HAL
open science

Méthode de “Fast Marching” générique pour le “Shape From Shading”

Emmanuel Prados, Stefano Soatto

To cite this version:

Emmanuel Prados, Stefano Soatto. Méthode de “Fast Marching” générique pour le “Shape From Shading”. 15ème Congrès de Reconnaissance des Formes et Intelligence Artificielle, Jan 2006, Tours, France. inria-00377406

HAL Id: inria-00377406

https://inria.hal.science/inria-00377406v1

Submitted on 21 Apr 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.

Méthode de “Fast Marching” générique pour le “Shape From Shading”

“Generic” Fast Marching Method for the Shape From Shading problem.

E. Prados S. Soatto

UCLA Vision Lab.
420 Westwood Plaza, UCLA computer science Dept.,
Los Angeles, CA 90095-1594, USA.

eprados@cs.ucla.edu

Résumé

Nous développons une méthode numérique permettant de résoudre en un “seul passage” les équations aux dérivées partielles associées au problème du Shape From Shading “générique” [26, 27]. Notre méthode, basée sur la théorie du contrôle et sur la propagation d’interfaces, est une extension des méthodes de “Fast Marching” développées par Sethian [31]. En particulier, nous étendons les méthodes de “Fast Marching” [38, 31, 33, 9] à une classe d’équations pour lesquelles la solution n’est pas nécessairement décroissante le long des trajectoires optimales. Notre méthode permet aussi de résoudre des équations dépendantes en u, alors que les méthodes précédentes [38, 31, 33, 9] ne traitent d’équations ne dépendant que du gradient (équations de la forme $H(x, \nabla u(x)) = 0$). Par ailleurs, contrairement aux précédents algorithmes de “Shape From Shading” basés sur le “Fast Marching” [17, 39, 35], notre méthode est complètement générique. En particulier un seul algorithme permet de résoudre indifféremment les diverses équations rassemblées dans [26, 23]. Nous appliquons avec succès notre méthode de “Shape From Shading” à passage unique sur des images réelles et sur des images synthétiques.

Mots Clef

“Shape From Shading”, reconstruction 3D, Méthode de “Fast Marching”, propagation de fronts, EDP, théorie du contrôle, équations de Hamilton-Jacobi-Bellman (HJB).

Abstract

We develop an one-pass method to approximate the solutions to the “generic” Shape From Shading equations recently introduced by Prados and Faugeras [26, 27]. Our method, which is based on the control theory and the interfaces propagation, is an extension of the “Fast Marching Method” (FMM) recently introduced by Sethian [31]. In particular our method extends the FMM to some equations for which the solution is not systematically decreasing along the optimal trajectories. Our method also extends the FMM to some equations of the form $\lambda F(u(x)) + H(x, \nabla u(x)) = 0$ when [38, 31, 33, 9] only deal with equations which do not *depend on u. We apply with success our one-pass method to the SFS problem.*

Keywords

Shape From Shading, 3D reconstruction, Fast Marching Methods, front propagation, PDEs, control theory, Hamilton-Jacobi-Bellman equations.

1 Introduction

1.1 Le problème du Shape From Shading

Dès les premiers temps, le problème du “Shape From Shading” (SFS) a été un problème central dans le domaine de la vision par ordinateur. Ce problème consiste à retrouver la forme tridimensionnelle d’une surface à partir de l’information d’ombrage présente dans une unique image en noir et blanc de cette surface. Horn, pionnier dans ce domaine, a été le premier à poser ce problème sous la forme d’une équation aux dérivées partielles (EDP) non-linéaire du premier ordre appelée l’équation d’irradiance [14]. La première EDP explicite considérée dans ce domaine est l’équation Eikonale

$$|\nabla u| = r(x), \quad \forall x \in \Omega \tag{1}$$

obtenue en modélisant le problème à l’aide d’une projection orthographique et d’une unique source de lumière frontale. Ω est un ouvert de $\mathbb{R}^2$ qui représente le domaine de l’image, c’est-à-dire, le rectangle $]0, X[\times]0, Y[$. $r : \Omega \to \mathbb{R}$ est une fonction positive directement reliée à la brillance de l’image. Depuis les travaux de Horn [14], un grand nombre d’EDP correspondant à différentes modélisations du problème du SFS ont été développées et étudiées [21, 30, 34, 25, 7]. En introduisant une équation “générique”

$$H_g(x, \nabla u(x)) = 0, \forall x \in \Omega$$

(H_g étant décrit ci-dessous), Prados et Faugeras [26, 23] ont récemment unifié un grand nombre de ces équations et donc les modélisations de SFS associées (en particulier ils unifient le “SFS orthographique” et le “SFS perspectif”).

$$H_g(x,p) = \kappa_x \sqrt{|A_x p + \mathbf{v}_x|^2 + K_x^2} + \mathbf{w}_x \cdot p + c_x,$$

où les fonctions $\kappa_x, K_x, \mu_x, \nu_x, c_x, \mathbf{v}_x, \mathbf{w}_x$ and A_x dépendent de la modélisation choisie et décrivent la géométrie de la caméra, la réflectance de la surface, et les paramètres de l’éclairage [26, 23]. Par exemple, l’Hamiltonien classique de Rouy/Tourin [30]

$$H_{R/T}(x,p) = I(x)\sqrt{1+|\nabla u|^2} + \mathbf{l} \cdot \nabla u - \gamma \quad (2)$$

obtenu en modélisant le problème du SFS orthographique avec une unique source de lumière éloignée, est un cas particulier de l’Hamiltonien du SFS “générique” H_g ($\mathbf{L} = (\mathbf{l}, \gamma)$ représente la direction de la source de lumière ; $I(x)$ est la brillance de l’image au pixel x). Récemment, Prados et Faugeras ont également montré qu’en prenant en compte l’atténuation de l’intensité de la lumière due à la distance [27, 28], le problème aboutit à une équation de la forme

$$-e^{-2u(x)} + H_{att}(x, \nabla u(x)) = 0 \quad (3)$$

où H_{att} est un cas particulier de l’Hamiltonien “générique” H_g. Contrairement aux équations précédentes, cette dernière équation, complétée par des contraintes d’état [27, 28, 23], permet d’aboutir à un problème complétement bien posé. Aussi cette modélisation s’est avérée très pertinente en pratique.
Dans [26, 24, 27, 23, 28] Prados et al. se concentrent sur des méthodes itératives. En fait, à l’exception de quelques rares travaux [4, 16, 31, 17, 39], la plupart des méthodes de SFS proposées dans la littérature sont itératives ; citons en particulier [21, 30, 26, 27] (parmi tant d’autres !). Malgré leur multiplicité, ces méthodes itératives souffrent de certaines faiblesses d’“optimalité”, étant donné qu’elles utilisent des scans alternatifs semblables à ceux proposés par Danielsson [10].

1.2 Méthodes de “Fast Marching”

Contrairement aux méthodes itératives, les méthodes du type “Fast Marching” [38, 31, 33, 9] sont des méthodes (dites) à “passage unique” (bien que la valeur de la solution à un même pixel est mise à jour plusieurs fois). Chacune de ces méthodes (“Fast Marching” versus “itératives”) a ses propres avantages et inconvénients ; ceci dépendant de l’application souhaitée. Par exemple, les méthodes itératives utilisent généralement des seuils arbitrairement choisis dans leurs tests d’arrêt ($\|U^n - U^{n-1}\| \leq \delta$, où $\delta > 0$ est le seuil donné ; U^n étant la n-ème approximation de la solution). Alors que les méthodes à “passage unique” ne nécessitent pas de seuil. Ces dernières ont cependant besoin d’une structure permettant le tri d’une pile de données (ce qui mène d’ailleurs l’algorithme global à une complexité en $N \log N$, où N est le nombre de pixels de l’image). Ce tri supplémentaire implique un coût additionnel en temps de calcul qui n’apparaît pas dans les méthodes itératives. Par contre, les méthodes itératives nécessitent en contrepartie beaucoup plus de mises à jour pour approximer une solution de viscosité. Enfin, remarquons que les méthodes itératives sont plus faciles à implémenter que les méthodes à “passage unique” car elles ne nécessitent pas de structure de tri. Dans les cas où la complexité de l’implémentation n’est pas une limitation, où le calcul de la mise à jour à chacun des sites est très coûteux (avec des équations complexes, le coût des mises à jour devient rapidement plus important que le coût induit par le tri), et/ou la convergence vers la solution de viscosité est essentielle, les FMM ont clairement l’avantage.

