



# Schémas numériques de résolution du flot de vecteur gradient

Djamal Boukerroui

## ► To cite this version:

Djamal Boukerroui. Schémas numériques de résolution du flot de vecteur gradient. TRAITEMENT ET ANALYSE DE L'INFORMATION: Méthodes et Applications, May 2009, Hammamet, Tunisie. pp.523-532. <hal-00430277>

**HAL Id: hal-00430277**

**<https://hal.science/hal-00430277v1>**

Submitted on 6 Nov 2009

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

# Schémas numériques de résolution du flot de vecteur gradient

Djamal Boukerroui

Université de Technologie de Compiègne  
CNRS UMR 6599 Heudiasyc  
BP 20529 - 60205 Compiègne Cedex, France.  
`djamal.boukerroui@hds.utc.fr`

**Résumé** Depuis sa publication il y a une décennie, le “Gradient Vector Flow” est utilisé dans de nombreuses applications. Sa popularité est certainement due à son efficacité à produire un champ de force qui permet de pallier aux problèmes de convergence connus des contours actifs. Le principal inconvénient de l’approche GVF et sa généralisation est leur coût important en temps de calcul, et sa conséquence sur la zone d’influence. Nous proposons et nous comparons dans ce travail, différents schémas numériques efficaces pour résoudre le problème du GVF et ses généralisations.

**Mots clés :** contours actifs, flot de vecteur gradient (GVF), schémas numériques.

## 1 Introduction

La segmentation d’images en utilisant les modèles déformables se ramène à la minimisation d’une fonctionnelle d’énergie à deux termes (interne et externe)[9]. L’énergie interne modélise notre connaissance *a priori* en terme de régularité de la solution recherchée. L’énergie externe dépend des données image et sert à attirer le modèle déformable aux caractéristiques désirées, telles les frontières de l’objet d’intérêt. Traditionnellement, l’énergie externe est basée uniquement sur l’information gradient de l’image  $I$  et est définie comme  $-f(\cdot)$ , où  $f(\cdot)$  est une carte de contours de l’image  $I(\cdot)$ , habituellement définie comme étant la norme du gradient d’une version filtrée de l’image originale par un filtre gaussien d’écart type  $\sigma$  ; c’est-à-dire  $|\nabla G_\sigma(\cdot) * I(\cdot)|^2$ . La valeur de  $\sigma$  permet le contrôle de la zone d’influence du contour, dite également zone d’attractivité. Les vecteurs gradients de la carte de contour pointent vers les frontières et sont normaux aux points de discontinuités de l’image. Le champ de vecteurs  $\nabla f$  est le champ de force externe responsable de l’attraction du modèle déformable vers les discontinuités désirées. Cependant, cette force est importante seulement au voisinage immédiat de la frontière. De tels modèles rencontrent de nombreuses difficultés, parmi lesquelles leur sensibilité à l’initialisation, en raison de l’étroite zone d’influence de la force externe, et surtout leur difficulté à converger vers les concavités des frontières. Cela impose donc une initialisation du modèle très proche de la solution.

Plusieurs chercheurs ont souligné ces limitations et ont proposé des solutions alternatives afin d’accroître la zone d’influence du modèle et de réduire ainsi sa sensibilité à l’initialisation (voir par ex. [3,4,10]). De récents travaux proposent également des solutions au problème de convergence vers des frontières concaves. Nous citerons, par exemple, le

modèle de contour actif basé sur les forces de charges électrostatiques (CACE)[18] et le modèle de convolution avec un champ de vecteur (Vector Field Convolution)[11]. Dans ce dernier, la force externe est obtenue via une simple convolution de la carte de contours avec un noyau de convolution vectoriel défini par l'utilisateur. La simplicité de la méthode combinée avec sa robustesse au bruit fait du VFC une technique très prometteuse.

La première et la plus populaire des solutions proposées dans la littérature est toutefois le champ de flot de vecteur gradient ou 'Gradient Vector Flow' (GVF) [17] et sa généralisation, le GGVF [16]. Depuis sa publication il y a une décennie, le GVF a été utilisé et adapté à différents modèles et problématiques relevant du traitement d'images. Nous citerons à titre d'exemples, la segmentation [17,12,14], le suivi [13], la segmentation en utilisant des relations spatiales floues [5] et la squelettisation [8]. La popularité du GVF est certainement due à son efficacité. Le principal inconvénient de l'approche et de sa généralisation est son coût en temps de calcul, et sa conséquence sur la zone d'attractivité. En effet, traditionnellement, le GVF est obtenu via la minimisation itérative d'une fonctionnelle. Ainsi, la zone d'influence du flot résultant dépend bien sûr du nombre d'itérations utilisées. Très récemment, les techniques de résolution multigrille ont été utilisées avec succès pour résoudre le problème du GVF et ses généralisations [7]. Dans ce travail, nous proposons et comparons l'efficacité d'autres schémas numériques alternatifs.

