

Parallélisation des algorithmes d'analyse d'images par champs markoviens dans un contexte pyramidal

E. MEMIN et F. HEITZ

IRISA/INRIA, Campus de Beaulieu 35042 Rennes Cedex, France
Tel: 99 89 71 00 Telex: UNIRISA 950 473 F
FAX: 99 38 38 32

RÉSUMÉ

L'émergence, ces dernières années, de nombreuses applications de traitement d'image bas niveau utilisant une modélisation statistique par champ markovien et les problèmes d'optimisation globale qui leur sont associés ont motivé l'étude d'algorithmes de relaxation parallèles.

Dans cet article, différents modes de parallélisation pour les algorithmes de relaxation sont décrits.

La comparaison, pour une application de mesure du mouvement, de trois algorithmes (monorésolution, multigrille, multiéchelle) nous permet de dégager un algorithme de relaxation efficace et se parallélisant de façon naturelle.

1 Introduction

L'exploitation de la théorie des champs markoviens dans différents domaines de l'analyse d'image bas niveau n'a cessé de croître ces dernières années. Ces modèles statistiques ont été appliqués avec succès dans des domaines aussi divers que la restauration d'image [7], la segmentation par les textures, la stéréovision ou l'analyse du mouvement [3, 6, 9].

Ces modèles trouvent leur cadre naturel de mise en œuvre dans la théorie de l'estimation bayésienne globale. Dans sa forme usuelle (critère du MAP), [7], cette dernière conduit à la minimisation d'une fonction d'énergie non convexe sur un espace de configurations généralement extrêmement vaste.

Il s'agit là d'un problème d'optimisation globale qui peut se résoudre grâce à des algorithmes de relaxation.

L'objet de cet article est d'étudier des classes particulières d'algorithmes de relaxation se prêtant à des implémentations parallèles efficaces. Différents types d'algorithmes (monorésolution, multigrille, multiéchelle) ainsi que différentes versions parallèles de ces algorithmes sont évalués dans le cadre de la mesure du mouvement. Il s'en dégage un algorithme multiéchelle particulièrement efficace et structurellement simple, dont l'implantation parallèle est en cours.

2 Champs de Markov et algorithmes de relaxation

L'estimation bayésienne globale offre un cadre méthodologique au problème de l'estimation d'un champ aléatoire de primitives $e = \{e_s, s \in S\}$ à partir d'un champ d'observations $o = \{o_s, s \in S\}$. Chaque primitive locale e_s sera supposée prendre ses valeurs dans un ensemble discret Λ et Ω désignera l'ensemble fini de toutes les configurations possibles de e . Le critère du MAP (*maximum a posteriori*) conduit à chercher le champ \hat{e} tel que : $\hat{e} = \arg \max_{e \in \Omega} p(o/e)p(e)$

La distribution $p(o/e)$ est obtenue en modélisant les liens statistiques entre observations et primitives. Le deuxième terme résulte d'une modélisation *a priori* des primitives. Il peut être spécifié en supposant le champ e markovien. Celui-ci est entièrement déterminé par ses caractéristiques locales définies sur un voisinage Γ et suit une distribution de Gibbs, [7] :

$$p(e) = \frac{\exp(-U(e))}{Z}$$

La fonction U , ou fonction d'énergie se décompose comme une somme de potentiels locaux définis sur des sous-ensembles de sites mutuellement voisins $c \in C$ appelés cliques : $U(e) = \sum_{c \in C} V_c(e)$. Le terme Z est une constante de normalisation définie par : $Z = \sum_{e \in \Omega} \exp -U(e)$.

ABSTRACT

The use of Markov Random Field models has recently brought new powerful solutions to low level image analysis problems. Those models lead to massively parallel algorithms such as stochastic or deterministic relaxation. In this paper comprehensive parallelization methods for those relaxation algorithms are described. Different algorithms (monoresolution, multigrid, multiscale) are compared in a motion measurement problem. The multiscale approach exhibits excellent convergence properties and may be easily parallelized.