Initialement, les méthodes de “Fast Marching” étaient limitées aux équations Eikonales [38, 31, 9, 15, 12, 11]. Récemment, Sethian et Vladimirsky [33] ont étendu ces méthodes aux équations de “propagation” du type :

$$|\nabla u| F\left(x, \frac{\nabla u}{|\nabla u|}\right) = 1, \quad \forall x \in \Omega,$$

où F est une fonction continue, bornée, strictement positive, convexe et homogène d’ordre 1 par rapport à $\frac{\nabla u}{|\nabla u|}$. Ici, nous allons encore plus loin et nous étendons les méthodes “Fast Marching” aux équations de la forme

$$\lambda F(u(x)) + H(x, \nabla u(x)) = 0 \quad \forall x \in \Omega, \quad (4)$$

où Ω est un ouvert de $\mathbb{R}^N$, $H : \Omega \times \mathbb{R}^N \to \mathbb{R}$ est convexe par rapport à ∇u, $F : \mathbb{R} \to \mathbb{R}$ est une fonction strictement croissante[1] et $\lambda \geq 0$. Il est bien connu que, en général, les équations de HJ n’ont pas de solution au sens classique (les solutions de ce type d’équations ont généralement des singularités dans leurs dérivées) ; la notion de solutions de viscosité [8, 18, 2] fournit un cadre mathématique véritablement approprié pour traiter ce type d’équations. Dans [33] les auteurs se limitent aux équations du type (4), vérifiant $\lambda = 0$ et $H(x, 0) = 0$ (c’est également le cas pour d’autres méthodes numériques rapides telles que [37]). Ici, nous nous affranchissons de ces hypothèses augmentant ainsi le potentiel des domaines d’applications de ces méthodes (qui d’ailleurs déjà ont démontré leur utilité dans nombre d’applications [6, 17, 32]). Aussi, de nouvelles applications sont d’ores et déjà prévisibles par exemple en finance [36, 19] et en animation [1]. Notre extension permet par ailleurs d’utiliser des modèles plus sophistiqués et réalistes en SFS. En effet, de manière générale, les équations de SFS (en particulier celles recensées dans [26, 23, 27]) ne rentrent pas dans le cadre considéré par Sethian et Vladimirsky dans [33]. C’est en particulier le cas des équations (3) et (2). En effet, pour certains pixels x de l’image I considérée, $H_{R/T}(x, 0) = I(x) - \gamma$ peut être strictement négatif. Par ailleurs, (3) dépend de u (les équations considérées dans [38, 31, 33, 9] ne dépendent que de ∇u). Dans

[1] Nous supposons également qu’à la suite du changement de variables $v = F(u)$, l’Hamiltonien reste convexe par rapport à ∇v.

cet article, nous proposons une extension des méthodes de "Fast Marching" applicables à toutes les équations explicitées [26, 27, 23]. Par ailleurs, notre méthode est "générique". Ainsi, un unique algorithme permet de calculer des solutions numériques des différentes modélisations considérées jusqu'à présent.

1.3 Méthodes de "Fast Marching" pour le Shape From Shading : état de l'art

Les méthodes à "passage unique" reliées à la propagation d'interfaces telles que la méthode des lignes de niveau ("level set method") introduite par Osher et Sethian [22] et les Méthodes de Fast Marching [38, 31] ont déjà été appliquées avec succès au problème du SFS [16, 31, 17, 39, 35].

En particulier, Sethian [31] a été le premier à appliquer la méthode de "Fast Marching" pour résoudre le problème du SFS. Ce premier travail traitait seulement la version la plus simple du problème du SFS associé à l'équation Eikonale (1) (caméra orthographique ; unique source de lumière frontale et distante). Récemment, Kimmel et Sethian [17] ont proposé une adaptation de ce premier algorithme de manière à prendre en compte une source de lumière (distante) oblique. Cependant, ce nouvel algorithme est basé sur un changement de variables délicat et ne semble pas pouvoir s'adapter aux modélisations plus générales. En fait, la plupart des auteurs [17, 35, 7] utilisent différentes techniques (changement de variables, introduction de nouvelles inconnues) de manière à retrouver l'équation Eikonale et ainsi pouvoir utiliser les outils initiaux développés par Rouy et Tourin [30] et/ou Sethian [31].

Yuen, Tsui et al. [39] sont les premiers à proposer une méthode de "Fast Marching" pour le "Shape From Shading perspectif". Contrairement à [35], Yuen et ses coauteurs [39] proposent une véritable adaptation des FMM à ce problème. Cependant, dans [39], les auteurs se restreignent au cas d'une unique source de lumière frontale ; aussi dans ce cas, la fonction coût l sous-jacente reste positive. Enfin, ce travail [39] souffre d'un défaut de simplification.

Contrairement à tous les travaux précédents, la méthode de "Fast Marching" que nous proposons dans cet article permet de résoudre indifféremment la plupart des EDP associées aux modélisations les plus récentes, réalistes et pertinentes de la littérature du SFS. En particulier, elle permet de résoudre toutes les équations de SFS recensées dans [23] (dont une grande partie ont des fonctions coûts l de signe arbitraire). Contrairement par exemple à [17], nous n'avons besoin d'aucun changement de variables.

2 Formulation du problème en termes de contrôle

Dans cette section, nous reformulons le problème en utilisant les outils de la théorie du contrôle. Le lecteur peu familier avec ces outils pourra se référer au livre [2]. Tout d'abord, nous utilisons la transformée de Legendre [18] afin de réécrire l'équation (4) sous la forme : $\forall\, x \in \Omega$,

$$\lambda F(u(x)) + \sup_{a\in A}\{-f(x,a)\cdot\nabla u(x) - l(x,a)\} = 0. \quad (5)$$

Les fonctions l et f sont respectivement appelées la "fonction coût" et la "dynamique" ; $a \in A$ est une "valeur admissible du contrôle". Par exemple, nous pouvons réécrire l'Hamiltonien H_g du SFS "générique" en tant que sup avec

$$\begin{aligned} f(x,a) &= -\,[\,\kappa_x R_x^T D_x R_x.a + \mathbf{w}_x\,],\\ l(x,a) &= -\,[\,K_x\kappa_x\sqrt{1-|a|^2} + \kappa_x(R_x^T\mathbf{v}_x)\cdot a + c_x\,], \end{aligned}$$

et où A est la boule unité fermée de $\mathbb{R}^2$; R_x^T est ici la matrice transposée de R_x ; voir [23] pour tous les détails.
Pour être bien posé, ce type d'équation doit être complétée par des conditions aux limites. Nous ajoutons donc les contraintes suivantes :

$$u(x) = \varphi(x) \qquad \forall x \in \mathcal{T}, \quad (6)$$

$$u(x) = +\infty \qquad \forall x \in \partial\Omega \setminus \mathcal{T}, \quad (7)$$

où la "cible" $\mathcal{T}$ est un sous-ensemble fermé non vide de $\overline{\Omega}$ et $\varphi : \mathcal{T} \to \mathbb{R}$ définit une contrainte de type Dirichlet. Ainsi les équations (5) et (4) sont maintenant considérées sur $\Omega \setminus \mathcal{T}$ au lieu de Ω. Sans perte de généralité[2], supposons que $F(u) = u$. Dans ce cas, l'équation (5) est précisément une équation d'*Hamilton-Jacobi-Bellman* (2). Soit V la fonction valeur

$$V(x) = \inf_{\alpha\in\mathcal{A}}\left\{\int_0^{\tau_x(\alpha)} l(y_x(\tau,\alpha),\alpha(\tau))e^{-\lambda\tau}d\tau\ + \varphi(y_x(\tau_x(\alpha),\alpha))e^{-\lambda\tau_x(\alpha)}\right\}. \quad (8)$$

$\mathcal{A}$ est l'ensemble des "contrôles admissibles" (un ensemble de fonctions mesurables de $t \in [0,+\infty[$ à valeurs dans A : ce dernier étant un fermé borné de $\mathbb{R}^M$). $y_x(.,\alpha)$ est une "trajectoire" contrôlée par α partant de x ; c'est la solution du système dynamique

$$\begin{cases} y'(\tau) = f(y(\tau),\alpha(\tau)), & \tau > 0,\\ y(0) = x. \end{cases} \quad (9)$$

$\tau_x(\alpha)$ est la plus petite valeur τ telle que $y_x(\tau,\alpha)$ atteigne[3] $\mathcal{T}\cup\partial\Omega$. Il est bien connu [18, 2] que V est l'unique solution de viscosité[4] de l'équation (5)-(6)-(7) et que V vérifie le *principe de programmation dynamique* [18, 2] : pour $x \in \Omega$ et le voisinage $\mathcal{N}_x$ de x (tel que $\mathcal{N}_x \cap (\mathcal{T}\cup\partial\Omega) = \emptyset$)

$$V(x) = \inf_{\alpha\in A}\left\{\int_0^{\tau_x^{\mathcal{N}}(\alpha)} l(y_x(\tau,\alpha),\alpha(\tau))e^{-\lambda\tau}d\tau\ + V(y_x(\tau_x^{\mathcal{N}}(\alpha),\alpha))e^{-\lambda\tau_x^{\mathcal{N}}(\alpha)}\right\}, \quad (10)$$

[2]Nous pouvons définir $v = F(u)$, et considérer l'Hamiltonien obtenu après le changement de variables associé. Rappelons que nous avons supposé que l'Hamiltonien reste convexe par rapport à ∇v.