## 2 Formulation du GVF

L'approche du flot de vecteur gradient permet l'extension de la zone d'influence de la carte de contour en utilisant un processus de diffusion. Le flot est défini comme le champ de vecteurs  $\mathbf{v}(x) : \Omega \rightarrow \mathbb{R}^d$  qui minimise la fonctionnelle suivante [16] :

$$\mathcal{E} = \int_{\Omega} g(\mathbf{x}) |\nabla \mathbf{v}|^2 + h(\mathbf{x}) |\mathbf{v} - \nabla f|^2 d\mathbf{x} , \quad (1)$$

$g(\mathbf{x})$  et  $h(\mathbf{x})$  sont des fonctions à valeurs positives. L'idée de base de cette formulation variationnelle est d'imposer une certaine régularité à la solution dans les régions lointaines aux données (premier terme), et d'être très proche du champ initial où des mesures fiables sont disponibles (second terme). Le calcul des variations du problème de minimisation ci-dessus nous conduit à exprimer la solution via l'équation d'évolution suivante [16]

$$\partial_t \mathbf{v} = \nabla \cdot (g(\mathbf{x}) \nabla \mathbf{v}) - h(\mathbf{x}) (\mathbf{v} - \nabla f) , \quad (2)$$

avec des conditions aux limites de type Neumann, et  $\mathbf{v}(\mathbf{x}, 0) = \nabla f$  comme condition initiale. Selon le choix des deux fonctions de pondérations  $g(\mathbf{x})$  et  $h(\mathbf{x})$ , Xu et Prince [16] ont proposé deux champs de flot de vecteur gradient :

**a) Le flot de vecteur gradient (GVF) :** Il est défini pour le cas où  $g(\mathbf{x}) = \mu$  et  $h(\mathbf{x}) = |\nabla f|^2$ .  $\mu$  est une constante positive qui permet le contrôle du terme de régularisation dans (1). Pour ce cas particulier, l'équation d'évolution (2) se simplifie à

$$\partial_t \mathbf{v} = \mu \nabla^2 \mathbf{v} - h(|\nabla f|) (\mathbf{v} - \nabla f) , \quad (3)$$

où  $\nabla^2$  est l'opérateur laplacien.

**b) Le flot de vecteur gradient généralisé (GGVF) :** Afin d'éviter un lissage excessif dans le voisinage des discontinuités détectées, et de renforcer la régularisation dans les régions lointaines, les choix suivants sont proposés par [16] :

$$\mathbf{g}(\mathbf{x}) = \exp \left\{ -\frac{|\nabla f|}{K} \right\} \quad , \quad \mathbf{h}(\mathbf{x}) = 1 - \mathbf{g}(|\nabla f|) \quad . \quad (4)$$

Xu et Prince [16] ont, cependant, décidé d'approcher l'équation d'évolution (2) en ignorant les variations de  $\mathbf{g}$ . Ils ont défini le **GGVF** comme la solution de

$$\partial_t \mathbf{v} = \mathbf{g}(|\nabla f|) \nabla^2 \mathbf{v} - \mathbf{h}(|\nabla f|) (\mathbf{v} - \nabla f) \quad . \quad (5)$$

Nous noterons **TGGVF**, pour 'True Generalized Gradient Vector Flow', la solution de (2) en utilisant les fonctions de pondération définie dans (4). Dans la suite de cet article, nous nous limiterons aux domaines bidimensionnels ( $d = 2$ ) et nous n'explicitons les relations que pour la composante  $u$  dans la direction  $x$  du champ  $\mathbf{v}$ . A partir de (5), nous obtenons

$$\partial_t u = \mathbf{g} \nabla^2 u - \mathbf{h} u + \mathbf{h}^1, \quad \text{avec} \quad \mathbf{h}^1 = \mathbf{h} \partial_x f \quad . \quad (6)$$

### 3 Résolution numérique

Notons par  $u_{i,j}^k$  la version discrète de  $u(i\Delta x, j\Delta y, k\tau)$  où  $\tau$  est le pas en temps,  $\Delta x$  et  $\Delta y$  sont les pas de discrétisation du domaine spatial. Il est naturel de supposer que le maillage discret est isotrope (i.e  $\Delta x = \Delta y = h$ ). Les différences finies sont généralement utilisées dans ce contexte pour l'approximation des différentes dérivées partielles, ce qui nous conduit à résoudre un système d'équations discrètes.

**a) Schéma explicite :** Le schéma explicite du problème du **GGVF** est donné par [17] :

$$u_{i,j}^{k+1} = \mathbf{r}_{i,j} (u_{i+1,j}^k + u_{i,j+1}^k + u_{i-1,j}^k + u_{i,j-1}^k - 4u_{i,j}^k) + (1 - \tau \mathbf{h}_{i,j}) u_{i,j}^k + \tau \mathbf{h}_{i,j}^1, \quad (7)$$

avec  $\mathbf{r} = \tau h^{-2} \mathbf{g}$ . C'est un schéma explicite car il donne une formule explicite de calcul de la solution au temps  $t_{k+1}$  en fonction des valeurs de la solution au temps précédent. Il n'y a pas d'équation à résoudre pour obtenir la valeur  $u^{k+1}$  au nouvel instant  $t_{k+1}$ . L'étude de stabilité de ce schéma nous conduit à la condition suivante (§ App. A)<sup>†</sup> :

$$\tau \leq \frac{2h^2}{\max_{i,j} (h^2 \mathbf{h}_{i,j} + 8\mathbf{g}_{i,j})} \quad . \quad (8)$$

Les schémas explicites sont très populaires pour la résolution des équations aux dérivées partielles, principalement en raison de leur simplicité. Cette simplicité, en contre partie, impose des pas en temps très petits, ce qui conduit à des coûts importants en terme de temps de calcul. Les schémas explicites sont plutôt limités par leur stabilité que par leur précision [15]. Par conséquent d'autres schémas, semi-implicites ou implicites sont très souhaitables car ils restent numériquement stables pour des pas en temps importants, ce qui permet des gains non négligeables de temps de calcul. De tels schémas sont donc

à considérer avec intérêt lorsque la charge de calcul en plus est réduite par le biais de l'utilisation de méthodes numériques appropriées (voir [1,15]).

