Optimisation parallèle et mathématiques financières

Pierre Spiteri[1]

IRIT – ENSEEIHT, UMR CNRS 5505 – 2 rue Charles Camichel, B.P. 7122
F-31 071 Toulouse, France
Pierre.Spiteri@enseeiht.fr

Résumé. Pour un problème issu des mathématiques financières, on établit un lien entre la formulation de ce problème et un problème de minimisation sur un espace approprié en fonction de la nature économique de l'application. L'expression des conditions d'optimalité conduit alors à la résolution d'équations ou d'inéquations aux dérivées partielles qu'on résout numériquement par des algorithmes parallèles asynchrones. Après avoir analysé la convergence des algorithmes parallèles asynchrones, on compare les résultats de ces derniers aux méthodes synchrones sur diverses architectures.

Mots clés: Options américaines et européennes, optimisation convexe, équations et inéquations aux dérivées partielles, problèmes complémentaires, méthode du gradient projeté, algorithmes parallèles synchrones et asynchrones.

1 Introduction et modèle mathématique

Le modèle mathématique étudié intervient dans de nombreuses applications mécaniques ou économiques. Compte tenu de la diversité des applications on a l'habitude de nommer ce problème, problème de l'obstacle.

Pour fixer la problématique, exposons une situation intervenant en mécanique des structures, qu'on peut comprendre intuitivement [3]. On considère, par exemple, une structure occupant un domaine fermé borné Ω, soumise à un certain nombre de contraintes notées f, correspondant à diverses forces appliquées sur cette structure. On souhaite alors connaître le déplacement noté u de la structure. On peut envisager deux cas distincts

- le déplacement n'est soumis à aucune contrainte, auquel cas le modèle mathématique décrivant cette étude est une équation aux dérivées partielles, en général linéaire, qui est soit stationnaire auquel cas l'opérateur est elliptique, ou qui dépend de la variable temps, et on doit résoudre un problème d'évolution,
- le déplacement est soumis à une contrainte, par exemple, on suppose que $u \leq \psi$, où ψ est une quantité connue ; dans ce cas le modèle mathématique décrivant cette étude est une inéquation aux dérivées partielles, appelée aussi par les mathématiciens inéquation variationnelle, problème fortement non linéaire, qui est soit stationnaire auquel cas l'opérateur associé est elliptique, ou qui dépend de la variable

[1] Cette étude a bénéficié du soutien du CNRS dans le cadre du projet ANR-07-CIS7-011-03.

supplémentaire temps, auquel cas l'opérateur associé est parabolique où hyperbolique du second ordre.

En fait qu'il s'agisse d'un problème avec ou sans contrainte, la formulation découle de celle d'un problème d'optimisation. Dans le cas le plus simple du problème stationnaire, considérons la fonctionnelle à minimiser suivante

$$J(v) = \frac{1}{2} a(v,v) - L(v), \ \forall v \in V \ ; \tag{1}$$

lorsque la structure est homogène, la forme bilinéaire a(u,v) et de la forme linéaire L(v) sont définies comme suit

$$a(u,v) = \int_\Omega (\nabla u.\nabla v + \theta.u.v).dx \ \text{ et } \ L(v) = \int_\Omega f.v.dx, \ \theta \geq 0 ; \tag{2}$$

V est un espace fonctionnel choisi en fonction des conditions aux limites ainsi que de la nature du problème ; en pratique V est soit un espace vectoriel normé complet dans le cas de l'optimisation sans contrainte soit un ensemble convexe fermé dans le cas de l'optimisation avec contrainte. Une position d'équilibre correspond au problème de minimisation suivant

$$\begin{cases} \text{Déterminer } u \in V \text{ tel que} \\ J(u) \leq J(v), \ \forall v \in V \end{cases} . \tag{3}$$

Dans le cas sans contrainte, lorsque la structure admet un déplacement nul le long de la frontière $\partial\Omega$, l'espace V est l'espace $H_1^0(\Omega)$ des fonctions de carré sommable dont les dérivées (en toute rigueur au sens des distributions) sont de carré sommable et nulles au bord. Sous des hypothèses convenables, il est facile de vérifier que la condition d'Euler qui traduit la nullité de la dérivée de la fonctionnelle J, s'écrit

$$\langle J'(u), v \rangle = a(u,v) - L(v) = 0, \ \forall \ v \in H_1^0(\Omega), \tag{4}$$

expression qui conduit moyennant l'utilisation de la formule de Green, à la résolution de l'équation aux dérivées partielles suivante

$$\begin{cases} -\Delta u + \theta.u = f, \text{ presque partout dans } \Omega \\ u = 0, \text{ presque partout sur } \partial\Omega \end{cases} ; \tag{5}$$

le problème (5) correspond donc à la résolution d'un problème de Poisson stationnaire, avec conditions aux limites de Dirichlet homogènes. Toujours dans le cas sans contrainte, si de plus $\theta > 0$, et lorsque la dérivée normale du déplacement est nulle le long de la frontière $\partial\Omega$, l'espace V est l'espace $H_1(\Omega)$ des fonctions de carré sommable dont les dérivées sont de carré sommable ; dans ce cas, la condition d'Euler, appliquée dans les mêmes conditions que précédemment conduit à la résolution de l'équation aux dérivées partielles stationnaire suivante

$$\begin{cases} -\Delta u + \theta.u = f, \text{ presque partout dans } \Omega \\ \dfrac{\partial u}{\partial n} = \vec{\nabla} u.\vec{n} = 0, \text{ presque partout sur } \partial\Omega \end{cases} , \tag{6}$$

où n est la normale orientée vers l'extérieur au domaine Ω ; le problème (6) correspond à la résolution d'un problème de Poisson, avec conditions aux limites de

Neumann homogènes. De la même façon, lorsque la structure est soumise à d'autres conditions aux limites classiques, le problème de minimisation se traduit par la résolution d'un problème de Poisson, avec conditions aux limites appropriées, moyennant la définition correcte de l'espace V et celle de la forme bilinéaire a(u,v) et de la forme linéaire L(v).