[3]Pour simplifier les notations, nous étendons φ sur $\mathcal{T}\cup\partial\Omega$, et nous définissons $\varphi(x) = +\infty, \forall x \in \partial\Omega\setminus\mathcal{T}$.

[4]Dans le cas où $\lambda = 0$ et en présence de singularités, nous devons considérer la notion de "Solutions de Viscosité Singulières Discontinues" (SDVS) de (5)-(6)-(7) au lieu de la notion classique de solutions de viscosité [23].

$\tau_x^{\mathcal{N}}(\alpha)$ étant le plus petit τ tel que $y_x(\tau) \notin \mathcal{N}_x$. $\alpha_x^{\mathcal{N}}$ représente le contrôle optimal associé.
Finalement, nous notons α_x^*, τ_x^* et y_x^* le contrôle optimal, le temps optimal et la *trajectoire optimale* associés à (8) (c'est-à-dire pour laquelle l'$\inf$ de (8) est atteint).

Exemple 1 *Dans le cas particulier de l'équation Eikonale* (1), *les trajectoires optimales correspondent aux lignes de gradient. Cependant, comme le montre [33], ce n'est généralement pas vérifié* ($\nabla u(x) \neq y_x^{*\prime}(0) = f(x, \alpha_x^*(0))$).

3 Schéma d'Approximation et Causalité

Pour simplifier, nous utilisons dans cet article une grille régulière. Dans [23, 29] nous montrons comment étendre notre méthode aux grilles irrégulières. Le lecteur peu familier avec la notion de schémas d'approximation pourra se référer à [3]. Rappelons simplement que d'après [3], un schéma d'approximation est une équation fonctionnelle de la forme

$$S(\rho, x, u(x), u) = 0, \quad \forall x \in \overline{\Omega},$$

qui "approxime" l'EDP considérée. S est définie sur $\mathcal{M} \times \overline{\Omega} \times \mathbb{R} \times B(\overline{\Omega})$ dans $\mathbb{R}$, $\mathcal{M} = \left(\mathbb{R}^+\right)^N$ et $\rho = (h_1, ..., h_N) \in \mathcal{M}$ définit la taille de la grille utilisée dans les algorithmes numériques correspondants. $B(D)$ est l'espace des fonctions bornées définies sur un ensemble D. De manière à obtenir un schéma cohérent ("consistent"), la fonction S est obtenue en approximant ∇u par des différences finies, puis en substituant ∇u par son approximation dans l'équation initiale. La propriété principale permettant d'assurer la convergence de la solution calculée numériquement vers la solution de viscosité est la *monotonie* du schéma (c'est-à-dire la fonction $u \mapsto S(\rho, x, t, u)$ est décroissante) [3].
Dans la plupart des FMM, le gradient ∇u est discrétisé par

$$\nabla u(x) \simeq \left[\frac{t - u(x + s_i h_i e_i)}{-s_i h_i}\right],$$

où par abus nous notons $\left[\frac{t-u(x+s_i h_i e_i)}{-s_i h_i}\right]$ le vecteur dont la $i^{\text{ième}}$ composante est $\frac{t-u(x+s_i h_i e_i)}{-s_i h_i}$. Ci-dessus, t correspond à $u(x)$; nous remplaçons $u(x)$ par t de manière à insister sur le fait que celui-ci correspond à la *valeur mise à jour*. $(e_1, .., e_n)$ est la base canonique ; $\forall i,\ s_i \in \{\pm 1\}$. A l'exception de [33], le simplex $\{x, x + s_1 h_1 e_1, ..., x + s_n h_n e_n\}$ (en pratique le vecteur de signes $(s_1, .., s_n)$) est choisi de manière à "contenir $-\nabla u$" ; c'est-à-dire tel que

$$\forall i = 1..n, \quad s_i \text{ est l'opposé du signe de } [\nabla u]_i. \qquad (11)$$

Dans le cas Eikonal, la formulation en terme de contrôle de l'équation (1) est $\sup_{a \in \overline{B}(0,1)}\{a \cdot \nabla u(x) - r(x)\} = 0$. Dans cet exemple, le contrôle optimal est toujours $a_x = \frac{\nabla u(x)}{|\nabla u(x)|}$. Ainsi le schéma

$$\begin{aligned} S_{Eiko}(\rho, x, t, u) &= \left|\frac{t - u(x + s_i h_i e_i)}{-s_i h_i}\right| - r(x) \\ &= \sup_{a \in \overline{B}(0,1)} \left\{a \cdot \left[\frac{t - u(x + s_i h_i e_i)}{-s_i h_i}\right] - r(x)\right\}, \qquad (12) \end{aligned}$$

avec s_i défini par (11), est clairement monotone. Cependant, ceci est généralement faux pour toute équation de la forme (4). En effet, dans le cas général, le schéma

$$\begin{aligned} S_c(\rho, x, t, u) &= \lambda F(t) + H\left(x, \left[\frac{t - u(x + s_i h_i e_i)}{-s_i h_i}\right]\right) \\ &= \lambda F(t) + \sup_{a \in A}\{-f(x, a) \cdot \left[\frac{t - u(x + s_i h_i e_i)}{-s_i h_i}\right] - l(x, a)\} \qquad (13) \end{aligned}$$

avec s_i défini par (11), n'est plus monotone dès qu'il existe $i \in [1..n]$ tel que $sign[f(x, a_x)]_i \neq s_i$ $(= -sign[\nabla u]_i)$, où a_x est le a optimal de (13) [t étant la racine de $S_c(\rho, x, t, u) = 0$]. Pour obtenir un schéma monotone, nous devons changer le choix du simplex (c'est-à-dire le choix des s_i). En fait, nous devons choisir le simplex par rapport à la dynamique du contrôle optimal (et non par rapport au gradient). Si nous définissons $s_i(x, a) = sign f_i(x, a)$, le schéma[5]

$$\begin{aligned} &S(\rho, x, t, u) = \lambda F(t) \ + \\ &\sup_{a \in A}\{-f(x, a) \cdot \left[\frac{t - u(x + s_i(x, a) h_i e_i)}{-s_i(x, a) h_i}\right] - l(x, a)\} \qquad (14) \end{aligned}$$

est naturellement monotone, et le "bon". En d'autres termes, ce schéma est obtenu à partir de (4) en utilisant seulement les approximations en arrière ("backward") et en avant ("forward") des dérivées partielles et en choisissant la bonne approximation *selon la dynamique du contrôle optimal*. Dans le cas Eikonal, ce schéma coincide avec celui classique de Rouy/Tourin [30]. Par ailleurs, nous souhaitons mettre en avant une autre propriété du schéma (14). Pour simplifier l'exposé, nous supposerons temporairement $\lambda = 0$.
Soit $\mathcal{N}_x$ un "infiniment" petit voisinage de x tel que la valeur de la solution sur $\partial\mathcal{N}_x$ soit connue. Par le principe de programmation dynamique nous avons

$$u(x) = u(y_x(\tau_x(\alpha_x^{\mathcal{N}}), \alpha_x^{\mathcal{N}})) + \int_0^{\tau_x(\alpha_x^{\mathcal{N}})} l(y_x(\tau, \alpha_x^{\mathcal{N}}), \alpha_x^{\mathcal{N}}(\tau))\, d\tau.$$

En utilisant les approximations classiques

$$\begin{aligned} &y_x(\tau_x(\alpha_x^{\mathcal{N}}), \alpha_x^{\mathcal{N}}) \simeq x + \tau_x(\alpha_x^{\mathcal{N}}) \cdot f(x, \alpha_x^{\mathcal{N}}(0)), \\ &u(x + \tau_x(\alpha_x^{\mathcal{N}}) f(x, \alpha_x^{\mathcal{N}}(0)) \simeq \\ &\left[1 - \frac{\tau_x(\alpha_x^{\mathcal{N}})}{h}\sum_{j=1}^N |f_j|\right] u(x) + \frac{\tau_x(\alpha_x^{\mathcal{N}})}{h}\sum_{i=1}^N |f_i| u(x + h s_i e_i), \end{aligned}$$

où s_i = sign $f_i(x, \alpha_x^*(0))$ – nous choisissons le simplex contenant $x + \tau_x(\alpha_x^*) f(x, \alpha_x^*(0))$ – le lecteur peut vérifier facilement que nous obtenons le schéma

$$\sum_{i=1}^N |f_i(x, \alpha_x^*(0))| \frac{(t - u(x + h s_i e_i))}{h_i} - l(x, \alpha_x^*(0)) = 0$$

qui est équivalent au schéma (14). De cette manière, la relation entre la dynamique du contrôle optimal et le choix du simplex peut paraître plus naturelle. En supposant que nous connaissons la valeur de la solution sur un voisinage $\mathcal{W}$ de

[5] Schéma déjà suggéré par Prados et Faugeras [25, 23].

