ierr)
do i = 1, nout
  ! Unpack the current messages for completed

```

```

  requests only
  mpos = outarray (i)
  badr = mlpt%RCV_BT (mpos)
  call MSGPK (PK_U,mlpt%RCV_HD
    (mpos),handl%LOCFV,X,buf (badr))
  ! Start receiving messages
  call MPI_START (rcvt (mpos),ierr)
end do

```

For all considered decomposition of the problem to solve, convergence of the proposed sequential and parallel algorithms is ensured. Concerning the rate of convergence, when the convergence analysis is done by contraction technics, an estimate of the rate of convergence is obtained by considering on one hand the spectral radius of the Jacobi matrix associated with the discretization matrix A and on the other hand, the associated subdomain decomposition (see [19]).

Many experiments have been performed to exhibit elapsed times of parallel Schwarz alternating method in both synchronous and asynchronous iterative schemes and using two or three distant sites. The latter case corresponds to an heterogeneous architecture since computing processors are different; indeed the machines located in Sophia-Antipolis and Nancy used Intel processors and the machines located in Toulouse used AMD processors.

In the parallel simulations only one core of each machine could be used. Indeed the amount of data to be stored in each RAM memory, for each MPI process, is very large; consequently very large memory size is needed, in fact greater than 4 GB. On other hand, the advantage of using 64 bits machines and compilers is the capability to use RAM with size greater than 4 GB. Thus, in the case where many cores per machine are used, due to memory swapping, the performances of parallel algorithms will decrease. Note that, due to the size of the algebraic systems to solve, only machines having large memory could be used.

The results of the experiments are summarized in Tables 2–7 and Figs. 4–9, for the obstacle problem formulated by a multi-valued algebraic system [30] and for the linearized and discretized obstacle problem, called in the sequel Schwarz–Howard method. In the following tables, P denotes the number of processors.

In fact, the two considered formulations of the discretized obstacle problem lead to similar performances of the associated

Table 2

Elapsed time (seconds) and number of relaxations for sequential algorithm with Schwarz alternating method for the multivalued problem on cluster suno (Sophia), paradent (Rennes), griffon (Nancy) and pastel (Toulouse).

| 32,768,000 mesh points | | | | | |
|------------------------|------|---------|----------|--------|------|
| suno | | griffon | | pastel | |
| 7494 | 2105 | 8748 | 2105 | 11,780 | 2105 |
| 64,000,000 mesh points | | | | | |
| suno | | | paradent | | |
| 16,099 | 2380 | | 21,214 | 2380 | |

Table 3

Elapsed time (seconds) and number of relaxations for sequential algorithm with Schwarz–Howard method on cluster suno (Sophia), paradent (Rennes), griffon (Nancy) and pastel (Toulouse).

| 32,768,000 mesh points | | | | | |
|------------------------|------|---------|----------|--------|------|
| suno | | griffon | | pastel | |
| 7269 | 2066 | 8915 | 2066 | 9611 | 2066 |
| 64,000,000 mesh points | | | | | |
| suno | | | paradent | | |
| 15,820 | 2337 | | 21,752 | 2337 | |

Table 4

Elapsed time (s), number of relaxations and communication time with Schwarz alternating method for the multivalued problem on cluster suno (Sophia) and paradis (Rennes) for 64,000,000 mesh points.

| <i>P</i> | Async. | Relaxations | | | Comm | Sync. | Relaxations | Comm | τ |
|----------|--------|-------------|------|---------|------------|--------|-------------|------------|--------|
| | | Min | Max | Average | | | | | |
| 4 | 8409 | 3617 | 4424 | 4014 | 521 (6.2%) | 11,300 | 2380 | 5255 (46%) | 1.34 |
| 8 | 3977 | 3234 | 4290 | 3729 | 249 (6.3%) | 6033 | 2380 | 2411 (40%) | 1.51 |
| 16 | 1709 | 2334 | 3963 | 3269 | 68 (3.9%) | 2625 | 2380 | 830 (32%) | 1.53 |
| 32 | 934 | 2208 | 4317 | 3355 | 60 (6.4%) | 1599 | 2380 | 397 (25%) | 1.67 |
| 64 | 428 | 1310 | 4621 | 2876 | 37 (8.6%) | 1120 | 2380 | 261 (13%) | 2.61 |

Table 5

Elapsed time (s), number of relaxations and communication time with Schwarz–Howard method on cluster suno (Sophia) and paradis (Rennes) for 64,000,000 mesh points.