**b) Schéma semi-implicite :** Le schéma implicite de l'équation (6) est donné par :

$$\frac{u_{i,j}^{k+1} - u_{i,j}^k}{\tau} = \frac{g_{i,j}}{h^2} (u_{i-1,j}^{k+1} + u_{i+1,j}^{k+1} + u_{i,j-1}^{k+1} + u_{i,j+1}^{k+1} - 4u_{i,j}^{k+1}) - \mathbf{h}_{i,j} u_{i,j}^{k+1} + \mathbf{h}_{i,j}^1. \quad (9)$$

Ce schéma est dit implicite car le calcul de la solution au temps  $t_{k+1}$  nécessite la résolution d'un grand système linéaire avec une matrice très creuse. Il est inconditionnellement stable (voir App. A). La difficulté de résolution augmente avec le nombre de dimensions du domaine  $\Omega$ . Nous proposons l'étude du schéma semi-implicite suivant :

$$\frac{u_{i,j}^{k+1} - u_{i,j}^k}{\tau} = \frac{g_{i,j}}{h^2} (u_{i-1,j}^{k+1} + u_{i+1,j}^{k+1} + u_{i,j-1}^{k+1} + u_{i,j+1}^{k+1} - 4u_{i,j}^{k+1}) - \mathbf{h}_{i,j} u_{i,j}^k + \mathbf{h}_{i,j}^1, \quad (10)$$

où un seul terme à droite est au temps  $t_k$ . Ce schéma est stable pour  $\tau \leq 2/\mathbf{h}_{max}$  (voir App. A). Ici aussi, nous avons besoin de résoudre un système linéaire pour obtenir la solution à l'itération  $k+1$ . Notons cependant que pour le cas particulier du GVF, la matrice du système est indépendante des données du problème (c'est-à-dire lorsque  $g_{i,j} = \mu$ ). La transformée en cosinus discrète (DCT) peut donc être utilisée pour résoudre efficacement ce schéma (La DCT respecte les conditions aux limites de type Newmann). Sans perte de généralité, considérons avec un pas de discrétisation spatiale  $h = 1$ , l'application de la DCT à l'équation (10), qui nous conduit à :

$$\hat{u}_{m,n}^{k+1} = \frac{\hat{d}_{m,n}^k}{1 + 4\mu\tau \left( \sin^2 \left( \frac{\pi m}{N_1} \right) + \sin^2 \left( \frac{\pi n}{N_2} \right) \right)}, \quad (11)$$

avec  $\hat{d}_{m,n}^k = \text{DCT} \left\{ (1 - \tau \mathbf{h}_{i,j}) u_{i,j}^k + \tau \mathbf{h}_{i,j}^1 \right\}$  et  $(m, n) \in [0, N_1 - 1] \times [0, N_2 - 1]$ . Ainsi, le calcul du champ GVF en utilisant ce schéma nécessite une transformée en cosinus directe et une transformée inverse pour chaque composante du champ par itération.

**c) Schéma des directions alternées explicite :** Le schéma des directions alternées explicite ou ADE pour 'Alternating Direction Explicit' est un schéma à deux pas et il est connu pour être inconditionnellement stable. Les deux étapes du schéma numérique de l'équation d'évolution (6) sont données par

$$u_{i,j}^{k+1} - \frac{\mathbf{r}_{i,j} (u_{i-1,j}^{k+1} + u_{i,j-1}^{k+1})}{1 + 2\mathbf{r}_{i,j} + \tau \mathbf{h}_{i,j}} = \frac{(1 - 2\mathbf{r}_{i,j}) u_{i,j}^k}{1 + 2\mathbf{r}_{i,j} + \tau \mathbf{h}_{i,j}} + \frac{\mathbf{r}_{i,j} (u_{i+1,j}^k + u_{i,j+1}^k)}{1 + 2\mathbf{r}_{i,j} + \tau \mathbf{h}_{i,j}} + \frac{\tau \mathbf{h}_{i,j}^1}{1 + 2\mathbf{r}_{i,j} + \tau \mathbf{h}_{i,j}}, \quad (12)$$

$$u_{i,j}^{k+2} - \frac{\mathbf{r}_{i,j} (u_{i+1,j}^{k+2} + u_{i,j+1}^{k+2})}{1 + 2\mathbf{r}_{i,j} + \tau \mathbf{h}_{i,j}} = \frac{(1 - 2\mathbf{r}_{i,j}) u_{i,j}^{k+1}}{1 + 2\mathbf{r}_{i,j} + \tau \mathbf{h}_{i,j}} + \frac{\mathbf{r}_{i,j} (u_{i-1,j}^{k+1} + u_{i,j-1}^{k+1})}{1 + 2\mathbf{r}_{i,j} + \tau \mathbf{h}_{i,j}} + \frac{\tau \mathbf{h}_{i,j}^1}{1 + 2\mathbf{r}_{i,j} + \tau \mathbf{h}_{i,j}}. \quad (13)$$