Si $p(o/e) \neq 0$ la solution au sens du MAP peut s'écrire :

$$\hat{e} = \arg \max_{e \in \Omega} \frac{\exp -(U(e) - \ln(p(o/e)))}{Z} = \arg \min_{e \in \Omega} U(o, e)$$

avec $U(o, e) = U(e) - \ln(p(o/e))$

On se ramène donc à un problème de minimisation d'une fonction d'énergie globale. Cette minimisation est en général difficile, l'ensemble des configurations Ω étant extrêmement vaste et les fonctions d'énergie généralement non convexes (et présentant donc de nombreux minima locaux). La minimisation de la fonction d'énergie s'effectue généralement grâce à des algorithmes de relaxation, [2, 7].

2.1 Exemple : un modèle d'estimation du flux optique (cas discret)

Nous considérerons dans la suite un modèle d'estimation des déplacements entre deux images consécutives, qui servira de comparaison entre les différentes versions d'algorithmes de relaxation étudiées ici.

Soit $f(s, t)$ la fonction d'intensité au point $s = (x, y)$, $s \in S$ et au temps t . Le vecteur vitesse au point s est noté : $w_s(u_s, v_s)$, $u_s = \frac{dx}{dt}(s)$, $v_s = \frac{dy}{dt}(s)$ et $\vec{w} = \{w_s, s \in S\}$ désigne le champ des vecteurs vitesse à estimer. Les vitesses sont discrétisées sur l'ensemble $\Lambda = (-u_{max} : u_{max}, -v_{max} : v_{max})$ avec un pas δ . Le champ de Markov adopté est associé à un huit voisinage et est spécifié par la fonction d'énergie suivante :

$$U(f, \vec{w}) = \sum_{s \in S} [f(s, t) - f(s + \vec{w}_s, t + dt)]^2 + \alpha^2 \sum_{(r, s) \text{ voisins}} \|\vec{w}_s - \vec{w}_r\|^2$$

Le premier terme (différence interimage déplacée) exprime la conservation de la luminance d'un point après déplacement. Ce terme non linéaire conduit à une fonction d'énergie non convexe. Le deuxième terme exprime les propriétés *a priori* du champ à estimer et peut être interprété comme un terme de régularisation favorisant les champs "lisses". Cet exemple a été choisi pour son caractère représentatif et pour sa difficulté en terme de minimisation.

2.2 Relaxation stochastique et relaxation déterministe

Les algorithmes de relaxation stochastiques peuvent être vus comme la combinaison d'une procédure d'échantillonnage selon la distribution de Gibbs et d'une procédure de descente en température. Deux algorithmes permettent d'échantillonner la distribution de Gibbs : l'algorithme de Metropolis et l'échantillonneur de Gibbs. Ces algorithmes construisent des chaînes de Markov admettant comme distribution stationnaire la distribution *a posteriori* :

$$p_{e/o}(e) = \frac{1}{Z(T)} \exp\left(-\frac{U(e, o)}{T}\right)$$



ou T désigne la température. Si celle-ci décroît suffisamment lentement l'algorithme converge théoriquement vers un mode de la distribution c'est-à-dire vers une configuration \hat{e} d'énergie minimale, (pour une analyse détaillée voir [7]). Une descente continue de la température, suivant une loi exponentielle a été adoptée dans nos expérimentations.

Nous considérons ici l'échantillonneur de Gibbs qui peut être décrit de la manière suivante :

- Tirer un site s d'après une loi uniforme sur S
 - Tirer une nouvelle configuration d'après la distribution locale $p_{e/o}(e(s)/e(s'), s \neq s')$
 - Itérer jusqu'à convergence.
- Ces calculs sont locaux, grâce au caractère markovien des modèles utilisés et n'implique qu'un site et ses voisins, [7].

Les algorithmes de relaxation déterministes sont des versions sous-optimales des algorithmes stochastiques. Ils ont les mêmes propriétés que leurs homologues stochastiques à température nulle. En particulier seules des configurations d'énergie plus faible sont acceptées. Ces algorithmes restent donc piégés dans les minima locaux de la fonction d'énergie et sont sensibles aux conditions initiales. L'algorithme que nous avons considéré ici est l'algorithme des ICM [2] qui peut s'interpréter comme une relaxation de Jacobi.