| <i>P</i> | Async. | Relaxations | | | Comm | Sync. | Relaxations | Comm | τ |
|----------|--------|-------------|------|---------|------------|--------|-------------|------------|--------|
| | | Min | Max | Average | | | | | |
| 4 | 8254 | 3341 | 4132 | 3741 | 628 (7.6%) | 11,447 | 2337 | 3880 (34%) | 1.39 |
| 8 | 4141 | 2910 | 4362 | 3694 | 329 (7.9%) | 6098 | 2337 | 2778 (45%) | 1.47 |
| 16 | 1818 | 2474 | 4098 | 3351 | 72 (3.9%) | 2705 | 2337 | 872 (32%) | 1.49 |
| 32 | 956 | 2313 | 4328 | 3326 | 58 (6.0%) | 1607 | 2337 | 398 (24%) | 1.68 |
| 64 | 414 | 1720 | 4021 | 2928 | 28 (6.7%) | 1162 | 2337 | 281 (24%) | 2.80 |

Table 6

Elapsed time (s), number of relaxations and communication time with Schwarz alternating method for the multivalued problem on cluster suno (Sophia), griffon (Nancy) and pastel (Toulouse) for 32,768,000 mesh points.

| <i>P</i> | Async. | Relaxations | | | Comm | Sync. | Relaxations | Comm | τ |
|----------|--------|-------------|------|---------|------------|-------|-------------|------------|--------|
| | | Min | Max | Average | | | | | |
| 4 | 3198 | 2885 | 3443 | 3041 | 46 (1.4%) | 5257 | 2105 | 2284 (43%) | 1.64 |
| 8 | 1822 | 2971 | 3405 | 3201 | 78 (4.0%) | 2494 | 2105 | 873 (34%) | 1.37 |
| 16 | 773 | 2176 | 3424 | 2802 | 25 (3.2%) | 1156 | 2105 | 278 (26%) | 1.50 |
| 32 | 442 | 2059 | 3712 | 2931 | 33 (7.4%) | 958 | 2105 | 239 (24%) | 2.17 |
| 64 | 202 | 1139 | 4265 | 2594 | 22 (11.0%) | 815 | 2105 | 189 (23%) | 4.03 |
| 128 | 57 | 683 | 3500 | 2267 | 8 (14.0%) | 549 | 2105 | 133 (24%) | 9.63 |

Table 7

Elapsed time (s), number of relaxations and communication time with Schwarz–Howard method on cluster suno (Sophia), griffon (Nancy) and pastel (Toulouse) for 32,768,000 mesh points.

| <i>P</i> | Async. | Relaxations | | | Comm | Sync. | Relaxations | Comm | τ |
|----------|--------|-------------|------|---------|------------|-------|-------------|-----------|--------|
| | | Min | Max | Average | | | | | |
| 4 | 3283 | 2702 | 3497 | 3010 | 61 (1.9%) | 5323 | 2066 | 2282(43%) | 1.62 |
| 8 | 1830 | 2893 | 3483 | 3099 | 73 (3.9%) | 2697 | 2066 | 955 (35%) | 1.47 |
| 16 | 833 | 2158 | 3585 | 2860 | 29 (3.4%) | 1164 | 2066 | 280 (24%) | 1.40 |
| 32 | 443 | 2058 | 3761 | 2908 | 29 (6.5%) | 913 | 2066 | 224 (24%) | 2.06 |
| 64 | 211 | 1359 | 4479 | 2625 | 23 (10.9%) | 788 | 2066 | 184 (23%) | 3.73 |
| 128 | 58 | 718 | 3360 | 2131 | 8 (13.8%) | 493 | 2066 | 122 (25%) | 8.5 |

parallel computational methods. Nevertheless, considering the more general Hamilton–Jacobi–Bellman problem (17), the multivalued formulation is not relevant, while the linearized formulation by the Howard method is interesting.