Si on considère à présent le cas avec contrainte, on doit considérer le problème de minimisation, non pas sur un espace vectoriel normé complet V, mais sur un ensemble convexe fermé $\mathcal{C}$, défini par

$$\mathcal{C} = \left\{ v \in V | v \leq \psi, \text{ presque partout sur } \Omega \right\}, \tag{7}$$

V étant choisi comme précédemment en fonction des conditions aux limites ; l'expression de la condition d'optimalité, c'est à dire ici de l'inéquation d'Euler, s'exprime classiquement par

$$<J'(u),v> = a(u,v-u) - L(v-u) \geq 0, \forall\, v \in \mathcal{C} ; \tag{8}$$

en considérant plusieurs choix distincts de la fonction test v, et si de plus V est l'espace $H_1^0(\Omega)$, on montre classiquement que l'inéquation d'Euler s'exprime par

$$\begin{cases} -\Delta u + \theta.u - f \leq 0 \text{ et } u \leq \psi, \text{ presque partout sur } \Omega, \\ (-\Delta u + \theta.u - f).(u - \psi) = 0, \text{presque partout sur } \Omega, \\ u = 0, \text{presque partout sur } \partial\Omega, \end{cases} \tag{9}$$

qui correspond à la résolution d'une inéquation aux dérivées partielles stationnaire.

Il est à noter que, dans le cas avec ou sans contrainte, la fonctionnelle J(v) est strictement convexe et de plus $\lim_{v \to \infty} J(v) = +\infty$; cette dernière propriété découle du fait que d'une part, la forme bilinéaire a(v,v) est continue et coercive et d'autre part que la forme linéaire est continue. Donc, dans les deux cas avec ou sans contrainte, le problème de minimisation admet une solution unique.

Ce type d'équations se retrouve également de manière analogue en mathématiques financières où on a à résoudre soit des équations soit des inéquations aux dérivées partielles [7]. Cependant l'analogie se borne uniquement à l'expression formelle des équations ou des inéquations aux dérivées partielles à résoudre. La modélisation du problème se conçoit de manière nettement moins intuitive que le problème précédent de mécanique. A ce stade, pour situer les choses, il est nécessaire d'introduire quelques éléments de terminologie. On appelle actif, une action, une obligation, une devise, un taux de change ou encore une matière première ; dans le jargon financier, on parle d'actif sous-jacent sur lequel porte l'option. Une option d'achat (ou call) donne à son détenteur le droit et non l'obligation d'acheter un actif financier (ou risqué) à une date future convenue d'avance, appelée échéance et notée T (ou date d'expiration qui limite la durée de vie de l'option) et à un prix fixé d'avance à la signature du contrat, conventionnellement appelé contrat interne. Une option de vente (ou put) donne à son détenteur le droit et non l'obligation de vendre un actif financier (ou risqué) à une date future convenue d'avance et à un prix fixé d'avance à la signature du contrat. Une option est également caractérisée par son montant, c'est-à-dire la quantité d'actif sous-jacent à acheter ou à vendre. Il y a deux

types d'option. Une option européenne est une option dont la date de maturité (c'est-à-dire la date future d'expiration) est fixée d'avance à la date d'échéance ; c'est-à-dire qu'il est interdit d'exercer le droit d'acheter ou de vendre avant la date d'échéance. Une option américaine est telle que le droit d'acheter ou de vendre l'actif considéré peut être exercé à n'importe quel moment avant la date d'échéance. Le prix d'exercice (ou strike) , est le prix fixé d'avance. L'option considérée (call ou put) a un prix appelé habituellement prime.

Pour fixer les idées, considérons le cas d'un call européen (option d'achat européen), d'échéance T, sur une action dont le cours à la date t, est donnée par S_t. Soit K le prix d'exercice. De deux choses l'une

- si, à l'échéance T, le prix K est supérieur au cours S_T, le détenteur de l'option n'a pas intérêt à exercer,
- si, par contre, $S_T > K$, l'exercice de l'option permet à son détenteur de réaliser un profit égal à $S_T - K$, en achetant l'action au prix K et en la revendant sur le marché au cours S_T .

On voit qu'à l'échéance la valeur du call est donnée par la quantité $Max(S_T - K,0)$. Pour le vendeur de l'option (appelé aussi trader), il s'agit, en cas d'exercice, d'être en mesure de fournir une action au prix K, et par conséquent de pouvoir produire à l'échéance une richesse égale à $Max(S_T - K,0)$. Au moment de la vente de l'option, qu'on choisira comme origine des temps, le cours S_T est inconnu et il se pose deux questions :

- combien faut-il payer à l'acheteur de l'option, autrement dit comment évaluer à l'instant t = 0 une richesse $Max(S_T - K,0)$ disponible à la date T ? C'est le problème du pricing,
- comment le vendeur qui touche la prime à l'instant t = 0, parviendra t'il à produire la richesse $Max(S_T - K,0)$ à la date T ? C'est le problème de la couverture, qui correspond à la détermination de la prime du call.