$\mathcal{T}$, ce second cheminement permet de comprendre pourquoi le schéma (14) permet de calculer une approximation pertinente de la valeur au point x de la solution de l'équation (5)-(6)-(7) dans laquelle nous remplaçons Ω par $\mathcal{W}$, où $x \in \partial\mathcal{W}$. Nous renvoyons le lecteur à [29, 23] pour plus de détails. Notons cependant que le schéma d'approximation proposé par Sethian et Vladimirsky est différent de celui présenté ici : en particulier, le schéma proposé dans [33] est basé sur l'utilisation de plus de simplex que ceux contenus dans des voisinages immédiats du site considéré, contrairement à notre schéma (14). Notons au passage qu'en utilisant les résultats et contributions présentés ici, il est alors possible (même relativement direct) d'étendre la méthode développée par Sethian et Vladimirsky [33] (OUM) aux diverses équations présentées ici.

Une autre conséquence fondamentale de l'étude ci-dessus est la suivante : pour qu'un algorithme à "passage unique" approxime la solution de viscosité, la reconstruction doit suivre (traquer) les trajectoires optimales. Sethian et Vladimirsky [33] démontrent cet aspect fondamental intuitivement et expérimentalement. Contrairement aux premiers travaux [38, 31] dans lequel la causalité était directement liée aux lignes de gradient, dans [33] la propriété de causalité est basée sur les trajectoires optimales. Cependant, la causalité de Sethian et Vladimirsky est basée sur le fait que la solution est strictement décroissante le long des trajectoires optimales. Cette propriété est vérifiée par les équations considérées dans [33]

$$\sup_{a\in\mathcal{A}}\{-\mathsf{f}(x,a)a\cdot\nabla u-1\}=0,\quad \forall x\in\Omega, \tag{15}$$

car, pour ces équations, la fonction coût est $l(x,a)=1>0$. Mais, pour les équations du type (4) pour lesquelles la fonction coût l prend des valeurs négatives, cette propriété de monotonie n'est plus valable : la solution peut de manière arbitraire et alternative croître et décroître le long des trajectoires optimales. Aussi, dans le cas général la propriété de causalité utilisée par Sethian et Vladimirsky [33] ne s'applique pas.

Finalement, insistons sur le fait qu'il a été montré dans [23, 26], que la plupart des équations de Shape From Shading ont généralement des fonctions coût de signe arbitraire. C'est le cas par exemple pour l'Hamiltonien classique de Rouy/Tourin $H^{orth}_{R/T}$ (où, $l_{R/T}(x,a)=I(x)\sqrt{1+|a|^2}-\gamma$) et pour l'Hamiltonien du SFS perspectif $H^{pers}_{P/F}$ [25] qui rentre dans la classe des Hamiltoniens donnés par l'équation (4).

Dans la suite, nous réinterprétons légèrement les FMM ; nous généralisons et affaiblissons le principe de causalité. Puis nous proposons une nouvelle propriété de causalité fonctionnelle qui étend celle utilisée par [33] à toutes les équations du type (4).

4 Ordre de mise à jour pour les méthodes à "passage unique"

4.1 Travaux précédents et idées basiques

Temporairement, supposons que *nous connaissons les trajectoires optimales* y^*_x. Nous pouvons alors retrouver directement la solution de l'équation (5)-(6)-(7) en remontant ces courbes :

$$V(x)=\int_0^{\tau_x(\alpha^*_x)} l(y^*_x(\tau),\alpha^*_x(\tau))e^{-\lambda\tau}d\tau\ + \varphi(y^*_x(\tau_x(\alpha^*_x)))e^{-\lambda\tau_x(\alpha^*_x)}. \tag{16}$$

D'un point de vue numérique, nous pouvons alors reconstruire la solution par une intégration directe le long des trajectoires optimales. Cette idée correspond grossièrement à la méthode des *bandes caractéristiques* introduites dans la littérature du Shape from Shading par Horn [13] pour résoudre l'équation Eikonale (dans [13], la connaissance des trajectoires optimales était implicitement remplacée par la connaissance de conditions de Neumann qui étaient alors propagées). Cependant, cette méthode est fortement limitée par des problèmes d'instabilité et de "recollage". Pour s'affranchir de ces limitations, nous devons remonter toutes les trajectoires optimales simultanément et non pas séparément. L'idée est alors la suivante. Nous supposons que la solution est connue à l'intérieur d'une courbe fermée $\mathcal{C}_t$ ($t\geq 0$) qui se propage le long des trajectoires optimales. Conceptuellement, ces courbes $\mathcal{C}_t$ doivent vérifier : pour tout x dans Ω, la trajectoire optimale y^*_x coupe toujours la courbe $\mathcal{C}_t$, mais seulement une fois pour chaque $t\geq 0$. De plus, si $t_1<t_2$ alors $\Omega_{t_1}\subset\Omega_{t_2}$ (où Ω_t est l'ouvert vérifiant $\partial\Omega_t=\mathcal{C}_t$). Cette idée correspond exactement à celle introduite par Bruckstein [4] pour résoudre le problème du SFS Eikonal. En fait, la méthode de Bruckstein [4] est une variante de la méthode des bandes caractéristiques de Horn [13] qui permet aussi de s'affranchir des conditions de Neumann nécessaires à [13] en imposant des conditions de Dirichlet spécifiques, contraignant la courbe initiale à être un iso-contour de la solution, puis en propageant ce contour le long des caractéristiques. Cependant la méthode de Bruckstein est une méthode Lagrangienne qui souffre d'instabilité et de problèmes topologiques, voir par exemple [22]. Pour éviter ces problèmes numériques, un moyen très efficace consiste à utiliser les techniques de "level set" introduites par Osher et Sethian [22] ou bien les techniques de Fast Marching [31, 32]. Notons au passage l'apparition notable de quelques extensions de la FMM traitant plusieurs points en parallèle. Ces méthodes appelées "Group Marching" [15] et "Massive Marching" [11, 12] proposent en un sens deux affaiblissements de l'ordonnancement de la FMM.

4.2 Propagations de courbes et coûts associés

Maintenant, nous supposons que *nous ne connaissons pas les trajectoires optimales*. C'est généralement le cas dans la

plupart des applications, en particulier en Shape From Shading. Dans ce cas, nous devons reconstruire simultanément la solution u et les courbes $\mathcal{C}_t$. Ainsi, nous sommes certains que les mises à jour des valeurs de la solution se propagent de la même manière que les trajectoires optimales.
L'idée est alors la suivante : supposons que nous connaissons déjà $\mathcal{C}_t$ et les valeurs de la solution dans Ω_t. Maintenant, si nous voulons calculer les valeurs de la solution u sur Ω_{t+dt}, nous devons calculer explicitement ces valeurs (bien sûr), mais nous devons aussi calculer la nouvelle courbe $\mathcal{C}_{t+dt}$ (ou son ensemble associé Ω_{t+dt}).
De manière à *manipuler les courbes $\mathcal{C}_t$ facilement, nous les définisssons comme les ensembles de niveau ("level sets")* d'un *"coût"* $\mathsf{C}(x)$ défini sur $\overline{\Omega}$ à valeurs dans $\mathbb{R}$.

Exemple 2 *Dans le cas de la méthode de "Fast Marching" basique pour l'équation Eikonale [38, 31], le coût choisi* C *est égal à* u*. Aussi, les courbes* $\mathcal{C}_t$ *correspondent aux lignes de niveau (isocontours) des solutions.*

De manière à atteindre notre but, insistons sur une importante (mais bien trop souvent négligée) difficulté rencontrée lorsque nous traitons des équations du type (5)-(6)-(7) : les trajectoires optimales y_x^* et la solution V dépendent de manière significative de l'ensemble Ω (bien sûr, elles dépendent également de l'Hamiltonien considéré et des conditions de Dirichlet imposées (6)).