Tables 2 and 3 display the sequential elapsed times obtained on each cluster. The number of machines used has been limited up to 64 or 128 according to the duration of the exploitation of the computational codes. Moreover, for the considered size of the model problem, since the overhead damages the performances of the parallel algorithms, additional machines are not necessary particularly when synchronous scheme are tested; indeed more processors are not really interesting and the performances will not be improved.

In these tables and figures, comparison of elapsed time for parallel synchronous and asynchronous methods are presented; since the notion of speed-up and efficiency are not relevant when distant and heterogeneous clusters are used, the parameter τ measures the ratio of elapsed time between synchronous and asynchronous

methods; this parameter is more appropriate for the comparison between the previous two kinds of parallel methods.

Figs. 4 and 5 (Figs. 6 and 7, respectively) present the elapsed time obtained when the mesh is constituted by 64,000,000 (32,768,000, respectively) mesh points and when 2 (3, respectively) sites are used. Figs. 8 and 9 show the evolution of the parameter τ when the number of processors increases and when 2 or 3 sites are used.

Tables 4–7, Figs. 4–7 show clearly the benefit of using parallel algorithms. Note that the asynchronous parallel iterative algorithms are more efficient than the synchronous ones; so the advantage of the first method compared to the second one is clearly showed.

In Tables 4–7, it can be noted, that in asynchronous parallel experiments, the number of relaxations necessary to reach convergence is greater when few processors are used than the one necessary when parallel synchronous schemes are used. Furthermore,

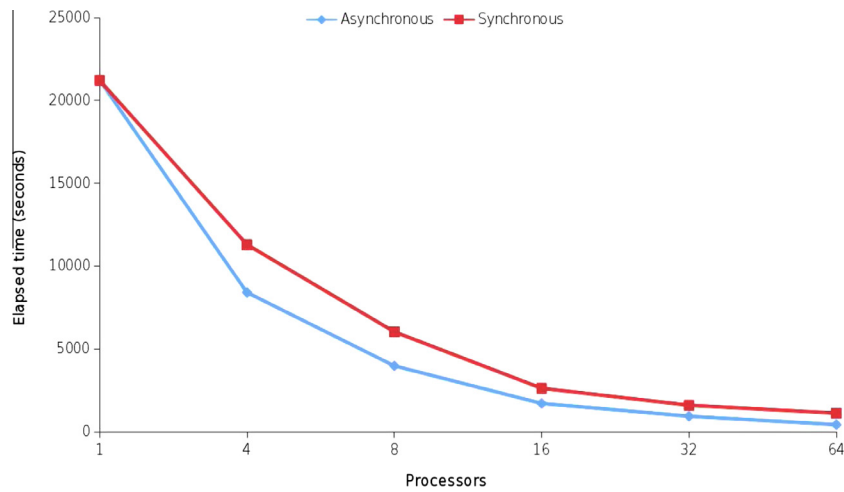


Fig. 4. Elapsed time (s) with Schwarz alternating method for the multivalued problem on cluster suno (Sophia) and parudent (Rennes) for 64,000,000 mesh points.