Notons que pour un put, la valeur à échéance est donnée par $Max(K - S_T ,0)$.

La réponse à ces deux questions ne peut se faire qu'à partir d'un minimum d'hypothèse de modélisation. L'hypothèse retenue est le principe d'absence d'opportunité d'arbitrage (A.O.A.) qui s'énonce comme suit : « *il est impossible de réaliser une série d'opérations financières qui procure un profit certain sans mise de fonds initiales c'est à dire sans prendre de risque* ».

Pour déterminer le montant de cette prime d'achat ou de vente, Black et Scholes d'une part, Merton d'autre part, ont effectué en 1973, une modélisation stochastique qui conduit à la résolution d'un problème aux limites. Par exemple, pour déterminer le prix d'un call européen, Black et Scholes ont proposé un modèle permettant de déterminer la prime par une formule explicite. Cependant pour le calcul d'un put américain, il n'existe pas de formules explicites et l'utilisation de méthodes de calcul numérique est indispensable. En fait, on montre que la prime du put ou du call américain, peut se calculer en résolvant une inéquation aux dérivées partielles. Dans la suite on note A un opérateur aux dérivées partielles, $\psi(x)= Max(S-K,0)$ pour un call ou $\psi(x) = Max(K-S,0)$ pour un put, suivant le cas, K est le prix d'exercice, r est le taux d'intérêt et S_t est le cours d'exercice à la date t. Avec ces notations, l'inéquation aux dérivées partielles s'écrit

$$\begin{cases} \dfrac{\partial u(x,t)}{\partial t} + A.u(x,t) - r.u(x,t) \geq 0,\ \mathrm{u}(x,t) \geq \psi(x),\ \text{dans } \left[\mathrm{o},\mathrm{T}\right] x R^n \\ (\dfrac{\partial u(x,t)}{\partial t} + A.u(x,t) - r.u(x,t))(\mathrm{u}(x,t) - \psi(x)) = 0,\ \text{dans } \left[\mathrm{o},\mathrm{T}\right] x R^n \\ u(x,T) = \psi(x) \end{cases} , \tag{10}$$

Dans cette équation (10) u correspond au supremum de l'espérance mathématique des flux actualisés que rapporte la mise de fonds initiale. A noter que dans le cas d'option européenne, on a à résoudre une équation aux dérivées partielles d'évolution, alors que dans le cas du calcul d'option américaine on a à résoudre une inéquation aux dérivées partielles d'évolution, à la différence près que les conditions aux limites ne sont pas ici précisées puisqu'on travaille dans un milieu non borné ; cependant aussi bien dans le cas du calcul d'option européenne que dans celui d'option américaine, des difficultés supplémentaires subsistent.

Dans la suite le papier est organisé comme suit : au second paragraphe est exposée la résolution numérique du problème de mathématiques financières, tout particulièrement les algorithmes parallèles avec échanges asynchrones entre les processeurs ; s'agissant de méthode itérative, la convergence est analysée. Le dernier paragraphe présente les résultats des expérimentations parallèles d'une part sur un clusteur et d'autre part sur une grille de calcul.

2 Résolution numérique du calcul d'options

2.1 Remarques préliminaires

A partir de la formulation (10) pour le calcul d'option américaine, ou de manière analogue pour le calcul d'option européenne, on peut effectuer plusieurs remarques

- le problème de mathématique financière est défini dans un domaine non borné, soit ici R^n ; sur le plan pratique et numérique, on résout le problème dans un domaine fermé borné Ω inclus dans R^n et, par des arguments d'analyse très sophistiqués, on montre que la solution du problème défini dans le domaine Ω tend vers celle du problème à résoudre, c'est-à-dire dans R^n, quand le mesure du domaine Ω tend vers l'infini [7],
- aussi bien dans le cas de calcul d'options américaine qu'européenne, on doit résoudre un problème d'évolution. Il est à noter que dans ce cas, on ramène habituellement la résolution numérique d'un tel problème dépendant du temps, à celle d'une suite de problèmes stationnaires par discrétisation convenable et classique de la variable temporelle qui sera abordée au paragraphe 2.2,
- en général l'opérateur A est l'opérateur de convection – diffusion ; cet opérateur n'est pas auto – adjoint. Cependant si les coefficients des dérivées sont constants, on peut toujours effectuer un changement de variable, tel que dans la nouvelle expression de l'opérateur, le coefficient du terme de gradient est nul ; ainsi l'opérateur intervenant dans le modèle est auto – adjoint. Considérons par exemple, le problème unidimensionnel suivant

$$\begin{cases} \dfrac{\partial v(x,t)}{\partial t} + c\dfrac{\partial v(x,t)}{\partial x} - \dfrac{\partial^2 v(x,t)}{\partial x^2} = f,\ [0,1]x[0,T] \\ v(x,0) = v_0(x) \\ v(0,t) = v(1,t) = 0 \end{cases} ;$$

si on pose v(x,t) = exp((αx+βt)).u(x,t), avec $\alpha = \dfrac{c}{2}$ et $\beta = -\alpha^2 = -\dfrac{c^2}{4}$, on vérifie que ce changement de variable conduit à résoudre le problème suivant

$$\begin{cases} \dfrac{\partial u(x,t)}{\partial t} - \dfrac{\partial^2 u(x,t)}{\partial x^2} = \exp(-(\alpha x + \beta t)).f,\ [0,1]x[0,T] \\ u(x,0) = \exp(-\alpha x).v_0(x) \\ u(0,t) = u(1,t) = 0 \end{cases} .$$