2.3 Algorithmes parallèles de relaxation

Depuis quelques années, la parallélisation des algorithmes d'optimisation globale par relaxation a motivé des recherches dans des domaines aussi variés que la conception de circuit ou le traitement d'image [1, 5, 7, 10]. Dans le contexte de l'analyse d'image nous avons dégagé trois classes d'approches parallèles correspondant à des niveaux de parallélisme différents :

- chaînes de Markov multiples en parallèle, [1, 8]
- remise à jour simultanée des sites images, [7, 10]
- parallélisation de la remise à jour locale, [5]

Les premières approches historiquement étudiées appartiennent à la deuxième classe et consistent en une parallélisation massive des algorithmes de relaxation dans laquelle plusieurs sites de l'image sont remis à jour simultanément. En traitement d'image les travaux [7, 10] et en conception de circuit [4] appartiennent à cette classe.

La troisième classe consiste en une parallélisation de grain plus fin où c'est la réactualisation en un site qui est elle-même parallélisée. Un schéma d'architecture systolique est développé à cet effet dans [5].

La première classe a été introduite par Aarts et Laarhoven [1]. Il s'agit d'un parallélisme de niveau "haut" qui consiste schématiquement à lancer plusieurs algorithmes de relaxation en parallèle et à les faire interagir périodiquement. Une méthode de ce type a récemment été étudiée par Graffigne [8]. Son algorithme consiste à développer plusieurs chaînes de Markov homogènes en parallèle, à température constante mais différente d'une chaîne à l'autre. Chaque chaîne peut être contrôlée par un processeur. Ces chaînes interagissent périodiquement de la manière suivante : toutes les s transitions, un transfert de configuration est accepté par un processeur si une chaîne à plus haute température propose une configuration plus basse en énergie. Cet algorithme présente de bonnes propriétés de convergence vers les vallées d'attraction du minimum global et a donc été retenu comme version représentative de ce type de parallélisme. Dans la section suivante nous évaluons les propriétés de cette approche dans le cas de la mesure du mouvement. Trois algorithmes originaux sont développés : une version monorésolution (proche de celle présentée dans [8]), une version multigrille et une version multiéchelle.

3 Chaînes de Markov multiples en parallèle

3.1 Algorithme monorésolution

Dans l'algorithme décrit dans [8] le transfert d'une configuration d'une chaîne à l'autre est basé sur la comparaison de l'énergie de leurs configurations courantes respectives. Cependant du fait de la taille importante de l'espace des configurations dans les problèmes de traitement d'image, les interactions sont peu nombreuses. En effet les processeurs à haute température n'ont qu'une faible probabilité de tirer des configurations d'énergie plus faible que celles générées par des chaînes à plus basses températures.

Un moyen efficace d'augmenter les échanges de configurations entre

processeurs consiste à interagir non plus sur des images entières mais sur des configurations locales définies sur des blocs réguliers. Les blocs de la chaîne P transmis à la chaîne $P + 1$ sont alors ceux dont l'énergie est plus faible que celle de leurs correspondants dans la chaîne $P + 1$.

Notons $T(k)$ la température du processeur k , X_j^i désigne la configuration de la chaîne j après s transitions et $P_{T(k)}(X)$ est la réalisation d'un pas de la chaîne à température $T(k)$ en partant de la configuration X . Soit $X = \cup_j B_i(X)$, $B_i(X)$ correspondant au i ème bloc de la configuration X , l'algorithme s'écrit :

$$\begin{aligned} X_{n+1}^1 &= P_{T(1)}(X_n^1) \\ \forall l \neq 1 \quad X_{n+1}^l &= P_{T(l)}(X_n^l) \text{ si } n+1 \neq ks \\ X_{ks}^l &= \{B_i / U(B_i(P_{T(l)}(X_{ks-1}^l))) \leq U(B_i(X_{ks-1}^{l-1}))\} \cup \\ &\quad \{B_i / U(B_i(X_{ks-1}^{l-1})) < U(B_i(P_{T(l)}(X_{ks-1}^l)))\} \end{aligned}$$

$T(1) > T(2) > \dots > T(k) > \dots T(P) \simeq 0$ et P désigne le nombre de chaînes.