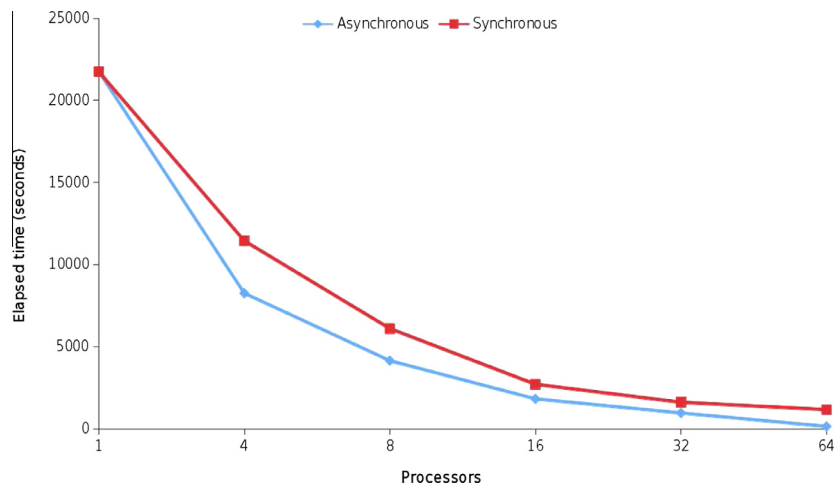


Fig. 5. Elapsed time (s) with Schwarz-Howard method on cluster suno (Sophia) and parudent (Rennes) for 64,000,000 mesh points.

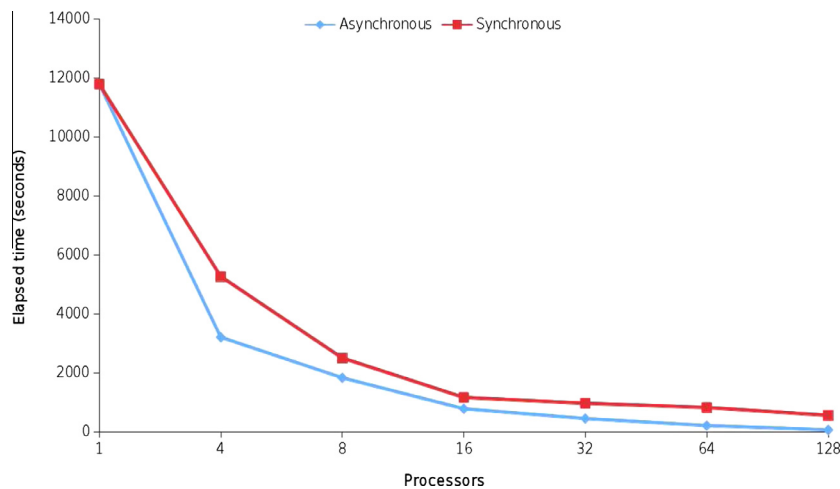


Fig. 6. Elapsed time (s) with Schwarz alternating method for the multivalued problem on cluster suno (Sophia), griffon (Nancy) and pastel (Toulouse) for 32,768,000 mesh points.

despite higher number of relaxations, elapsed time of asynchronous scheme is less than the one obtained when synchronous scheme is used. This asynchronism is an efficient way to deal with

communication overhead and load unbalance. It turns out that the overhead generated by additional relaxations in the case of asynchronous algorithms is smaller than the synchronization overhead

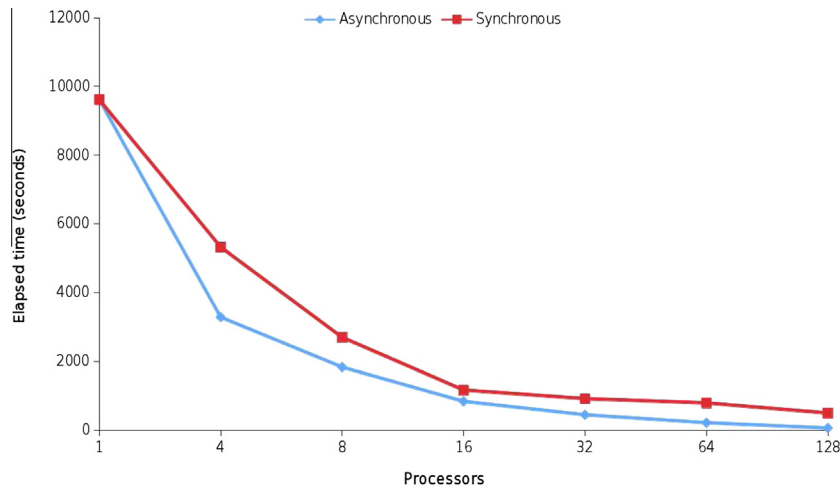


Fig. 7. Elapsed time (s) with Schwarz–Howard method on cluster suno (Sophia), griffon (Nancy) and pastel (Toulouse) for 32,768,000 mesh points.