Cette remarque est particulièrement importante, car il est alors facile de montrer que la résolution du problème aux limites est équivalente à celle d'un problème de minimisation. Si les coefficients des dérivées ne sont pas constants, on n'a pas besoin de résoudre le problème d'optimisation associé et on résout directement le problème aux limites,

- dans le cas du calcul d'options européennes, en fonction des conditions aux limites, l'espace de travail est un espace vectoriel normé complet, et en exprimant la condition d'Euler, on aboutit à la résolution d'une équation aux dérivées partielles du type (5) ou (6) dont la résolution ne pose aucune difficulté. Dans le cas du calcul d'options américaines, on travaille dans un convexe fermé $\mathcal{C}$ et en exprimant la condition d'optimalité d'Euler, on aboutit à la résolution d'une inéquation aux dérivées partielles du type (10). Précisons que le problème de minimisation est strictement équivalent à celui de la résolution des équations ou des inéquations aux dérivées partielles, c'est à dire que toute solution du premier problème est solution du second et inversement,
- précisons enfin qu'on prend pour origine des temps, le moment de la vente ou de l'achat de l'option. Comme souligné lors des questions abordant les notions de pricing et de couverture, cela découle du fait que la valeur de la prime à l'échéance est connue comme une des données du problème et qu'en fait le problème revient à déterminer la valeur de la prime à l'instant t=0.

2.2 Discrétisation du problème de l'obstacle

Le calcul d'options européennes revenant, comme on l'a rappelé, à résoudre une équation aux dérivées partielles, ne pose aucun problème méthodologique. C'est pourquoi nous ne développerons pas cette problématique classique.

Par contre, nous développerons plus en détail le calcul d'options américaines. Sur le plan numérique, comme déjà indiqué, on doit résoudre ce même

problème dans un domaine Ω ce qui nécessite d'adjoindre, en plus de la condition finale u(x,T) = ψ(x) (rappelons que T représente l'échéance), des conditions aux limites ; en pratique on choisit soit des conditions aux limites de Dirichlet spécifiant la valeur de u sur la frontière du domaine Ω, soit des conditions aux limites de Neumann fixant la valeur de la dérivée normale à la frontière du domaine Ω (c.f. [7]).

La discrétisation des problèmes précédents ne pose pas de difficulté majeure.

Pour la variable temporelle on utilise les différences finies. En pratique, on préfère utiliser des schémas implicites ou semi – implicites, car ces derniers sont insensibles aux propagations d'erreurs de troncature liées à la discrétisation des opérateurs aux dérivées partielles ainsi qu'aux erreurs de chute liées à la représentation des nombres réels en machine ; ce qui assure par conséquent la stabilité numérique des schémas utilisés et évite un effet papillon. A chaque pas de temps, on est donc conduit à la résolution d'un système algébrique.

En ce qui concerne la variable d'espace, on peut utiliser soit des différences finies, soit des éléments finis soit même des volumes finis. En pratique, on utilise habituellement une discrétisation en espace par différences finies classiques, ce qui assure que les matrices de discrétisations vérifient des propriétés spécifiques, assurant la convergence des méthodes itératives parallèles synchrones ou asynchrones utilisées dans la suite. Ainsi, la dérivée seconde de u par rapport à la variable x sera approximée par le quotient différentiel suivant

$$\frac{\partial^2 u(x,y,z)}{\partial x^2} \approx \frac{u(x+\delta x,y,z) - 2.u(x,y,z) + u(x-\delta x,y,z)}{(\delta x)^2} + O(\delta x^2) \ , \qquad (11)$$

où δx représente le pas de discrétisation spatial. On utilise des formules analogues pour obtenir une approximation des dérivées secondes par rapport aux variables y et z. De plus, pour discrétiser les dérivées premières à 0(δx) près, on considère un schéma décentré forward si le coefficient de convection est négatif ou un schéma décentré backward si le coefficient de convection est positif, défini respectivement par

$$\frac{\partial u(x,y,z)}{\partial x} \approx \frac{u(x+\delta x,y,z) - u(x,y,z)}{\delta x} \text{ ou } \frac{\partial u(x,y,z)}{\partial x} \approx \frac{u(x,y,z) - u(x-\delta x,y,z)}{\delta x} \qquad (12)$$

On est donc conduit à résoudre à chaque pas de temps, des systèmes algébriques de grandes tailles, pour lesquels, compte tenu de la propagation des erreurs de chute, l'utilisation de méthodes itératives est fortement recommandée.

Dans la suite, nous noterons par des lettres majuscules les analogues discrets des inconnues et des données du problème ; par exemple U représentera l'analogue discret de u intervenant dans (11)–(12). On notera aussi **A** la matrice de discrétisation du problème aux limites, obtenue en utilisant les formules de type (11)-(12), et F le second membre du système discrétisé.

2.3 Linéarisation du problème de l'obstacle

Comme déjà indiqué, le problème de calcul d'options américaines est fortement non linéaire. On doit donc à chaque itération, linéariser par un procédé approprié le système à résoudre. On dispose de deux méthodes de linéarisation
- soit l'utilisation d'une méthode de Richardson projetée sur le convexe $\mathcal{C}$,
- soit la méthode d'Howard à partir de l'écriture sous forme complémentaire (voir [2]) du problème de l'obstacle.

Nous présentons d'abord la méthode de Richardson projetée puis plus brièvement la méthode d'Howard.