Exemple 3 *Soit* $x \in \Omega \setminus \mathcal{T}$ *telle que sa trajectoire optimale* y_x^* *associée à l'équation* (5)-(6)-(7) *atteigne la cible* $\mathcal{T}$ *; c'est-à-dire* $y_x^*(\tau_x^*) \in \mathcal{T}$*. Soit* $\mathcal{W}$ *un ouvert connexe tel que*
- $x \in \mathcal{W}$ *et* $\mathcal{T} \subset \mathcal{W} \subset \Omega$,
- *il existe* $t \geq 0$ *tel que* $y_x^*(t) \notin \mathcal{W}$*. En d'autres termes, la trajectoire optimale* y_x^* *sort à un moment de* $\mathcal{W}$*.*

Maintenant, considérons l'équation (5)-(6)-(7) *dans laquelle nous remplaçons* Ω *par* $\mathcal{W}$*. Notons* $(5_{\mathcal{W}})$-$(6_{\mathcal{W}})$-$(7_{\mathcal{W}})$ *cette nouvelle équation.* Par définition, *la trajectoire optimale démarrant de* x *et associée à la nouvelle équation* $(5_{\mathcal{W}})$-$(6_{\mathcal{W}})$-$(7_{\mathcal{W}})$ *reste à l'intérieur de* $\mathcal{W}$ *(avec des hypothèses basiques*[6] *sur la dynamique* f*, nous pouvons démontrer qu'elle atteint* $\mathcal{T}$*) et donc elle diffère totalement de* y_x^**. De la même façon, la solution* $V_{\mathcal{W}}$ *de* $(5_{\mathcal{W}})$-$(6_{\mathcal{W}})$-$(7_{\mathcal{W}})$ *est différente de la solution* V *de* (5)-(6)-(7).

C'est pourquoi dans la suite, lorsque cela est pertinent, nous spécifions en indice l'ensemble considéré, comme suit : $\alpha_x^* = \alpha_{x(\Omega)}^*$, $y_x^* = y_{x(\Omega)}^*$, $\tau_x^* = \tau_{x(\Omega)}^*$, $V = V_\Omega$.
Donc, pour être totalement rigoureux, si les trajectoires optimales dépendent de l'ensemble sur lequel nous travaillons, alors les courbes $\mathcal{C}_t$ et leur coût associé C doivent également en dépendre. Aussi, nous utiliserons le même type de notations que ci-dessus pour le coût C (cependant, pour éviter les confusions, dans la suite nous utilisons les notations $\mathcal{C}_t$ et Ω_t associées aux ensembles de niveau de C, seulement en référence à Ω et à l'équation (5)-(6)-(7)).
Les précautions ci-dessus sont principalement motivées par la remarque suivante. Premièrement, rappelons que nous ne connaissons le coût C_Ω que sur Ω_t (mais pas sur $\Omega \setminus \Omega_t$) (connaître le coût C_Ω est équivalent à connaître les courbes $\mathcal{C}_t$). Il semble raisonnable de supposer que nous ne pouvons pas calculer directement C_Ω sur un voisinage (même très proche) $\mathcal{N}$ de Ω_t ($\mathcal{N} \subset \Omega$). En effet, tout point $x \in \mathcal{N} \setminus \overline{\Omega}_t$ (même très proche de $\mathcal{C}_t$) peut avoir une trajectoire optimale $y_{x(\Omega)}^*$ qui s'éloigne énormément de $\mathcal{C}_t$ avant de revenir dans Ω_t. Cependant, il est raisonnable de supposer que nous pouvons calculer (une approximation de) $\mathsf{C}_\mathcal{N}$ sur un proche voisinage $\mathcal{N}$ de Ω_t (rappelons que $\mathsf{C}_\mathcal{N}$ est le coût associé à l'équation $(5_\mathcal{N})$-$(6_\mathcal{N})$-$(7_\mathcal{N})$).

Exemple 4 *Selon les notations de [33], le voisinage* $\mathcal{N}$ *(considéré ci-dessus) correspond avec l'ensemble des points "*Considered*" des FMM classiques. Dans ces méthodes, le coût* C *correspond à la solution de l'équation considérée. Aussi, les valeurs "*$V(x)$*" calculées dans [33] pour tous points* x *de l'ensemble "*Considered*" sont des approximations numériques des coûts* $\mathsf{C}_\mathcal{N}(x)$*.*

En pratique, il semble donc raisonnable de pouvoir calculer $\mathsf{C}_\mathcal{N}$ sur $\mathcal{N}$ et non C_Ω. Cependant, à partir de la connaissance du coût $\mathsf{C}_\mathcal{N}$ sur $\mathcal{N}$, nous espérons pouvoir déduire C_Ω sur un sous-ensemble $\mathcal{C}_{t+dt}$ de $\mathcal{N}$ ($\mathcal{C}_t \subset \mathcal{C}_{t+dt} \subset \mathcal{N}$).
Aussi, si les coûts vérifient les hypothèses suivantes :
(H1) $\mathsf{C}_\mathcal{W}(x)$ est décroissant le long des trajectoires optimales $y_{x(\mathcal{W})}^*$, c'est-à-dire : pour tout x et $\tau > 0$ (assez petit) $\mathsf{C}_\mathcal{W}(y_{x(\mathcal{W})}^*(\tau)) \leq \mathsf{C}_\mathcal{W}(x)$,
(H2) $\mathsf{C}_\mathcal{W}$ est décroissant par rapport à $\mathcal{W}$; c'est-à-dire : si $\Omega_1 \subset \Omega_2$, alors $\mathsf{C}_{\Omega_1}(x) \geq \mathsf{C}_{\Omega_2}(x)$,
(H3) soit $x_1 \in \Omega_1 \subset \Omega_2$. Si $y_{x_1(\Omega_2)}^*(.)$ reste dans Ω_1 alors $\mathsf{C}_{\Omega_2}(x_1) = \mathsf{C}_{\Omega_1}(x_1)$,
alors nous avons la proposition suivante

Proposition 1 *Supposons que les hypothèses* (H1), (H2) *et* (H3) *soient vérifiées. Soit* $\mathcal{C}_{t+dt}$ *telle que* $\mathcal{C}_t \subset \mathcal{C}_{t+dt} \subset \mathcal{N}$*. Alors pour tout* x *dans* Ω_{t+dt}*,*

$$\mathsf{C}_\Omega(x) = \mathsf{C}_\mathcal{N}(x).$$

En d'autres termes, $\mathsf{C}_\mathcal{N}$ coincide avec C_Ω sur Ω_{t+dt}.

Preuve Soit x dans Ω_{t+dt}. Par l'hypothèse (H1), la trajectoire optimale $y_{x\,(\Omega)}^*$ reste dans Ω_{t+dt}. Nous appliquons alors l'hypothèse (H3) avec $\Omega_1 = \Omega_{t+dt}$ et $\Omega_2 = \Omega$. Nous avons donc $\mathsf{C}_\Omega(x) = \mathsf{C}_{\Omega_{t+dt}}(x)$. Comme $\mathcal{C}_{t+dt} \subset \mathcal{N} \subset \Omega$, alors par l'hypothèse (H2), $\mathsf{C}_\Omega(x) = \mathsf{C}_{\Omega_{t+dt}}(x) = \mathsf{C}_\mathcal{N}(x)$. □

Maintenant, considérons le problème du point de vue discret. Nous supposons que l'ensemble $\overline{\Omega}$ est "recouvert" par une grille de points. Nous divisons l'ensemble des points de la grille en trois classes (comme pour les méthodes de "Fast Marching" classiques [33]) : *Accepté* **A**, *Considéré* **C**, *Eloigné* **F** ("Far"). Les points *Acceptés* sont ceux dans Ω_t (c'est-à-dire à l'intérieur de $\mathcal{C}_t$) ; nous supposons donc que nous connaissons déjà la valeur de la solution pour tous les points de **A**. L'ensemble des points *Considérés* **C** est l'ensemble des points adjacents aux points *Acceptés*. L'union $\mathbf{C} \cup \mathbf{A}$ est la version discrète du voisinage $\mathcal{N}$ de $\overline{\Omega}_t$. L'ensemble des points *Eloignés* **F** correspond aux autres points de la grille.

[6] ou par exemple, en supposant que l'Hamiltonien est coercif.

Rappelons que nous connaissons les valeurs de la solution de (5)-(6)-(7) sur Ω_t et que nous voulons calculer celles sur Ω_{t+dt}. Aussi, ceci nécessite un calcul préliminaire du nouvel ensemble Ω_{t+dt}. D'un point de vue pratique et discret, nous voulons étendre Ω_t à Ω_{t+dt} de manière à ce que $\Omega_{t+dt} \setminus \Omega_t$ ne contienne qu'un seul point de la grille. Ceci est donc équivalent à trouver le point x_0 de $\mathbf{C}$ qui a le coût $\mathsf{C}_\Omega(x_0)$ le plus bas ($\mathsf{C}_\Omega(x_0) = \min_{x\in\mathbf{C}} \mathsf{C}_\Omega(x)$) et ensuite à le transférer dans l'ensemble des points *Acceptés* $\mathbf{A}$ (lequel est alors élargi). Ainsi, si les coûts C vérifient les hypothèses (H1), (H2) et (H3), alors la proposition suivante permet de trouver ce point x_0 (en supposant que nous connaissons les coûts $\mathsf{C}_\mathcal{N}(x)$ pour tout x dans $\mathcal{N}$).