Résultats expérimentaux

Pour valider les propriétés de convergence de l'algorithme parallèle, un jeu de 41 séquences tests d'images 64x64 auxquelles a été appliqué un mouvement synthétique a été construit. Le mouvement est constitué de quatre blocs de mouvements différents : une translation, une rotation, une divergence et une composée de rotation et de divergence. La figure 1 présente pour les 41 paires d'images le rapport entre l'énergie finale obtenue par relaxation stochastique séquentielle (RS) (section 2.2) et l'énergie finale des configurations obtenues par l'algorithme basé sur les chaînes multiples en parallèle (CMP). L'algorithme CMP a été systématiquement lancé pour 100 itérations et pour quatre processeurs virtuels, avec le même jeu de températures ($T_0 = 80$; $T_1 = 30$; $T_2 = 10$; $T_3 = 0.01$). Une itération correspond à un passage complet sur l'image. Pour l'algorithme RS la descente en température est de la forme $T_n = \alpha \cdot T_{n-1}$ ($\alpha = 0.97$). Le nombre d'itérations nécessaire pour atteindre la convergence avec l'algorithme RS est présenté Fig 4, il est de 409 en moyenne.

On constate que la déviation entre l'algorithme parallèle et la relaxation stochastique est faible (8% au pire et pour une seule image). Une centaine d'itération de l'algorithme CMP est cependant nécessaire pour atteindre des résultats satisfaisants. Le processeur à haute température ayant à parcourir de manière presque exhaustive l'espace des configurations, il peut être intéressant de réduire cet espace. Pour cela nous proposons d'introduire une technique multigrille qui permet d'explorer rapidement des espaces de configurations réduits aux résolutions grossières.

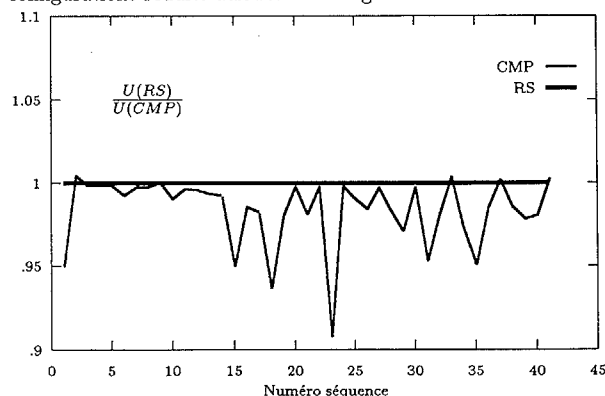


Figure 1 : Rapport des énergie finales $\frac{U(RS)}{U(CMP)}$

3.2 Algorithme multigrille

Afin de construire une succession d'espaces de configurations de résolutions décroissantes, une pyramide passe-bas des observations est créée (figure 2). Chaque niveau de cette pyramide est associé à une grille, de résolution équivalente qui supporte les primitives à estimer. Deux algorithmes de relaxation sont comparés sur cette structure.

- Un algorithme multigrille standard de type descendant, [12] (MGD) dans lequel l'estimation se fait de la résolution la plus grossière vers la plus fine. A chaque niveau un algorithme de re-

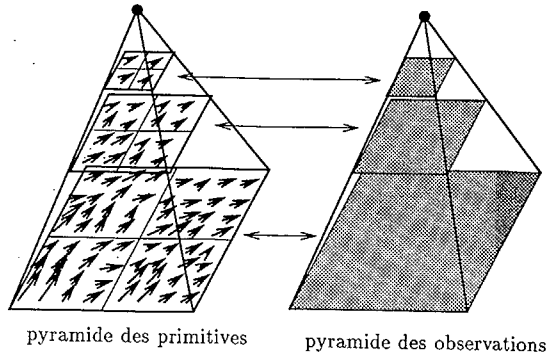


Figure 2 : Structure multigrille

laxation déterministe de type ICM est employé.

• Un algorithme multigrille basé sur les chaînes de Markov en parallèle (MGP) où chaque niveau de la pyramide est associé à un processeur qui, comme précédemment, développe une chaîne homogène à température constante. Les températures sont décroissantes : une température haute est associée au processeur travaillant à la résolution la plus grossière tandis qu'une température proche de zéro est associée au processeur travaillant à la résolution la plus fine. Comme pour l'algorithme présenté précédemment les interactions sont périodiques et définies sur des blocs de l'image. La comparaison de deux niveaux de la pyramide nécessite une interpolation du niveau supérieur. Par souci de simplicité et d'allègement des calculs la fonction d'interpolation considérée est une simple répétition de primitives. Les interactions se font de la manière suivante : un transfert de bloc entre deux niveaux donnés a lieu si l'énergie d'un bloc interpolé est plus faible que l'énergie du bloc correspondant au niveau inférieur.