Il est simple de voir que ce schéma est explicite. En effet, le calcul dans l'équation (12) se fait dans l'ordre lexicographique, et l'équation (13) procède dans le sens inverse. Cependant cette méthode souffre d'anisotropie pour des pas en temps très importants. Afin

---

<sup>†</sup> Notre condition de stabilité est plus restrictive que celle de Xu et Prince  $\tau \leq \Delta x \Delta y / 4\mathbf{g}_{max}$  [16,17]. Nous supposons une erreur dans leur dérivation. Heureusement les deux conditions sont équivalentes pour le **GGVF** dans le cas particulier des fonctions de pondération données en équation (4) pour  $h = 1$ .

de réduire cet effet, un retournement gauche-droite du domaine est recommandé entre les itérations, (i.e une itération procède dans le sens de la diagonale d'équation  $y = x$  et la suivante dans le sens de la diagonale  $y = -x$ ). Cette variante sera dénommée ADES.

**d) Méthodes d'approximation par factorisation :** Notons que le terme de diffusion dans l'équation de l'évolution (2), prend exactement la même forme que celui de la diffusion non-linéaire [15, eq. 1]. Les méthodes d'approximation par factorisation ont été utilisées avec succès pour résoudre ce problème. Dans notre cas, il y a un terme en plus qui, nous verrons, ne nous posera guère de problème. Pour une meilleure lisibilité nous réécrivons le schéma implicite de (2), donné en (9), sous sa forme matricielle, donnée par :

$$(\mathbf{I} - \tau \mathbf{A}_x - \tau \mathbf{A}_y) u^{k+1} = u^k + \tau \mathbf{h}^1, \quad (14)$$

où  $\mathbf{I}$  est la matrice identité et  $u \in \mathbb{R}^2$  est représenté comme un vecteur. Les matrices  $\mathbf{A}_l$  correspondent aux discrétisations des dérivées secondes dans la direction  $l$ , c'est-à-dire que le produit matrice-vecteur  $\mathbf{A}_l u$  est l'approximation discrète de l'opérateur  $\partial_{x_l}(\mathbf{g}(\mathbf{x})\partial_{x_l}u) - \frac{1}{d}\mathbf{h}(\mathbf{x})u$ . Ainsi, les éléments  $\mathbf{A}_l$  sont donnés explicitement par [15,2]

$$(a_l)_{ij} = \begin{cases} \frac{\mathbf{g}_j + \mathbf{g}_i}{2h^2} & (j \in \mathcal{N}(i)), \\ -\frac{\mathbf{h}_i}{d} - \sum_{n \in \mathcal{N}(i)} \frac{\mathbf{g}_i + \mathbf{g}_n}{2h^2} & (j = i), \\ 0 & (\text{sinon}), \end{cases} \quad (15)$$

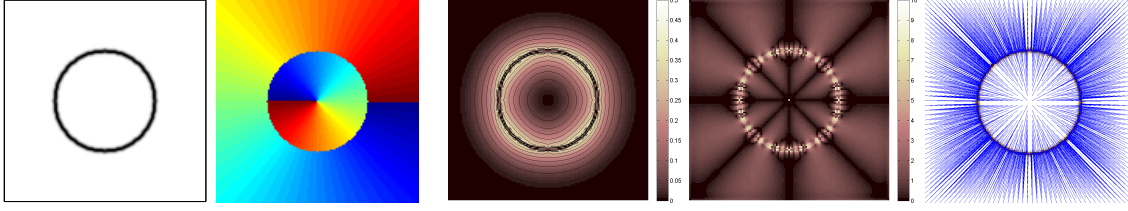
où  $\mathcal{N}(i)$  est l'ensemble des deux voisins du pixel  $i$  le long de axe  $l$ . Barash et al. [2] ont proposé et comparé différents opérateurs d'approximation par factorisation pour la résolution de la diffusion non linéaire qui pourraient être utilisés pour résoudre (14) : la méthode de factorisation additive AOS (pour 'Additive Operator Splitting'), multiplicative LOD (pour 'Locally One Dimensional'), et combinée AMOS (pour 'Additif Multiplicative Operator Splitting'), données respectivement par :

$$u^{k+1} = \frac{1}{2} \sum_{l=1}^2 (\mathbf{I} - 2\tau \mathbf{A}_l)^{-1} (u^k + \tau \mathbf{h}^1), \quad (16)$$

$$u^{k+1} = \prod_{l=1}^2 (\mathbf{I} - \tau \mathbf{A}_l)^{-1} (u^k + \tau \mathbf{h}^1), \quad (17)$$

$$u^{k+1} = \frac{1}{2} \left[ (\mathbf{I} - \tau \mathbf{A}_1)^{-1} (\mathbf{I} - \tau \mathbf{A}_2)^{-1} + (\mathbf{I} - \tau \mathbf{A}_2)^{-1} (\mathbf{I} - \tau \mathbf{A}_1)^{-1} \right] (u^k + \tau \mathbf{h}^1). \quad (18)$$