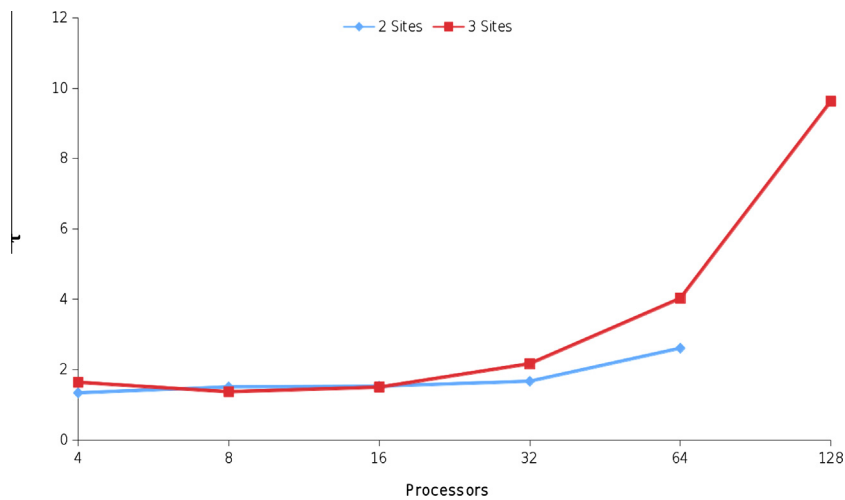


Fig. 8. Comparison of the values of the parameter τ with Schwarz alternating method for the multivalued problem when the number of processors increases and when 2 and 3 sites are used.

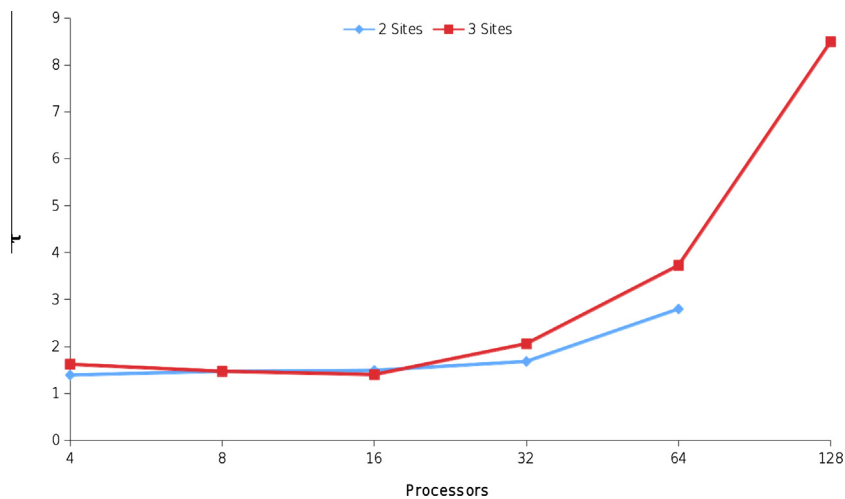


Fig. 9. Comparison of the values of the parameter τ with Schwarz–Howard method when the number of processors increases and when 2 and 3 sites are used.

combined with processor idle time of parallel synchronous schemes of computation. Nevertheless these additional relaxations improve the solution's accuracy and the numerical quality of the

computations. Moreover, the efficiency of the synchronous algorithm decreases faster than the efficiency of asynchronous algorithm when the number of processors increases. The lack of

synchronization points and the use of current values of the components of the iterate vector induce better performances for the asynchronous Schwarz alternating methods when several processors are used.

We focus now on experiments performed for the same problem with the same parameters on two and three distant sites.

Results are summarized in Tables 4–7. Many interesting issues can be pointed out from these tables:

1. it can be noticed that with both Schwarz alternating methods, we obtain good results; indeed this is due to the fact that the discretization matrix is nonsymmetric and it is well known that the block relaxation method is well adapted to this case,
2. the results on two or three distant sites show that the synchronous version of both Schwarz alternating methods is slower than the asynchronous one. The ratio τ obviously highlights the power of the asynchronous version compared to the synchronous one.