Résultats expérimentaux

Ces deux algorithmes ont été comparés sur le même banc test que précédemment. Les courbes de la figure 3 représentent pour les 41 séquences, le rapport de l'énergie finale obtenue par les différents algorithmes à celle obtenue par la relaxation stochastique (RS). Une structure pyramidale à trois niveaux est utilisée et le jeu des températures pour l'algorithme MGD est : $T_0 = 80$; $T_1 = 15$; $T_2 = 0.01$. Dans cette version, la taille des blocs d'interactions entre niveaux croît en cours de relaxation. La convergence est supposée atteinte lorsque l'énergie ne décroît plus sur 10 itérations successives. La courbe 4 représente pour chacun des algorithmes le nombre d'itérations requis pour atteindre la convergence. A titre de comparaison la courbe obtenue pour les ICM monorésolution est aussi donnée. En moyenne l'algorithme basé sur les chaînes multiples (MGP) se comporte mieux que l'algorithme descendant (MGD) et que l'ICM monorésolution. La déviation par rapport à la relaxation stochastique est cependant importante (15% dans le meilleur des cas). Les mauvaises performances de la méthode multigrille s'explique principalement par la perte d'informations que le filtrage passe-bas des observations introduit aux niveaux de résolution grossiers. L'algorithme multiéchelle, que nous présentons maintenant, permet de pallier ce problème.

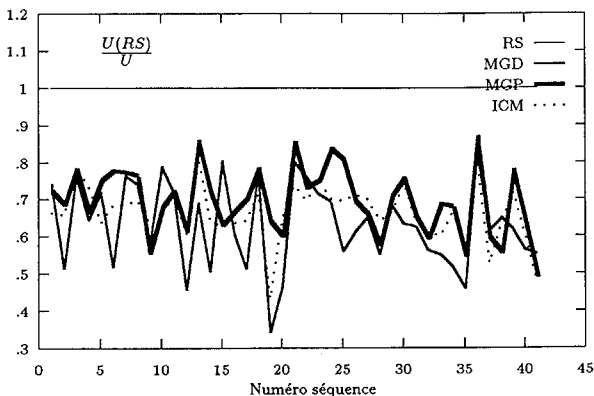


Figure 3 : Rapport des énergie finales

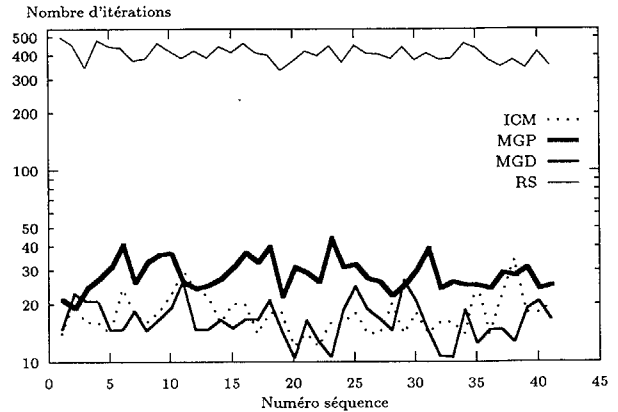


Figure 4 : Nombre d'itérations

3.3 Algorithme multiéchelle

Afin d'éviter la dégradation des observations le long de la pyramide, un nouveau schéma hiérarchique ne comportant qu'un seul niveau d'observation a été élaboré (Fig 5). Le modèle multiéchelle