On a vu que la résolution du problème d'options européennes ou américaines revient à résoudre un problème de minimisation soit dans un espace vectoriel, soit dans un convexe. Cette équivalence entre les deux types de problèmes joue un rôle important au plan algorithmique. On va donc considérer une variante parallèle synchrone ou asynchrone de la méthode de Richardson projetée sur le convexe $\mathcal{C}$, pour le calcul d'option américaine, ce qui revient à résoudre numériquement, une équation de point fixe.

La parallélisation d'un algorithme nécessite la décomposition du problème en plusieurs sous - problèmes interconnectés. Sur la plan mathématique, on travaille donc sur des espaces produits. Si E désigne l'espace de travail en dimension finie, on décompose donc E en un produit fini de p espaces E_i comme suit $E = \prod_{i=1}^{p} E_i$, où ici p désigne le nombre de processeurs utilisés. De plus, dans la mesure où on souhaite résoudre un problème d'optimisation, il est clair que l'espace E ainsi que les espaces E_i, i=1,..p, sont des espaces de Hilbert. Pour tout $V \in E$, soit $P_C(V)$ la projection orthogonale de E sur le convexe discrétisé C; on considère une décomposition compatible de $P_C(V)$ avec la décomposition en espace produit définie par $P_C(V) = \left\{....,P_{C_i}(V_i),...\right\}$, où $P_{C_i}(V_i)$ est le projecteur orthogonal de E_i sur C_i, pour i = 1,..,p. Soit δ un nombre réel positif donné ; définissons l'application de point fixe suivante

$$U = P_C(V - \delta(AV - F)) = F_\delta(V) \ ; \tag{13}$$

De façon analogue, on peut décomposer l'application de point fixe précédente, comme suit $F_\delta(V) = \left\{....,F_{i,\delta}(V),...\right\}$; pour i =1,..,p, on écrit donc

$$U_i = P_{C_i}(V_i - \delta(\sum_{j=1}^{p} A_{i,j}V_j - F_i)) = F_{i,\delta}(V). \tag{14}$$

L'approximation initiale $U^{(0)}$ étant donnée, on définit les itérations asynchrones par

$$U_i^{(q+1)} = \begin{cases} U_i^{(q)} & \text{si i } \notin \text{ s(q)} \\ F_{i,\delta}(....,U_j^{\rho_j(q)},....) & \text{si i } \in \text{ s(q)} \end{cases} \tag{15}$$

où s(q) représente l'ensemble des composantes i relaxées et vérifie $s(q) \subset \{1,….,p\}$, condition prenant en compte la mise à jour des composantes en parallèle ; de plus s(q) vérifie également la condition

$$\text{pour tout i} \in \{1,…,p\}, \text{ l'ensemble } \{q \mid i \in s(q)\} \text{ est dénombrable,} \tag{16}$$

ce qui traduit que, théoriquement, on relaxe une infinité de fois chaque composante. De plus les exposants $\{\rho_j(q)\}$ modélisent les échanges asynchrones entre les processeurs; ces exposants sont tels que pour tout j = 1,…,p, $\rho_j(q) \in N$, pour tout nombre entier q ; de plus, ces exposants vérifient les propriétés suivantes

$$0 \leq \rho_j(q) \leq q \quad \text{et} \quad \lim_{q \to \infty} (\rho_j(q)) = +\infty, \tag{17}$$

cette dernière hypothèse prenant en compte une hypothétique panne des processeurs.

Remarque 1 : lorsque pour tout j = 1,…, p et pour tout $q \in N$, on a $\rho_j(q) = q$, alors (15) modélise un algorithme parallèle synchrone, qui apparaît comme un cas particulier d'algorithme parallèle asynchrone. De plus, toujours dans ce contexte synchrone, si $s(q) = \{1, \ldots, p\}$ ou s(q) = q modulo(p) +1, respectivement, alors (15) modélise la méthode séquentielle de Jacobi ou de Gauss – Seidel, respectivement.

Pour résumer, lorsqu'on utilise les méthodes itératives parallèles asynchrones, les processeurs effectuent en parallèle la résolution de chaque sous - problème en utilisant les données d'interactions disponibles produites par les autres processeurs. L'analyse de la convergence de ces méthodes s'effectue en utilisant la notion de M-matrice dont on rappelle la définition

Définition 1 : une matrice ayant des coefficients hors diagonaux négatifs, est une M – matrice si celle-ci est non singulière et d'inverse non négative.

En application du théorème de Perron Frobenius (c.f. [13]), il est à noter que si J est la matrice de Jacobi d'une M – matrice, alors J est une matrice non négative admettant un rayon spectral $\mu \in R$, tel que $0 < \mu < 1$, auquel est associé un vecteur propre Γ de composantes Γ_l strictement positives. Cette propriété joue un rôle essentiel pour l'analyse de la convergence des algorithmes parallèles asynchrones.