Proposition 2 *Supposons que les coûts* C *vérifient* (H1), (H2) *et* (H3). *Supposons de plus qu'il existe* $x_1 \in \mathsf{C}$ *tel que la ligne de niveau de* C_Ω *de valeur* $\mathsf{C}_\Omega(x_1)$ *soit incluse dans* $\mathcal{N}$. *Alors le point* $x_0 \in \mathbf{C}$ *qui a le plus petit coût* $\mathsf{C}_\Omega(x_0)$ *est le point* x *de* $\mathbf{C}$ *qui a le plus petit coût* $\mathsf{C}_\mathcal{N}(x)$ *; de plus,* $\mathsf{C}_\Omega(x_0) = \mathsf{C}_\mathcal{N}(x_0)$.

Preuve Soit x_0 le point de $\mathbf{C}$ qui a le plus petit coût $\mathsf{C}_\Omega(x_0)$. Par définition $\mathsf{C}_\Omega(x_0) < \mathsf{C}_\Omega(x_1)$. Soit $\mathcal{C}_{t+dt}$ la ligne de niveau du coût C_Ω associée à la valeur $\mathsf{C}_\Omega(x_0)$. Nous avons donc $\mathcal{C}_t \subset \mathcal{C}_{t+dt} \subset \mathcal{N}$, donc par la Proposition 1, nous avons $\mathsf{C}_\Omega(x_0) = \mathsf{C}_\mathcal{N}(x_0)$. Soit x un point de $\mathbf{C}$. Par l'hypothèse (H2), nous avons $\mathsf{C}_\mathcal{N}(x) \geq \mathsf{C}_\Omega(x)$. Par définition de x_0, $\mathsf{C}_\Omega(x) \geq \mathsf{C}_\Omega(x_0)$. Donc, $\mathsf{C}_\mathcal{N}(x) \geq \mathsf{C}_\mathcal{N}(x_0)$. □

Ainsi du point de vue numérique, si (H1), (H2) et (H3) sont vérifiées, alors pour étendre Ω_t à Ω_{t+dt}, nous avons juste à calculer les coûts $\mathsf{C}_\mathcal{N}(x)$ pour tout x dans $\mathbf{C}$ et à chercher x_0 dans $\mathbf{C}$ qui ait le plus petit coût $\mathsf{C}_\mathcal{N}$.

4.3 Coût proposé

Rappelons que la méthode que nous proposons ici nécessite le calcul explicite du coût C et de la solution V. Vu que notre objectif final est uniquement de calculer des approximations de V, le calcul additionnel du coût C peut paraître inutile et peut donc diminuer fortement l'intérêt de la méthode. Il semble alors raisonnable et pertinent de chercher et d'utiliser des coûts $\mathsf{C}_\mathcal{W}$ qui sont directement liés aux valeurs de la solution $V_\mathcal{W}$, comme dans le cas Eikonal et dans les FMM classiques (voir les Exemples 2 et 4). Ainsi, ce surcoût potentiel n'apparaît pas.

Exemple 5 *Dans [33], Sethian et Vladimirsky considèrent des équations du type* (15), *c'est-à-dire des équations* (5) *avec* $\lambda = 0$, $f(x,a) = \mathsf{f}(x,a)a$ *et* $l(x,a) = 1$. *Donc la solution de viscosité* V *de* (15) *(complétée par des conditions au bord adéquates) est* $V(x) = \tau_x^* + \varphi(y_x^*(\tau_x^*))$. *Ainsi, si la condition au bord* φ *imposée sur* $\mathcal{T}$ *est une fonction constante (*$\varphi(x) = c \in \mathbb{R}$ *pour tout* x *dans* $\mathcal{T}$*) alors* $V(x) = \tau_x^* + c$. *Dans ce cas, il est donc judicieux de choisir* $\mathsf{C}_\mathcal{W}(x) = \tau^*_{x(\mathcal{W})}$. *Ainsi,* $V_\mathcal{W}$ *et* $\mathsf{C}_\mathcal{W}$ *sont donc directement reliés. Par conséquent les courbes* $\mathcal{C}_t$ *correspondent aux lignes de niveau de la solution de* (15). *Notons par ailleurs que les coûts* $\mathsf{C}_\mathcal{W}$ *vérifient trivialement les hypothèses (H1), (H2) et (H3).*

Dans le cas particulier considéré par Sethian et Vladimirsky [33] (Exemple 5), le coût directement relié aux valeurs de la solution est trivial. Cependant dans le cas général, en particulier pour les équations de HJB avec une fonction coût $l(x,a)$ dépendant de a ou avec un signe arbitraire, nous devons travailler un peu plus.

De manière générale, dès que les valeurs de la solution ne sont *pas décroissantes* le long des trajectoires optimales (c'est-à-dire pour tout x, la fonction $t \mapsto u(y_x^*(t))$ n'est pas décroissante), les lignes de niveau de la solution ne peuvent pas jouer le rôle des courbes $\mathcal{C}_t$. C'est en particulier le cas lorsque la fonction coût l associé à l'Hamiltonien est de signe arbitraire.
Une seconde idée basique pourrait être de choisir systématiquement le coût $\mathsf{C}(x) = \tau_x^*$ vu que ce coût vérifie naturellement les hypothèses (H1), (H2) et (H3). Cependant, calculer ce coût peut s'avérer difficile, et généralement celui-ci n'est pas en rapport avec la solution u de l'équation considérée.
De manière à définir un coût adapté au cas général, introduisons les définitions suivantes. Par souci de simplicité, nous supposons que $\lambda = 0$, mais ce qui suit peut être aisément étendu au cas général (voir notre rapport technique [29]).
Soit ψ une sous-solution de (4), c'est-à-dire

$$H(x, \psi(x)) \leq 0, \quad \forall x \in \Omega.$$

Dans [23], Prados et Faugeras détaillent les sous-solutions des principales équations classiques de SFS. Par exemple, pour l'Hamiltonien de Rouy/Tourin $H_{R/T}$ (équation (2))

$$\psi(x) = -\frac{1}{\gamma}\mathbf{1} \cdot x$$

est une sous-solution.
Soit $\mathcal{Z}(x)$ la carte multivaluée (associée à l'Hamiltonien) [5] définie sur $\overline{\Omega}$ par :

$$\mathcal{Z}(x) = \{p \in \mathbb{R}^N : H(x,p) \leq 0\}.$$

Soit $\delta : \overline{\Omega} \times \mathbb{R}^N \to \mathbb{R}$ la fonction support [5] de l'ensemble $\tilde{\mathcal{Z}}(x) = \mathcal{Z}(x) - \nabla\psi(x)$, c'est-à-dire :

$$\delta(x,p) = \max\{pq : q \in \tilde{\mathcal{Z}}(x)\}.$$

Finalement, pour tout x_1, x_2 dans $\overline{\mathcal{W}}$, notons

$$L_\mathcal{W}(x_1,x_2) = \inf_{\xi \in \Xi_{x_1,x_2}} \left\{ \int_0^1 \delta(\xi(t), -\dot{\xi}(t))dt \right\}$$

où Ξ_{x_1,x_2} est l'ensemble des $\xi(t) \in W^{1,\infty}([0,1], \overline{\mathcal{W}})$ tels que $\xi(0) = x_1$ et $\xi(1) = x_2$.
Une étude et une description complète de ces notions se trouvent dans [24, 23] et [5]. En particulier, dans [24, 23], il est démontré que

$$V(x) = \psi(x) + \min\{L_{\Omega\setminus\mathcal{T}}(x,z) + \varphi(z) - \psi(z) \mid z \in \mathcal{T}\} \quad (17)$$

est l'unique Solution de Viscosité Singulière Discontinue (SDVS) de (5)-(6)-(7), voir [23]. Insistons ici sur le fait que $\delta(.,.)$ et donc $L_{\mathcal{W}}$ sont positifs. Par ailleurs, $L_{\mathcal{W}}(x,z)$ est une semi-distance.
Ces résultats et notations en main, nous pouvons maintenant définir un coût approprié pour notre équation de HJB "générique" :

$$\mathsf{C}_{\mathcal{W}}(x) = L_{\mathcal{W}}(x, z_x^*),$$

où z_x^* est le z optimal de (17). Ce coût est clairement décroissant le long des trajectoires optimales et il vérifie naturellement les hypothèses (H1)-(H2)-(H3). Finalement, si la condition au bord φ vérifie "$\varphi - \psi$ est constant sur la cible $\mathcal{T}$" alors la solution de viscosité $V_{\mathcal{W}}$ est directement reliée à $\mathsf{C}_{\mathcal{W}}$: elle coïncide avec $\mathsf{C}_{\mathcal{W}} + \psi$ (à une constante près). Ainsi, les lignes de niveau du coût C (i.e. $\mathcal{C}_t$), correspondent avec les lignes de niveau de $V - \psi$.