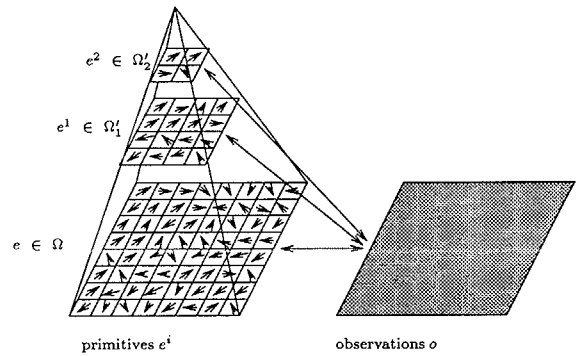


Figure 5 : Modèle multiéchelle

est décrit en détail, et justifié mathématiquement dans [11]. Il est basé sur une approche originale qui permet de définir un modèle markovien multirésolution globalement cohérent d'un point de vue mathématique. Le principe de l'approche multiéchelle est d'estimer le champ des primitives sur une succession de sous-espaces emboîtés $\Omega_n \subset \Omega_{n-1} \subset \dots \subset \Omega_0 = \Omega$. L'espace Ω_i correspond à un espace de configurations "grossières" constantes sur des blocs de taille $2^i \times 2^i$ (Fig. 6). On peut montrer, [11] que la résolution du problème d'optimisation dans le sous-espace Ω_i revient à résoudre le problème à une échelle réduite d'un facteur 2^i dans un nouvel espace Ω'_i . L'énergie à chaque niveau est déduite de celle au niveau le plus fin. Ce modèle a de multiples avantages par rapport à l'approche multigrille : il n'y a pas de dégradation des observations, il conduit à une représentation globalement cohérente et permet une visite efficace des configurations.

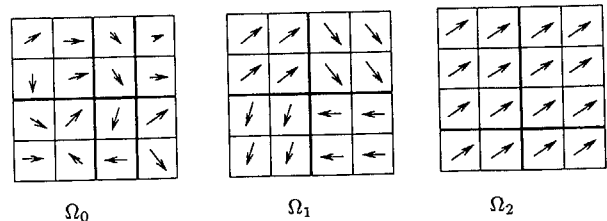


Figure 6 : exemple de configuration dans Ω_i

Comme précédemment deux algorithmes de relaxation sont évalués sur cette structure :

- un algorithme multiéchelle descendant (MED) dans lequel le champ des primitives estimé au niveau "grossier" est interpolé au prochain niveau et sert d'initialisation pour une nouvelle phase de relaxation (l'algorithme ICM est utilisé à chaque niveau).
- un algorithme (MEP) basé sur les chaînes de Markov multiples. Ici, cependant aucune interpolation n'est nécessaire, l'énergie à chaque niveau s'interprétant immédiatement au niveau le plus fin, [11].



Résultats expérimentaux

Pour le banc test, les figures 7 et 8 présentent respectivement les niveaux énergétiques atteints par les différents algorithmes ainsi que le nombre d'itérations requis dans chaque cas. La pyramide d'estimation a 4 niveaux et les températures associées à (MEP) sont : ($T_0 = 80$; $T_1 = 30$; $T_2 = 10$; $T_3 = 0.001$).

Les deux algorithmes construits sur cette structure possèdent des propriétés de convergence très satisfaisantes. Ils conduisent tous deux à des niveaux d'énergie finaux proches de la relaxation stochastique, avec un gain de 1 à 2 ordres de grandeur sur le nombre d'itérations. C'est cependant l'algorithme MED (non stochastique) qui converge le plus rapidement et c'est donc lui qui a été finalement retenu.