On suppose que la décomposition en blocs de la matrice A soit telle que les blocs diagonaux sont fortement définis positifs, c'est à dire

$$<A_{i,i}. V_i, V_i> \geq n_{i,i}. \|V_i\|_{i,2}^2, \tag{18}$$

où $\|V_i\|_{i,2}^2$ dénote la norme euclidienne du sous – vecteur $V_i \in E_i$; de plus, on suppose que les normes induites des sous – matrices $A_{i,j}$ sont bornées par des nombres - $n_{i,j}$ (les réels $n_{i,j}$ étant des nombres négatifs), ce qui est toujours vrai. Conformément à un résultat de [9], ces deux hypothèses sont équivalentes à l'hypothèse globale suivante

$$< A_{i,i}.V_i + \sum_{j \neq i} A_{i,j}.V_j, V_i > \geq \sum_{j=1}^{p} n_{i,j}.\|V_i\|_{i,2}.\|V_j\|_{j,2}, \ \forall\, i \in \{1,...,p\}, \ \forall\, V \in E. \tag{19}$$

Il convient de remarquer que lorsque les discrétisations sont effectuées convenablement et conformément à (11)-(12), la condition (18) est vérifiée, ce qui assure la validité de l'hypothèse (19).

Théorème 1 : L'hypothèse (19) étant vérifiée, et si de plus la matrice N de coefficients $(n_{i,j})$ est une M – matrice, il existe un nombre réel strictement positif δ_0, tel que pour tout $\delta \in]0, \delta_0[$, les variantes parallèles synchrone et asynchrone (14) - (15) de l'algorithme de Richardson projeté converge vers la solution du problème.

Principe de la démonstration : elle est basée sur le fait que, d'une part l'opérateur de projection orthogonal est une contraction et d'autre part sous les hypothèses considérées, l'application de point fixe est une contraction dans l'espace E normé par

la norme uniforme avec poids $W \rightarrow \|W\|_{\mu,J} = \underset{l}{Max}(\frac{\|W_l\|_{l,2}}{\Gamma_l})$, μ étant la constante de contraction strictement inférieure à l'unité [2] (c.f. [1]).

Comme indiqué précédemment, la méthode d'Howard utilise la formulation discrète correspondant à une formulation du problème d'obstacle mis sous forme complémentaire, soit

$$\text{Max}(\mathbf{A}.U - F, U - \Psi) = 0, \tag{20}$$

où Ψ correspond à l'analogue discret de l'obstacle ψ intervenant dans la discrétisation du problème (10). L'écriture précédente sous forme complémentaire fournit un procédé de résolution numérique du problème. En effet, pour linéariser ce problème (20), on va considérer une méthode analogue à la classique méthode de Newton utilisée pour résoudre un système algébrique non linéaire. Soit $U^{(0)}$ une approximation initiale de la solution du problème à résoudre au $k^{ième}$ pas de temps ; dans le cadre de l'intégration en temps du problème de l'obstacle, $U^{(0)}$ donnée initiale de la méthode de linéarisation, peut par exemple être choisie comme la valeur obtenue à la $(k-1)^{ième}$ itération en temps. Au départ, lors de l'intégration au premier pas de temps, dans la mesure où on intègre le problème aux limites dans le sens rétrograde, on choisit la condition finale $\psi(x)$ comme valeur d'initialisation de $U^{(0)}$ de l'algorithme de Howard . A (19), on associe un système linéarisé.

$$B^{(k)} . U = G^{(k)} ; \tag{21}$$

l'algorithme de Howard est schématisé par le diagramme suivant

Boucle en temps

$U^{(0)}$ donné;

Norme ← 1 + précision ;

Tant que norme > précision **faire** :

Construction de la matrice linéarisante $B^{(k)}$ par Howard
Construction du second membre $G^{(k)}$ par Howard

Résolution du système $B^{(k)}.U^{(1)} = G^{(k)}$

Norme ← $\| U^{(1)} - U^{(0)} \|$
$U^{(0)} \leftarrow U^{(1)}$

fait

fin boucle en temps

la matrice linéarisante $B^{(k)}$ et le second membre $G^{(k)}$ étant définis comme suit

- si la $l^{ième}$ composante de $(\mathbf{A}.U - F)$ est supérieure à la $l^{ième}$ composante de $(U - \Psi)$, alors la $l^{ième}$ ligne de la matrice $B^{(k)}$ sera la $l^{ième}$ ligne de la matrice $\mathbf{A}$ et la $l^{ième}$ composante de $G^{(k)}$ sera la $l^{ième}$ composante de F,
- sinon, dans le cas contraire, la $l^{ième}$ ligne de la matrice $B^{(k)}$ sera constituée par une ligne de zéros, sauf l'élément diagonal qui sera égal à un et la $l^{ième}$ composante de $G^{(k)}$ sera égale à la $l^{ième}$ composante de Ψ.

[2] μ rayon spectral de la matrice de Jacobi de la M-matrice $N = (n_{i,j})$

Pour la résolution du système linéarisé, on peut utiliser n'importe quelle méthode de résolution, y compris les méthodes parallèles synchrones et asynchrones par sous domaines avec ou sans recouvrement (méthode alternée de Schwarz). Compte tenu du procédé de discrétisation choisi, qui conduit à des matrices de discrétisations **A** qui sont des M – matrices, on vérifie que les matrices linéarisantes $B^{(k)}$ sont à chaque pas des M – matrices, ce qui assure la convergence des méthodes parallèles synchrones et asynchrones par sous domaines.

Théorème 2 : la matrice de discrétisation **A** étant une M – matrice, les méthodes parallèles synchrones et asynchrones par sous domaines avec ou sans recouvrement, convergent vers la solution du problème linéarisé (21).

Démonstration : on utilise des techniques d'ordre partiel (c.f. [11]).

Remarque 2 : Il est à noter qu'intuitivement, la méthode de Howard utilisée dans le cas d'option américaine, revient en fait à effectuer une projection sur le convexe.