5 Importance des trajectoires optimales stationnaires

Bien que quelques travaux essaient d'affronter la difficulté [24, 20], la prise en compte d'éventuelles trajectoires optimales stationnaires est généralement soigneusement évitée autant du point de vue théorique que pratique. Par exemple, les hypothèses (très restrictives) supposées par Sethian et Vladimirsky dans [33] ne permettent pas de traiter les cas où il existe de telles trajectoires. Ces cas sont cependant très fréquents et fondamentaux en pratique. En effet, dans les cas où $\lambda = 0$, les points où la trajectoire optimale est stationnaire correspondent aux points singuliers. En SFS, ces points correspondent aux points d'intensité maximale $I(x) = 1$. Ces points apparaissent donc tout naturellement dès que la direction du vecteur normal à la surface photographiée coïncide avec celle de la lumière ; voir [24] pour une étude complète de ces singularités en SFS pour les cas où $\lambda = 0$. Dans le cas où $\lambda > 0$, l'importance de ces trajectoires est encore plus cruciale...
Supposons donc temporairement $\lambda > 0$. Supposons par ailleurs que pour tout $x \in \overline{\Omega}$ il existe un contrôle α_0^x tel que $f(x, \alpha_0^x) = 0$ (ceci est toujours vrai pour les équations de SFS). Notons alors

$$V_\infty(x) = F^{-1}\left(\frac{1}{\lambda} l(x, \alpha_0^x)\right).$$

Par construction, V_∞ est une supersolution de l'équation (5). Sous l'hypothèse d'unicité forte (c.f. [23] pour une étude complète dans le cadre du SFS), la solution V de l'EDP considérée est alors inférieure à V_∞. Nous pouvons alors prouver facilement que

$$V(x) = \min\left(V_\infty(x), \inf_{\alpha \in A - \{\alpha_0^x\}} \left\{ \int_0^{\tau_x^{\mathcal{N}}(\alpha)} l(y_x(\tau,\alpha), \alpha(\tau)) e^{-\lambda\tau} d\tau \; + V(y_x(\tau_x^{\mathcal{N}}(\alpha), \alpha)) e^{-\lambda \tau_x^{\mathcal{N}}(\alpha)} \right\}\right), \quad (18)$$

Ainsi, contrairement au cas où $\lambda = 0$, la valeur de la solution aux points stationnaires est connue "d'avance" et n'a pas besoin d'être fixée artificiellement. La valeur des minima locaux n'a plus a être spécifiée. Ceci est d'ailleurs directement relié aux résultats théoriques présentés dans [28]. Notons enfin que les conséquences pratiques et algorithmiques sont très importantes. Par exemple, lors de la propagation du front, des points non connectés au front peuvent arriver à avoir la même valeur que la "ligne de niveau" et ainsi passer dans l'ensemble des points *Accepté* ; c.f. section 6 et [29] pour plus de détails.
Notation : Dans le cas où $\lambda = 0$, nous posons $V_\infty(x) = +\infty$.

6 Algorithme proposé

Nous subdivisons la grille de points en trois classes (comme dans les méthodes basiques de "Fast Marching") : *Accepté* **A**, *Considéré* **C**, *Eloigné* **F** ("Far"). ψ est une sous-solution de $H(x, \nabla u) = 0$ comme définie dans la Section 4.3. U définie sur la totalité de la grille est l'approximation de la solution de l'équation (5)-(6)-(7).

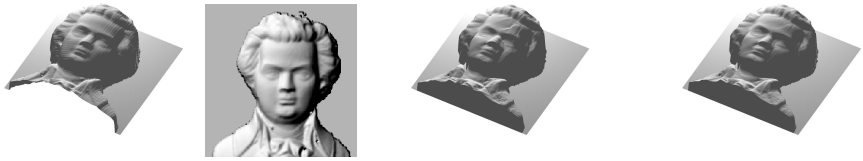
Algorithme 1

1. *Commencer avec tous les points de la grille dans* **F** *et poser* $U(x) = V_\infty(x)$ *(*V_{infty} *est définie dans la section 5).*
2. *Déplacer les points de la grille situés sur le bord* $\partial\Omega$ *et sur la cible* $\mathcal{T}$ *dans* **A**. *Pour tous ces points, poser* $U(x) = \varphi(x)$ *(*$U(x) = V_\infty(x)$*, dans le cas de contraintes d'état).*
3. *Déplacer tous les points de la grille* $x \in$ **F** *adjacents aux points* Acceptés *dans* **C** *et pour chacun de ces points, réévaluer* $U(x)$ *en utilisant le schéma de mise à jour* (14).
4. *Trouver le point de la grille* $x_0 \in \mathbf{C} \cup \mathbf{F}$ *qui a la plus petite valeur* $U(x) - \psi(x)$. *Déplacer* x_0 *dans* **A**.
5. *Déplacer tous les points* Eloignés *adjacents à* x_0 *dans* **C**.
6. *Réévaluer* U *en utilisant le schéma* (14), *pour tous les points* Considérés *adjacents à* x_0.
7. *Si l'ensemble* **C** *n'est pas vide, retourner à l'étape 4.*

Remarque : dans l'étape 4, dans le cas où $\lambda = 0$, la recherche de la plus petite valeur se réduit à l'ensemble **C**. Par ailleurs, de manière générale, il est possible de formuler l'algorithme en subdivisant la grille en seulement deux classes, au lieu de trois.

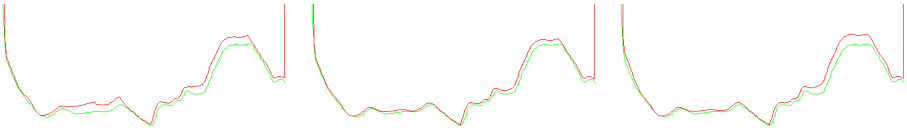
7 Résultats expérimentaux

Nous avons appliqué notre méthode aux "équations de SFS génériques" de Prados et Faugeras [23] et à l'équation (3) qui dépend de u ($\lambda \neq 0$). Considérons tout d'abord l'application aux équations de SFS de type $H(x, \nabla u(x)) = 0$ avec une fonction coût de signe arbitraire. Rappelons que dans ce cadre (Hamiltonien ne dependant pas de u), comme le suggère la théorie [24], nous devons supposer que nous connaissons la valeur de la solution à tous ses minima locaux et que nous devons utiliser ces valeurs sous la forme de conditions aux limites de Dirichlet. Pour ces équations, la contribution principale repose sur l'introduction de la nouvelle causalité. Cette contribution étant essentiellement



a) b) c) d)

FIG. 1 – a) surface originale ; b) image synthétisée à partir de a) ; c) surface reconstruite en utilisant la causalité classique ; d) surface reconstruite en utilisant la nouvelle causalité.



a) b) c)

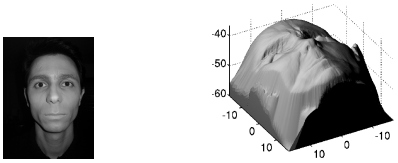
FIG. 2 – a) coupe verticale de Fig.1-c) ; b) coupe verticale de Fig.1-d) ; c) coupe verticale de la surface reconstruite par l'algorithme itératif après trois itérations complètes.

algorithmique (pas de contribution au niveau de la modélisation du problème du SFS), nous illustrons nos résultats seulement avec l'Hamiltonien classique de Rouy et Tourin $H_{R/T}$ (2) sur l'exemple classique du visage de Mozart [40]. La Figure Fig.1 montre : a) la surface originale du visage de Mozart ; b) l'image synthétisée à partir de a) suivant le modèle de Rouy et Tourin avec $\mathbf{l} = (-0.3, -0.3)$; c) la surface reconstruite en utilisant la causalité classique [notre schéma + replacement de ψ par 0] ; d) la surface reconstruite par notre algorithme [nouveau schéma + nouvelle causalité]. Même si la présence d'ombres portées, nous empêche de reconstruire parfaitement la surface originale, l'amélioration apportée par l'introduction de la nouvelle causalité est clairement visible. Les coupes verticales Fig.2-a) et Fig.2-b) de ces deux reconstructions mettent en evidence leur différences. Dans cette figure (Fig.2), les courbes rouges correspondes aux coupes des approximations calculées et la courbe verte à celle de la surface originale. La figure 2-c) affiche le résultat retourné par l'algorithme itératif [26, 23] après trois scans. Pour cette image qui contient approximativement 20200 pixels, notre FFM effectue autour de 40500 mises à jour ("updates"). La version itérative nécessite environ 76000 mises à jour pour approcher la solution avec une erreur de même ordre de grandeur.