In Tables 4–7, note that another feature of the parallel experiments consists in the fact that the parameter τ is increasing with the increasing number of machines used (see Figs. 8 and 9). This means again that the efficiency of asynchronous schemes is clearly better than the efficiency of the synchronous one. Then clearly, asynchronous algorithms are well-adapted to the use of distant and heterogeneous machines.

Finally, note that the use of asynchronous parallel schemes is very interesting, since the weight of synchronizations is low and, as a consequence, the obtained elapsed times are shorter. Indeed, since non-blocking MPI communication subroutines are used in asynchronous algorithm, the low values of communication ratio in this case can only illustrate the absence of idle time. On the other hand, a blocking communication subroutine is used in the synchronous algorithm; so, the communication time measurements represent the idle times due to synchronizations and the amount of CPU operation dedicated to communications. Moreover Tables 4–7 and Figs. 4–7 show clearly the impact of the communications on the behavior of the iterative parallel algorithm and the fact that such communications degrade the performances of the synchronous methods. Due to the weight of synchronizations, parallel asynchronous experiments on the grid platform show clearly the interest of such methods on distant clusters. So parallel asynchronous iterative algorithms, more particularly on grid platform, are appropriate ways to solve complex problems.

Conclusion

In this study we have presented the parallel numerical solution of the stationary obstacle problem arising for example in a model problem of civil-engineering, solved by the parallel synchronous or asynchronous Schwarz alternating method. Due to the size of the algebraic systems to be solved and also to the fact that such system is very ill-conditioned and finally that the computational time is very long, this kind of problem is very difficult to solve by sequential algorithms. For the numerical solution of the considered problem, the stability of the time marching scheme is classically ensured; the convergence of the implemented methods has been ensured by appropriate spatial discretization scheme. Experiments with the Grid'5000 architecture allow us to compare the behavior of both synchronous and asynchronous schemes of computation in various configurations with geographically distant and heterogeneous sites. We have obtained very good performances of the parallel algorithms, particularly for the asynchronous ones. Due to the weight of synchronizations, parallel asynchronous experiments on the grid platform show clearly the interest of such

method on distant sites. So parallel asynchronous iterative algorithms, more particularly on grid platform, are appropriate ways to solve this complex problem. In conclusion, it appears from our experience that parallel asynchronous algorithms are robust, efficient methods of computation and do not need load balancing between the processors.

Acknowledgement

This study has been made possible with the support of Grid'5000 platform.

References

- [1] Glowinski R, Lions JL, Tremolieres R. Analyse numérique des inéquations variationnelles. Dunod, tome 1 and 2; 1976.
- [2] Ciarlet PG. Numerical analysis of the finite element method. Presses de l'université de Montréal; 1975.
- [3] Lions P, Mercier P. Approximation numérique des équations de Hamilton–Jacobi–Bellman. RAIRO Anal Numér 1980;14:369–93.
- [4] Tai XC. Rate of convergence for some constraint decomposition methods for nonlinear variational inequalities. Numer Math 2003;93:755–86.
- [5] Badea L, Tai XC, Wang J. Convergence rate analysis of a multiplicative Schwarz method for variational inequalities. SIAM J Numer Anal 2004;41–3:1052–73.
- [6] Badea L, Wang J. An additive Schwarz method for variational inequalities. Math Comput 2000;69:1341–54.
- [7] Tai XC. Convergence rate analysis of domain decomposition methods for obstacle problems. East-West J Numer Anal 2001;9(3):233–52.
- [8] Kuznetsov YA, Neittaanmaki P, Tarvainen P. Block relaxation methods for algebraic obstacle problems with M-matrices. East-West J Numer Math 1996;4:69–82.
- [9] Kuznetsov YA, Neittaanmaki P, Tarvainen P. Schwarz methods for obstacle problems with convection–diffusion operators. In: Keyes DE, Xu JC, editors. Proceedings of domain decomposition methods in scientific and engineering computing, AMS; 1995. p. 251–6.