L'approximation LOD est connue pour être la plus efficace, mais elle n'est pas symétrique. La méthode AOS a l'avantage d'être symétrique et la méthode AMOS, comme son nom l'indique, est une combinaison des deux précédentes. Toutefois toutes ces approximations ne tiennent pas en compte de l'erreur de factorisation (c'est-à-dire la différence entre le vrai opérateur dans (14) et l'approximation). Testées sur notre problème, toutes ont données de bons résultats pour des petits pas en temps. Les résultats se détériorent considérablement pour des pas importants. Un calcul simple de l'erreur de factorisation de la méthode LOD nous donne :  $E_{AF} = \tau^2 \mathbf{A}_1 \mathbf{A}_2 u^{k+1}$ . Ce terme est d'ordre 2 en temps et



**Fig. 1.** De gauche à droite : Image de la carte de contour. Image des orientations théoriques du champ de vecteurs du flot. Amplitude du champ des vecteurs du flot obtenu après 300 itération du schéma explicite avec  $\tau = .2$ , les lignes de niveau sont également montrées. Image de l'erreur angulaire absolue du résultat du schéma explicite en degré. Exemple des lignes de trajectoires du champ de flot de vecteur gradient.

comprend des dérivées d'ordre 4 mixtes de la solution à l'étape  $k+1$ . Cela explique l'une de nos observations sur le comportement de l'erreur du schéma LOD. Cette approximation a tendance à manquer de précision dans le voisinage de la carte de contours. Comme alternative, nous proposons l'approximation suivante :

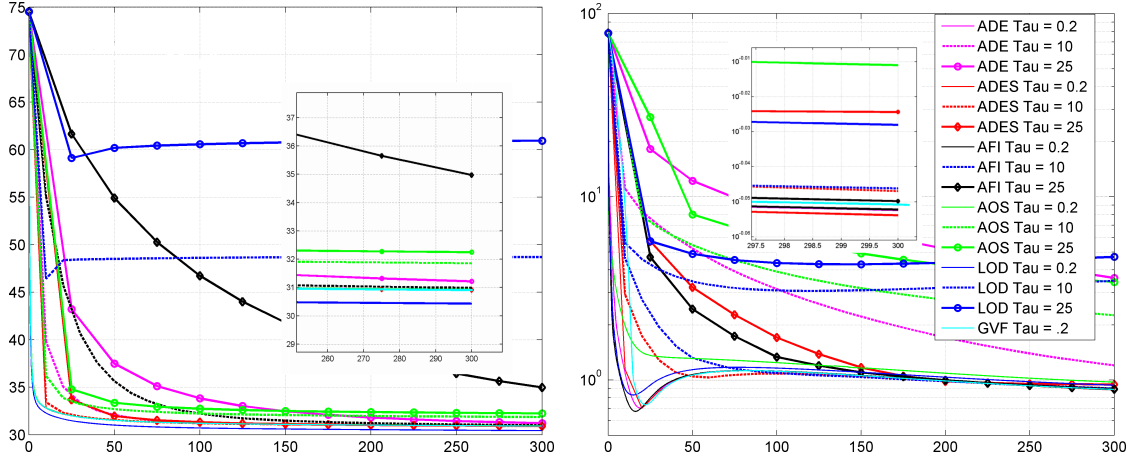
$$\begin{aligned} (\mathbf{I} - \tau \mathbf{A}_1)u^* &= (\mathbf{I} + \tau \mathbf{A}_2)(u^k + \tau \mathbf{h}^1) \\ (\mathbf{I} - \tau \mathbf{A}_2)u^{k+1} &= u^* - \tau \mathbf{A}_2 u^k \end{aligned} \quad (19)$$

L'erreur de factorisation de cette approximation est  $E_{AF} = \tau^2 \mathbf{A}_1 \mathbf{A}_2 (u^k - u^{k+1})$ . Contrairement à l'approximation multiplicative LOD, elle dépend du résidu et non pas de la solution à l'itération  $k+1$ . Par conséquent, on peut s'attendre à la voir diminuer à mesure que la méthode converge. Le schéma ci-dessus est noté AFI (pour 'Approximation Factorisation Implicit'). Compte tenu de la forme des matrices  $\mathbf{A}_i$ , toutes les approximations ci-dessus peuvent être résolues efficacement par l'algorithme de Thomas [15]. Il est à noter que les schémas itératifs ci-dessus résolvent l'équation du TGGVF.

#### 4 Résultats et discussion

Nous avons présenté plusieurs schémas numériques alternatifs au schéma de base explicite, qui tous permettent l'utilisation d'un plus grand pas en temps. Tous ces schémas sont du second ordre en espace et sont du premier ordre en temps, et sont inconditionnellement stable (à l'exception du semi-implicite). Toutes les méthodes ont une complexité  $O(N^2)$  à l'exception du schéma semi-implicite utilisant la DCT où la complexité est de l'ordre  $O(N^2 \log_2 N)$  pour une image de taille  $N \times N$ . Le temps de calcul pour une itération du schéma ADE est approximativement le même que pour le schéma explicite, et il est de 2 à 3 fois plus important pour les schémas ADES, LOD, AOS et AFI. La méthode AMOS a besoin en revanche d'un effort 5 fois plus important. Ces résultats sont basés sur une moyenne de 600 itérations et pour 8 dimensions différentes d'image. Cela impose donc un minimum de pas en temps entre 0,25 et 1,5 pour que les alternatives proposées soient plus rapides que l'explicite. Il est à retenir cependant qu'un pas en temps très grand introduit une perte en précision, et dans certaines approximations une perte d'isotropie. Dans ce qui suit, nous proposons un moyen de quantifier cette perte de précision lorsque des pas en temps importants sont utilisés.