3 Résultats expérimentaux

3.1 Résultats code séquentiel

Pour effectuer les calculs d'options américaines on a mis en œuvre la méthode de Richardson projetée; pour faciliter la portabilité des codes, ces derniers sont écrits en langage C. On limite essentiellement la présentation des résultats expérimentaux au cas du calcul d'options américaines; en effet le calcul d'options européennes revient à résoudre une équation aux dérivées partielles linéaire. Il est à noter que l'algorithme utilisé a de bonnes performances.

Le solveur de calcul d'options américaines a été testé avec une taille de problème correspondant au nombre de points de discrétisation dans le domaine Ω; soit ici 96 x 96 x 96 = 884 736 points de discrétisation. Sur le clusteur utilisé le temps de calcul est de 708 secondes. La vitesse de convergence est bonne.

2.4 Résultats code parallèle

Les essais numériques ont été effectués sur un réseau de processeurs POWER 3 organisé en grappe de type SMP (clusters) dans lequel chaque nœud possède 16 processeurs partageant une mémoire commune. Lors des expérimentations 2, 4, 8, 16 et 32 processeurs ont été utilisés. La parallélisation des codes séquentiels est effectuée au moyen de MPI. Il est à noter qu'au-delà de 16 processeurs, on utilise 2 nœuds, alors que jusqu'à 16 processeurs un seul nœud est utilisé.

La table 1 donne les temps de restitutions de l'algorithme de Richardson projeté sur le réseau de processeurs. La table 2 et 3, respectivement donnent un récapitulatif de l'accélération et de l'efficacité de l'algorithme de Richardson projeté et parallélisé. La figure n°1 permet de comparer l'efficacité des méthodes synchrones (x) et asynchrones (o) entre elles, dans le cas du problème considéré.

Sur le plan numérique, on constate que le nombre de relaxations nécessaires pour converger, effectuées par l'algorithme synchrone est identique à celui nécessaire en mode séquentiel. Par contre en mode asynchrone il y a un surcoût minime en nombre de relaxations, de l'ordre de 5%, du au comportement chaotique des communications inter processeurs. Le nombre de relaxation a donc tendance à augmenter légèrement

Table 1. Temps de restitution de l'algorithme parallélisé (96 x 96 x 96).

Nb de processeurs	Mode synchrone	Mode asynchrone
1	708 secondes	708 secondes
2	358 secondes	349 secondes
4	161 secondes	163 secondes
8	80 secondes	76 secondes
16	46 secondes	40 secondes
32	35 secondes	23 secondes

Table 2. Accélération de l'algorithme parallélisé (temps parallèle sur temps sequential).

Nb de processeurs	Mode synchrone	Mode asynchrone
2	1.98	2.03
4	4.4	4.34
8	8.85	9.32
16	15.39	17.7
32	20.23	30.78

Table 3. Efficacité de l'algorithme parallélisé (Accélération sur nb. de proc.).

Nb de processeurs	Mode synchrone	Mode asynchrone
2	0.99	1.01
4	1.10	1.08
8	1.10	1.16
16	0.96	1.10
32	0.63	0.96

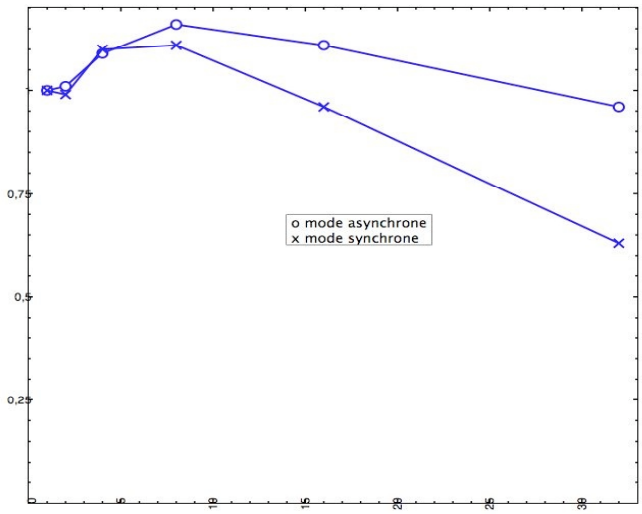

Fig. 1. *Efficacité en fonction du nombre de processeurs pour le problème 96 x 96 x 96*

sans que pour autant le temps de calcul soit pénalisé. En effet ce surcoût n'est pas pénalisant dans la mesure où le temps de restitution reste inférieur à celui obtenu avec l'algorithme en mode synchrone. De plus, un nombre supplémentaire de relaxations en mode asynchrone a une incidence positive sur la précision obtenue et a donc pour effet d'améliorer la qualité numérique de la solution obtenue.

Il est à noter que les algorithmes qu'ils soient en mode synchrone ou en mode asynchrone ont des performances très proches. Ceci est dû au fait que le nombre de points de discrétisation est relativement petit. En dessous de 8 processeurs, l'algorithme en mode synchrone peut avoir de meilleures performances ; en fait, on a pu remarquer que le découpage en blocs du vecteur solution a une influence sur les performances. Par contre, à partir de 8 processeurs, l'algorithme en mode asynchrone donne toujours de meilleures performances que l'algorithme en mode synchrone. Dans ce dernier cas, cette chute de performance est due au poids des synchronisations dans les méthodes synchrones car les temps d'attente entre processeurs pénalisent ce type d'algorithme. Soulignons que les communications en mode asynchrone ont un grand intérêt lorsque le nombre de processeurs augmente car les phénomènes de deadlock sont largement minimisés ce qui n'est pas le cas des communications en mode synchrone. Signalons que lors de la mise au point de l'implémentation, on a comparé les performances entre méthode synchrone et asynchrone, pour la résolution d'un problème de convection – diffusion correspondant par exemple au calcul d' options européennes. Pour un nombre de points de discrétisation de l'ordre de 3 750 000, avec 128 processeurs, les communications asynchrones permettent de diminuer le temps de restitution de l'ordre d'un tiers.