- [10] Li C, Zeng J, Zhou S. Convergence analysis of generalized Schwarz algorithms for solving obstacle problems with T-monotone operator. Comput Math 2004;48:373–86.
- [11] Tai XC, Tseng P. Convergence rate analysis of an asynchronous space decomposition method for convex minimization. Math Comput 2002;71–239:1105–35.
- [12] Bolze R, Cappello F, Caron E, Daydé M, Desprez F, Jeannot E, et al. Grid'5000: a large scale and highly reconfigurable experimental grid testbed. Int J High Perform Comput Appl 2006;20(4):481–94.
- [13] Baudet G. Asynchronous iterative methods for multiprocessors. J Assoc Comput Mach 1978;25:226–44.
- [14] Chau M, Couturier R, Bahi J, Spiteri P. Parallel solution of the obstacle problem in Grid environment. High Perform Comput Appl 2011;25(4):488–95.
- [15] Spiteri P, Chau M. Parallel asynchronous Richardson method for the solution of obstacle problem. In: Almhana JN, Bhavsar VC, editors. Proceedings of the 16th annual international symposium on HPCS, 2002 IEEE; 2002. p. 133–8.
- [16] Chau M, Couturier R, Bahi J, Spiteri P. Parallel solution of the sequence of obstacle problems in a grid environment. In: Topping BHV, Iványi P, editors. Proceedings of the second international conference on parallel, distributed, grid and cloud computing for engineering, Stirlingshire, Scotland: Civil-Comp. Press; 2011.
- [17] Chau M, Couturier R, Bahi J, Spiteri P. Asynchronous grid computation for american options derivatives. Adv Eng Softw 2013;136–44.
- [18] Miellou JC, Spiteri P. Two criteria for the convergence of asynchronous iterations. In: Chenin P et al., editors. Computers and computing. Paris: Wiley Masson; 1985. p. 91–5.
- [19] Miellou JC, Spiteri P. Un critère de convergence pour des méthodes générales de point fixe. M2 AN 1985;19(4):645–69.
- [20] Miellou JC, Spiteri P. Itérations asynchrones avec segments d'ordre. Méthodes numériques performantes, Contrat ATP – CNRS n 8096; 1984.
- [21] Miellou JC, ElBaz D, Spiteri P. A new class of asynchronous iterative algorithms with order interval. Math Comput 1998;67(221):237–55.
- [22] Ortega JM, Rheinboldt WC. Iterative solution of nonlinear equations in several variables. New York: Academic Press; 1970.
- [23] Barbu V. Nonlinear semigroups and differential equations in Banach spaces. Noordhoff International Publishing; 1976.
- [24] Spiteri P, Miellou JC, El Baz D. Asynchronous Schwarz alternating methods with flexible communication for obstacle problem. Réseaux Syst Répartis 2001;13(1):47–66.
- [25] Giraud L, Spiteri P. Résolution parallèle de problèmes aux limites non linéaires. M2 AN 1991;25:73–100.
- [26] Lions JL, Stampacchia G. Variational inequalities. Commun Pure Appl Math 1967;XX:493–519.
- [27] Hoffman KH, Zou J. Parallel efficiency of domain decomposition methods. Parallel Comput 1993;19:1375–91.
- [28] Evans DJ, Deren W. An asynchronous parallel algorithm for solving a class of nonlinear simultaneous equations. Parallel Comput 1991;17:165–80.

- [29] El Baz D, Frommer A, Spitéri P. Asynchronous iterations with flexible communication: contracting operators. *J Comput Appl Math* 2005;176:91–103.
- [30] Chau M, Garcia T, Spiteri P. Parallel asynchronous Schwarz alternating method for the obstacle problem on grid computing. In: Topping BHV, Tsompanakis Y, editors. Proceedings of the thirteenth international conference on civil, structural and environmental engineering computing, Chania, Crete, Greece, Civil-Comp Press, Stirlingshire, UK, paper 321; 2011.