Le schéma utilisant la DCT ayant une complexité plus importante et une condition de stabilité plus restrictive, généralement un pas en temps max de 3 à 5, le gain en temps



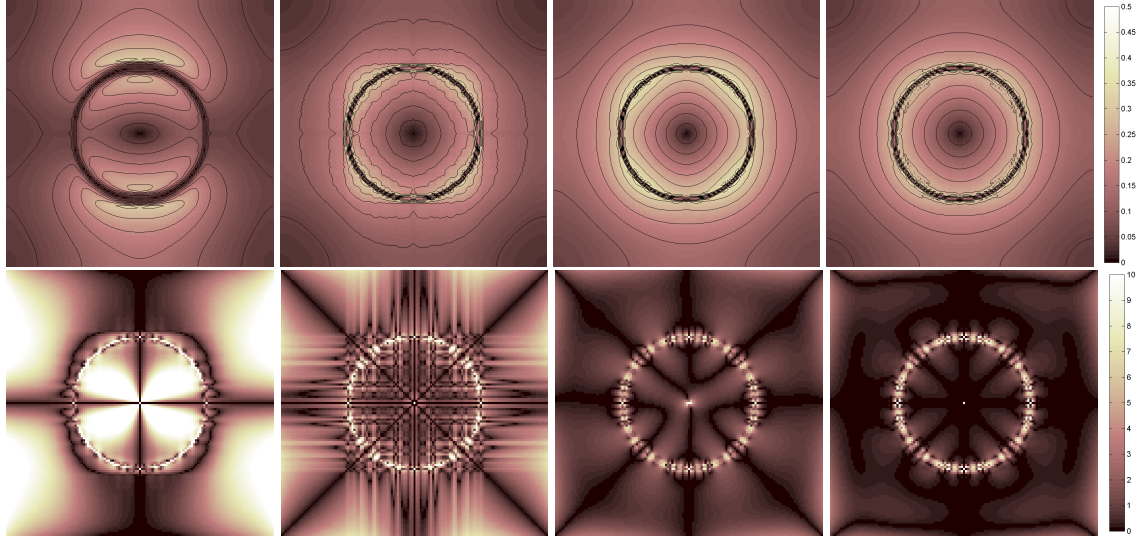
**Fig. 2.** (à gauche) Évolution de la fonction d'énergie du GGVF (1) en fonction du temps pour les différents schémas ; (à droite) Évolution de l'erreur absolue angulaire moyenne en degré en fonction du temps. Les petites images représentent un zoom autour de  $t = 300$ . Le terme GVF désigne le schéma explicite.

de calcul est moindre (de 2 à 3 par rapport au schéma explicite). Il convient de souligner toutefois que cette méthode peut être très intéressante dans la mesure où une approximation d'ordre supérieur est nécessaire pour la discrétisation du laplacien. Nous avons inclut ce schéma dans un souci d'exhaustivité, mais nous ne le discuterons pas plus amplement. Aussi, nous ne présenterons pas nos tests du schéma AMOS car son comportement peut-être prédit de celui du LOD et AOS, mais aussi pour son coût important.

**Critères de comparaison :** Lors de la mise au point d'une méthode de quantification de la précision des différents schémas numériques, nous devons également garder à l'esprit notre objectif final quant à l'utilisation du GVF. En effet, généralement, dans la plupart des utilisations du flot de vecteur gradient, l'information la plus importante est l'orientation des vecteurs du flot. L'amplitude est rarement utilisée dans des applications de segmentation par exemple. Pour ce faire, nous avons synthétisé une carte de contours d'un cercle parfait pour lequel nous disposons d'une vérité terrain de l'orientation théorique du champ de vecteurs en chaque point de l'image (voir figure 1). Deux critères sont alors utilisés pour la comparaison des différentes méthodes : l'erreur angulaire absolue moyenne estimée du champ de vecteurs par rapport à la vérité terrain, et la valeur de la fonction d'énergie  $\mathcal{E}$  donnée par l'équation (1). Les résultats pour différentes valeurs de pas en temps  $\tau$  sont présentés sur la figure 2.

**Comparaison :** La première observation est le comportement prévisible de toutes les méthodes pour de très faibles pas en temps. Nous observons que les résultats de toutes les méthodes sont très proches de ceux du schéma explicite ; cela permet en partie de vérifier notre implémentation. Une observation importante est que la mesure proposée est très informative vu que son comportement n'est pas toujours corrélé avec celui de l'énergie à minimiser. Nous remarquons que les courbes de l'erreur angulaire sont de l'ordre de  $1^\circ$  pour des pas en temps  $\tau$  inférieur à 10, à l'exception des schémas ADE, LOD et AOS. Les plus grands écarts sont obtenus pour les deux approximations LOD et AOS (la plus grande erreur approche les  $5^\circ$  à  $t = 300$  pour la méthode LOD avec  $\tau = 25$ ). Cela confirme





**Fig. 3.** 1<sup>ère</sup> ligne : Image de l'amplitude du champ du flot de vecteur gradient ; Les courbes en noir représentent les lignes de niveau. 2<sup>ème</sup> ligne : Image de l'erreur angulaire absolue en degré. Respectivement partant de la colonne de gauche, les schémas LOD, AOS, ADES et AFI. Tous les résultats sont obtenus pour  $\tau = 25$  en 20 itérations. L'échelle à droite est commune à toutes les images.

les observations faites par Papandreou et Maragos lors de l'utilisation du schéma AOS dans un modèle de contour actif [6].