L'étude des valeurs de l'accélération et de l'efficacité résumée respectivement en table 2 et 3 confirme l'intérêt des méthodes utilisées. Pour le problème considéré, l'algorithme asynchrone se démarque de l'algorithme synchrone à partir de 8 processeurs. Globalement, les deux versions de l'algorithme ont des performances équivalentes dans le cas où un seul nœud du cluster est utilisé. Une différence très significative entre les deux modes d'algorithmes parallèles apparaît lorsque 32 processeurs sont utilisés. La perte d'efficacité de l'algorithme en mode synchrone peut s'expliquer de la manière suivante

- lorsque 32 processeurs sont utilisés, chaque processus ne prend en charge que 3 blocs. Comme les messages et les blocs sont pratiquement de même taille, le surcoût de communications atteint une proportion significative,
- deux processus sont impliqués dans les communications faisant intervenir le réseau d'interconnexion entre les nœuds. Ce type de communication étant plus lent, cela entraîne un déséquilibrage de charge qui retarde les autres processeurs.
- Il est clair que l'algorithme parallèle asynchrone est moins sensible aux problèmes liés aux communications et à l'équilibrage des charges que l'algorithme parallèle synchrone. Il est à noter que l'algorithme asynchrone subit aussi une chute de performance normale lorsque 32 processeurs sont utilisés. Toutefois, dans ce cas, l'efficacité reste bonne, de l'ordre de 0.96.
- On remarque que lorsqu'il y a peu de processeurs, on obtient des efficacités supérieures à l'unité. Ce phénomène est dû au fait que dans ce cas, le poids des synchronisations est faible par rapport au temps de calcul pur ; ce phénomène découle aussi de défauts des mémoires caches, générés par le volume important de données à

échanger. Par contre, lorsque le nombre de processeurs augmente, les efficacités obtenues sont inférieures à l'unité, ce qui est normal.

Pour terminer et montrer l'intérêt du calcul parallèle asynchrone dans le cas d'applications de grande taille, signalons quelques tests en cours sur une grille de calcul. Si on considère des essais effectués sur un premier cluster comportant 20 processeurs, on obtient un temps de restitution synchrone de 53.164 secondes et un temps de restitution asynchrone de 34.627 secondes. Si on considère un second clusteur de 20 processeurs, on obtient un temps de restitution synchrone de 56.00 secondes et un temps de restitution asynchrone de 39.491 secondes. Donc, sur chacun des clusteurs, les algorithmes synchrones et asynchrones ont des comportements quasiment identiques. Si à présent on effectue les mêmes calculs en utilisant 10 processeurs sur chacun de ces clusters très distants l'un de l'autre, on obtient un temps de restitution synchrone de 5 minutes 12.574 secondes et un temps de restitution asynchrone de 39.989 secondes. On constate clairement que sur deux clusteurs la version synchrone est pénalisée avec une perte de performance d'un facteur de l'ordre de 8 et que les synchronisations dégradent nettement les performances des algorithmes.

Références

1. Chau, M., Spiteri, P., Parallel asynchronous Richardson method for the solution of obstacle problem. In : Proceedings of HPCS 2002, pp. 133 – 138, IEEE Press, Los Alamitos (2002)
2. Cottle, R.W., Golub, G.H., Sacher, R.S., On the solution of large structured linear complementary. The block partitioned case. Appl. Math. Optim., 347 – 363 (1978)
3. Duvaut, G., Lions, P.L., Les inéquations en mécanique et en physique. Dunod (1972)
4. Giraud, L., Spiteri, P., Résolution parallèle de problèmes aux limites non linéaires. M^2AN, 25 – 5, 579 – 606 (1991)
5. Glowinski, R., Lions, P.L., Tremolières, R., Analyse numérique des inéquations variationnelles. Dunod, Tome 1 & 2 (1976)
6. Hull, J., Options, futures et autres actifs dérivés. 5ème édition, Pearson Education (2004)
7. Lamberton, D., Lapeyre, B., Introduction au calcul stochastique appliqué à la finance. Ellipses (1991)
8. Miellou, J.C., Spiteri, P., Two criteria for the convergence of asynchronous iterations. Dans Computers and Computing, Chenin, P., et all ed., Masson & Wiley, 90 – 95 (1985)
9. Miellou, J.C., Spiteri, P., Un critère de convergence pour des méthodes générales de point fixe. M^2AN, 19 – 4, 645 – 669 (1985)
10. Spiteri, P., Simulations d'exécution parallèle pour la résolution d'inéquations variationnelles stationnaires. Revue EDF, Informatique et Mathématiques Appliquées, série C, 1, 149 – 158 (1983)
11. Spiteri, P., Miellou, J.C., El Baz, D., Asynchronous Schwarz alternating methods with flexible communications for the obstacle problem. Résaux et systèmes répartis, 13 – 1, 47 – 66 (2001)
12. Wilmott, P., Howison, S., Dewyne, J., The mathematics of financial derivatives : a studient introduction. Cambridge university press (1995)
13. Varga, R.S., Matrix iterative analysis, Prentice all (1962)