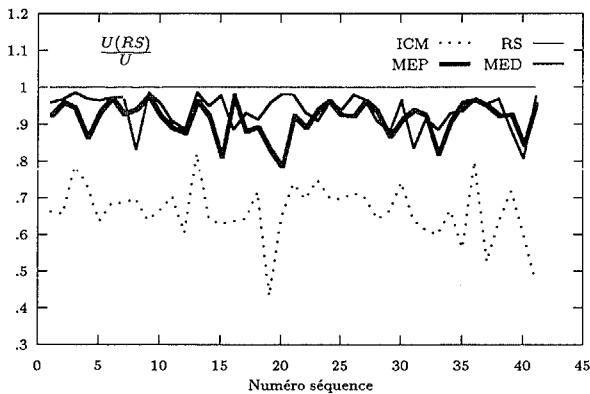


Figure 7 : Rapport des énergies finales

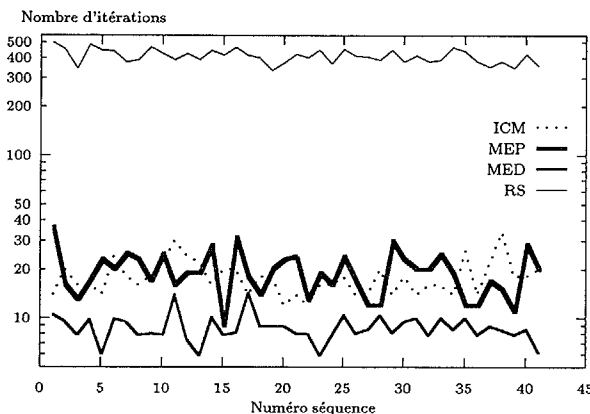


Figure 8 : Nombre d'itérations

4 Conclusion et perspectives

Trois schémas de parallélisations d'algorithmes de relaxation dans un contexte markovien ont été étudiés. La meilleure convergence est obtenue pour le schéma monorésolution avec une approche par chaînes multiples mais au prix d'un nombre d'itérations qui reste important. Dans le cas multiéchelle c'est le schéma de relaxation descendant qui se dégage nettement. Cet algorithme (MED) se prête de façon naturelle à un parallélisme massif SIMD par remise à jour simultanée des sites de l'image. L'implantation de l'algorithme multiéchelle sur machine parallèle, dans un problème d'estimation-segmentation au sens du mouvement est actuellement étudiée.

Bibliographie

- [1] E. AARTS, F. DE BONT, E. HABERS, and P. van LAARHOVEN. Parallel implementation of the statistical cooling algorithm. *Integration, the VLSI journal*, 4:pp. 209-238, 1986.
- [2] J. BESAG. On the statistical analysis of dirty pictures. *J. Royal Statist. Soc.*, Vol. 48, Serie B, No 3: pages 259-302, 1986.
- [3] P. BOUTHEMY and P. LALANDE. Detection and tracking of moving objects based on a statistical regularization method in space and time. In *Proc. First European Conference on Computer Vision*, pages 307-311, Springer, Antibes, France, April 1990.
- [4] F. DAREMA, S. KIRKPATRICK, and V. A. NORTON. Parallel techniques for chip placement by simulated annealing on shared memory systems. In *IEEE International Conference on Computer Design*, pages 87-90, 1987.
- [5] H. DERIN and C.S. WON. A parallel image segmentation algorithm using relaxation with varying neighborhoods and its mapping to array processor. *Comput. Vision, Graphics, Image Processing*, Vol. 40:pp 54-78, 1987.
- [6] E. FRANCOIS and P. BOUTHEMY. Multiframe-based identification of mobile components of a scene with a moving camera. In *IEEE Int. Conf. Computer Vision Pattern Recognition*, Hawaii, June 3-6 1991.
- [7] S. GEMAN and D. GEMAN. Stochastic relaxation, Gibbs distributions and the bayesian restoration of images. *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 6, No 6: pages 721-741, November 1984.
- [8] C. GRAFFIGNE. *A Parallel Simulated Annealing Algorithm*. Technical Report, CNRS, Université Paris-Sud, 1990.
- [9] F. HEITZ and P. BOUTHEMY. Multimodal motion estimation and segmentation using Markov random fields. In *Proc. 10th Int. Conf. Pattern recognition*, pages 378-383, Atlantic City, June 1990.
- [10] D.W. MURRAY, A. KASHKO, and H. BUXTON. A parallel approach to the picture restoration algorithm of Geman and Geman on an SIMD machine. *Image and Vision Computing*, vol.4(No.3):pp. 133-142, 1986.
- [11] P. PEREZ and F. HEITZ. Une approche multiéchelle à l'analyse d'images par champ markovien. In *8ième Congrès AFCET-RFIA*, Lyon, Nov. 1991.
- [12] TERZOPOULOS, D. Image analysis using multigrid relaxation methods. *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 8, No 2: pages 129-139, March 1986.
- [13] J. ZERUBIA and F. PLOYETTE. Parallel algorithms for low-level vision on a Connection Machine cm2. In *Proc. 13th Imacs World Congress on Computation and Applied Mathematics*, Dublin, July 1991.