Remarquons que les résultats des schémas AFI et ADES sont très satisfaisants. L'erreur sur l'orientation des vecteurs gradients est du même ordre que la méthode explicite pour  $t > 50$  avec  $\tau = 10$ , et pour  $t > 100$  pour  $\tau = 25$  (autrement dit seulement au bout de 4 à 5 itérations). Comme nous l'avons souligné dans l'analyse de l'erreur de factorisation, la méthode AFI améliore considérablement les performances des méthodes d'approximation. Elle présente cependant une baisse du taux de convergence, comme il est constaté sur la courbe d'énergie. Afin de compléter notre analyse, la figure (3) présente les images de l'amplitude des champs obtenus. Cela nous donne une indication de l'étendue de la zone d'attractivité. Les lignes de niveau sont superposées pour indiquer la direction des vecteurs. Les images de l'erreur angulaire absolue y figurent également. Des images équivalentes pour le schéma explicite sont données en figure (1). Notons la dissymétrie du LOD et l'imprécision de l'AOS. En revanche les deux schémas proposés donnent des résultats très satisfaisants. Ces résultats nous permettent d'avancer que le schéma ADES est probablement le mieux approprié.

Nous avons également mis en œuvre la technique de résolution multigrille proposée dans [7]. L'application du schéma FMG-GS(1,2,2), suggéré par les auteurs, à notre problème donne une erreur angulaire de  $1.9^\circ$ . La méthode ADES produit une erreur inférieure à  $1,9^\circ$  après 4 itérations avec un pas en temps  $\tau = 15$ . Les temps de calcul des deux méthodes sont approximativement les mêmes avec notre mise en œuvre. Nos expériences suggèrent toutefois l'utilisation d'environ 10 itérations. D'autres expériences devraient être menées pour une comparaison plus rigoureuse des deux méthodes. Nos résultats, certes préliminaires, suggèrent cependant que le schéma proposé est une alternative intéressante.

## 5 Conclusion

Depuis son introduction, le flot de vecteur gradient est devenu très populaire et est largement utilisé dans de nombreuses applications. Le principal inconvénient de l'approche GVF et de sa généralisation est son coût important en temps de calcul. Nous avons proposé et comparé dans ce travail différents schémas numériques efficaces pour résoudre cette problématique. Nous avons montré que le schéma ADES peut être une bonne alternative aux techniques multigrille. Nos résultats montrent également que notre proposition d'approximation par factorisation AFI est largement meilleure que les autres méthodes de factorisation connues dans la littérature. Enfin, il est à noter que le schéma ADE et ses variantes sont peu connus par la communauté de traitement d'image. Leur utilisation pour résoudre d'autres problèmes d'EDP de traitement d'image est certainement d'intérêt.

**Remerciement :** Ce travail de recherche est partiellement financé par la région Picardie dans le cadre du projet de Recherche Régional et Structurant 2008 (ECHOPEDIA). L'auteur remercie également JF. Lerallut pour sa relecture et ses nombreuses corrections.

### A Stabilité d'un schéma numérique

La condition minimale de stabilité impose que le vecteur de composantes  $u_{i,j}^k$  reste borné pour tout  $k \in [0, K]$ . Une technique simple de calcul de la stabilité d'un schéma est donnée dans le cas de problèmes linéaires par l'analyse de Fourier (voir, par exemple [1]). Cette approche étudie le comportement du schéma pour un mode de Fourier discret. Dans notre cas, donné par  $u_{i,j}^k = \xi^k \exp\{i\pi h(im + jn)\}$ , où l'indice  $k$  sur  $\xi$  est un exposant multiplicatif. Une condition nécessaire et suffisante de stabilité est obtenue par la restriction de  $h$  et  $\tau$  de telle sorte que  $|\xi| \leq 1$ .

a) **Schéma explicite :** L'étude de stabilité du schéma (7) nous conduit à :

$$\xi = [1 - \tau \mathbf{h}_{i,j} - 4\mathbf{r}_{i,j} + 2\mathbf{r}_{i,j}(\cos(\pi m h) + \cos(\pi n h))] .$$

Posons  $\gamma = \cos(\pi m h) + \cos(\pi n h)$  et remplaçons  $\mathbf{r}_{i,j}$  par sa valeur  $\tau h^{-2} \mathbf{g}_{i,j}$ , et après quelques manipulations algébriques, nous arrivons à la condition suivante

$$0 \leq \tau \left( \mathbf{h}_{i,j} + \frac{2\mathbf{g}_{i,j}}{h^2} (2 - \gamma) \right) \leq \tau \left( \mathbf{h}_{i,j} + \frac{8\mathbf{g}_{i,j}}{h^2} \right) \leq 2 .$$