La Figure 3 illustre enfin un exemple de reconstruction à partir d'une image réelle pour l'Hamiltonien (3)

$$-e^{-2u(x)} + H_{att}(x, \nabla u(x)) = 0$$

qui dépend de u. Contrairement à l'équation précédente, pour cette dernière équation, nous n'avons plus besoin de fournir de conditions aux limites de type Dirichlet et l'algorithme ne nécessite pas la définition d'un front de propagation initial.



a) image originale b) reconstruction

FIG. 3 – Un exemple de reconstruction avec l'Hamiltonien (3) qui dépend de u.

Nous référons le lecteur à notre rapport technique [29] et à notre article de journal en préparation pour plus de résultats expérimentaux.

8 Conclusion

Dans cet article, nous revisitons les méthodes de "Fast Marching" [38, 31, 33] et nous les étendons à une large classe d'équations de HJ. En particulier, notre méthode peut traiter les équations de HJB dépendant de u et avec des fonctions coûts l de signes arbitraires (alors que les méthodes précédentes traitaient juste les équations de la forme $H(x, \nabla u) = 0$, avec des fonctions coûts l positives). Dans les cas où la solution n'est pas décroissante le long des trajectoires optimales, nous corrigeons la propriété de causalité en utilisant une subsolution adéquate. Contrairement aux précédentes méthodes de "Fast Marching", notre méthode s'applique indifféremment aux diverses équations de "Shape From Shading". En particulier elle s'applique aux équations de SFS correspondant à des scènes illuminées par une source de *lumière non frontale*. Par ailleurs, contrairement par exemple à [17], notre méthode ne nécessite aucun changement de variables. Enfin, notre méthode s'applique aussi aux très récentes modélisations introduites dans [27] ; modélisations dont la pertinence théorique, algorithmique mais aussi pratique, a déjà été démontrée.

Références

[1] G. Amos. *Solving the Hamilton-Jacobi-Bellman Equation for Animation*. PhD thesis, Univ. of London., 2002.

[2] M. Bardi and I. Capuzzo-Dolcetta. *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations*. Birkhauser, 1997.

[3] G. Barles and P.E. Souganidis. Convergence of approximation schemes for fully nonlinear second order equations. *Asymptotic Analysis*, 4 :271–283, 1991.

[4] A. M. Bruckstein. On shape fron shading. *Computer. Vision Graphics Image Process*, 44 :139–154, 1988.

[5] F. Camilli and A. Siconolfi. Nonconvex degenerate Hamilton-Jacobi equations. *Mathematische Zeitschrift*, 242 :1–21, 2002.

[6] L. Cohen. Minimal paths and fast marching methods for image analysis. In N. Paragios, Y. Chen, and O. Faugeras, editors, *Mathematical Models in Computer Vision : The Handbook*, chapter 7. Springer, 2005.

[7] F. Courteille, A. Crouzil, J-D. Durou, and P. Gurdjos. Towards shape from shading under realistic photographic conditions. In *Proceedings of ICPR'04*, 2004.

[8] M.G. Crandall and P.-L. Lions. Viscosity solutions of Hamilton–Jacobi equations. *Trans. AMS*, 277 :1–43, 1983.

[9] E. Cristiani and M. Falcone. Fast Marching Semi-Lagrangian methods for the Eikonal equation. In *SIMAI 2004*, pages 20–24, Venezia, Italy, September 2004.

[10] P.-E. Danielsson. Euclidean Distance Mapping. *Computer Graphics and Image Processing*, 14(3) :227–248, 1980.

[11] E. Dejnozkova. *Architecture dédiée au traitement d'image base sur les équations aux dérivées partielles*. PhD thesis, École des Mines de Paris, 2004.

[12] E. Dejnozkova and P. Dokladal. A parallel architecture for curve-evolution pdes. *Image Analysis and Stereology*, 22, June 2003.

[13] B.K. Horn. *Robot Vision*. MIT Press, 1986.

[14] B.K.P. Horn. Obtaining shape from shading information. In P.H. Winston, editor, *The Psychology of Computer Vision*. McGraw-Hill, New York, 1975.

[15] S. Kim. An o(n) level set method for eikonal equations. *SIAM J. Sci. Comput.*, 22(6) :2178–2193, 2000.

[16] R. Kimmel and A.M. Bruckstein. Tracking level sets by level sets : A method for solving the shape from shading problem. *CVIU*, 62(2) :47–58, July 1995.

[17] R. Kimmel and J.A. Sethian. Optimal algorithm for shape from shading and path planning. *Journal of Mathematical Imaging and Vision*, 14(2) :237–244, May 2001.

[18] P.-L. Lions. *Generalized Solutions of Hamilton–Jacobi Equations*. Number 69 in Research Notes in Mathematics. Pitman Advanced Publishing Program, 1982.

[19] V. Naicker, K. Andriopoulos, and P.G.L. Leach. Symmetry reductions of a Hamilton-Jacobi-Bellman equation arising in financial mathematics. *J. of Nonlinear Math. Physics*, 12(2) :268–283, 2005.

[20] J. Oliensis. Shape from shading as a partially well–constrained problem. *CVGIP : Image Understanding*, 54(2) :163–183, 1991.

[21] J. Oliensis and P. Dupuis. Direct method for reconstructing shape from shading. *SPIE*, 1570 :116–128, 1991.

[22] S. Osher and J. Sethian. Fronts propagating with curvature dependent speed : algorithms based on the Hamilton–Jacobi formulation. *Journal of Comput. Physics*, 79 :12–49, 1988.

[23] E. Prados. *Application of the theory of the viscosity solutions to the Shape From Shading problem*. PhD thesis, Univ. of Nice-Sophia Antipolis, 2004.

[24] E. Prados, F. Camilli, and O. Faugeras. A unifying and rigorous shape from shading method adapted to realistic data and applications. *To appear in JMIV*, 2005.

[25] E. Prados and O. Faugeras. "Perspective Shape from Shading" and viscosity solutions. In *Proceedings of ICCV'03*, volume 2, pages 826–831, October 2003.

[26] E. Prados and O. Faugeras. A generic and provably convergent shape-from-shading method for orthographic and pinhole cameras. *à paraître dans IJCV*, 2005.

[27] E. Prados and O. Faugeras. Shape from shading : A well posed problem ? In *Proceedings of CVPR'05*, 2005.

[28] E. Prados and O. Faugeras. Rôle clé de la modélisation en shape from shading. In *RFIA'06*, 2006.

[29] E. Prados, S. Soatto Et al. Rapport Technique à paraître ; Article de journal en préparation. 2005/2006.

[30] E. Rouy and A. Tourin. A Viscosity Solutions Approach to Shape-from-Shading. *SIAM Journal of Numerical Analysis*, 29(3) :867–884, June 1992.

[31] J.A. Sethian. A fast marching sevel set method for monotonically advancing fronts. In *Proceedings of the National Academy of Sciences*, volume 93, pages 1591–1694, 1996.

[32] J.A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, 1999.

[33] J.A. Sethian and A. Vladimirsky. Ordered upwind methods for static hamilton–jacobi equations : Theory and algorithms. *SIAM J. on Num. Anal.*, 41(1), 2003.

[34] A. Tankus, N. Sochen, and Y. Yeshurun. A new perspective [on] Shape-from-Shading. In *Proceedings of ICCV'03*, volume 2, pages 862–869, October 2003.

[35] A. Tankus, N. Sochen, and Y. Yeshurun. Perspective Shape-from-Shading by Fast Marching. In *CVPR'04*, 2004.

[36] F. Trojani and P. Vanini. Perturbative solutions of Hamilton Jacobi Bellman equations in robust decision making. In *European Summer Symposium in Financial Markets*, 2002.

[37] Y.-H. Tsai, L.-T. Cheng, S. Osher, and H.-K. Zhao. Fast sweeping algorithms for a class of hamilton-jacobi equations. *SIAM J. Numer. Anal.*, 41(2) :673–694, 2003.

[38] J.N. Tsitsiklis. Efficient algorithms for globally optimal trajectories. *IEEE Trans. on Automatic Control*, 40 :1528–1538, 1995.

[39] S.Y. Yuen, Y.Y. Tsui, Y.W. Leung, and R.M.M. Chen. Fast marching method for shape from shading under perspective projection. In *VIIP'02*, pages 584–589, 2002.

[40] R. Zhang, P.-S. Tsai, J.-E. Cryer, and M. Shah. Shape from Shading : A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8) :690–706, August 1999.