La condition de stabilité du schéma explicite est alors donné par :

$$\tau \leq \frac{2h^2}{\max_{i,j} (h^2 \mathbf{h}_{i,j} + 8\mathbf{g}_{i,j})} \quad (\text{A.1})$$

b) **Schéma semi-implicite :** En utilisant la même procédure, la condition de stabilité pour le schéma donné en (10) implique

$$\left| \frac{1 - \tau \mathbf{h}_{i,j}}{1 + 2\mathbf{r}_{i,j}(2 - \gamma)} \right| \leq 1 , \quad (\text{A.2})$$

où  $\gamma$  est comme ci-dessus. Connaissant que  $(2 - \gamma) \geq 0$  et que  $\mathbf{r}_{i,j} \geq 0$ , il est simple de vérifier que le dénominateur est toujours  $\geq 1$ . En plus  $\mathbf{h}_{i,j} \geq 0$ , ce qui induit

$$\tau \leq \frac{2}{\mathbf{h}_{\max}} \quad \text{avec} \quad \mathbf{h}_{\max} = \max_{i,j} \mathbf{h}_{i,j}. \quad (\text{A.3})$$

c) **Schéma implicite** : pour le schéma implicite donné en (9) nous obtenons

$$\xi = \frac{1}{1 + \mathbf{h}_{i,j}\tau + 2\mathbf{r}_{i,j}(2 - \gamma)} , \quad (\text{A.4})$$

où  $\gamma$  est comme ci-dessus. La condition  $|\xi| \leq 1$  donne :  $|1 + \mathbf{h}_{i,j}\tau + 2\mathbf{r}_{i,j}(2 - \gamma)| \geq 1$  qui est toujours vrai car  $(2 - \gamma) \geq 0$ ,  $\mathbf{r}_{i,j} \geq 0$  et  $\mathbf{h}_{i,j} \geq 0$ . Donc le schéma implicite est inconditionnellement stable.

## Références

1. G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing*, volume 147 of *Applied Mathematical Sciences*. Springer Verlag, November 2001.
2. D. Barash, T. Schlick, M. Israeli, and R. Kimmel. Multiplicative operator splitting in nonlinear diffusion : from spatial splitting to multiple timesteps. *Journal of Mathematical Imaging and Vision*, 19(1) :33 – 48, March 2003.
3. L.D. Cohen. On active contour models and balloons. *CVGIP : Image Understanding*, 53 :211–218, 1991.
4. L.D. Cohen and I. Cohen. Finite element methods for active contour models and balloons for 2D and 3D images. *IEEE Trans. PAMI*, 15(11) :1131–1147, 1993.
5. O. Colliot, O. Camara, and I. Bloch. Integration of fuzzy spatial relations in deformable models–Application to brain MRI segmentation. *Pattern Recogn.*, 39(8) :1401–1414, August 2006.
6. Papandreou G. and Maragos P. Multigrid geometric active contour models. *IEEE Trans. Image Processing*, 16(1) :229–40, January 2007.
7. X. Han, C. Xu, and J.L. Prince. Fast numerical scheme for gradient vector flow computation using a multigrid method. *IET Image Processing*, 1(1) :48 – 55, March 2007.
8. M. Sabry Hassouna and Aly A. Farag. On the extraction of curve skeletons using gradient vector flow. In *ICCV, Brazil, Oct. 14-20*, pages 1–8, 2007.
9. M. Kass, A. Witkin, and D. Terzopoulos. Snake : Active contour model. *Int J. Computer Vision*, 1 :321–331, 1987.
10. B. Leroy, I. Herlin, and L.D. Cohen. Multi-resolution algorithm for active contour models. In *12th Int. Conf. Anal. Opt. System*, 15(11) :58–65, 1996.
11. Bing Li and Scott T Acton. Active contour external force using vector field convolution for image segmentation. *IEEE Trans. Image Processing*, 16(8) :2096–2106, August 2007.
12. N. Paragios, O. Mellina-Gottardo, and V. Ramesh. Gradient vector flow fast geometric active contours. *IEEE Trans. PAMI*, 26(3) :402–407, March 2004.
13. N. Ray and S. T. Acton. Motion gradient vector flow : An external force for tracking rolling leukocytes with shape and size constrained active contours. *IEEE Trans. Med. Imag.*, 23(12) :1466–1478, 2004.
14. P. Tauber, H. Batatia, and A. Ayache. A robust active contour initialization and gradient vector flow for ultrasound image segmentation. In *IAPR, Tsukuba, Japan*. Prof. Katsushi Ikeuchi, 2005.
15. Joachim Weickert, Bart M. ter Haar Romeny, and Max A. Viergever. Efficient reliable schemes for nonlinear diffusion filtering. *IEEE Trans. Image Processing*, 7(3) :398–410, 1998.
16. C. Xu and J.L. Prince. Generalized gradient vector flow external forces for active contours. *Signal Processing*, 71 :131–139, 1998.
17. C. Xu and J.L. Prince. Snakes, shapes, and gradient vector flow. *IEEE Trans. Image Processing*, 7(3) :359–369, March 1998.
18. R. Yang, M. Mirmehdi, and X. Xie. A charged active contour based on electrostatics. In *Advanced Concepts for Intelligent Vision Systems*, pages 173–184. Springer-Verlag LNCS 4179, 2006.