

Deformation de Courbes Multiresolution sous Contraintes

Stefanie Hahmann, Georges-Pierre Bonneau, Basile Sauvage

► **To cite this version:**

Stefanie Hahmann, Georges-Pierre Bonneau, Basile Sauvage. Deformation de Courbes Multiresolution sous Contraintes. Revue internationale d'ingénierie numérique, 2005, 1 (1), pp.77-90. hal-00387041

HAL Id: hal-00387041

<https://hal.archives-ouvertes.fr/hal-00387041>

Submitted on 27 Apr 2012

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.

Déformation de courbes multirésolution sous contraintes

Stefanie Hahmann*, Georges-Pierre Bonneau†, Basile Sauvage*

*Laboratoire LMC-IMAG, BP.53
F-38041 Grenoble cedex 9, France

email: Stefanie.Hahmann@imag.fr, Basile.Sauvage@imag.fr

†Laboratoire iMAGIS-GRAVIR
INRIA Rhône-Alpes

655, avenue de l'Europe

F-38330 Montbonnot, France

email: Georges-Pierre.Bonneau@imag.fr

Abstract. *Nous décrivons une méthode de déformation multirésolution de courbes planes fermées qui conserve l'aire intérieure. Nous utilisons une représentation multirésolution basée sur des ondelettes, et les courbes sont représentées par un nombre fini de points de contrôle à chaque échelle. Il est alors possible de déformer la courbe en modifiant un ou plusieurs points de contrôle à une échelle quelconque. Ce processus est connu sous le nom d'édition multirésolution, auquel nous ajoutons la contrainte d'aire constante. Nous développons aussi une expression multirésolution du moment d'aire. Nous nous assurons que tous les calculs sont rapides et que les déformations peuvent être calculées interactivement.*

Keywords. analyse multirésolution, déformation, contrainte d'aire, courbes de subdivision, ondelettes, aire constante, courbes fermées.

1. INTRODUCTION

L'analyse multirésolution a reçu une attention particulière ces dernières années dans de nombreux domaines de la géométrie algorithmique, de la modélisation géométrique et de la visualisation. Elle apporte un outil puissant pour représenter efficacement des fonctions à différents niveaux de détail. Dans ce cadre, une fonction est représentée par une "tendance grossière" qui contient l'information basse résolution, couplée avec une série de coefficients de détails qui permettent la reconstruction exacte du signal original.

Des représentations multirésolution basées sur les ondelettes ont été développées pour les courbes paramétriques et peuvent être généralisées aux surfaces produit tensoriel, aux surfaces de topologie arbitraire, aux données sphériques et volumiques. D'autres modélisations multirésolution (MR) existent pour des données définies sur des maillages irréguliers.

Toutes ces techniques montrent les nombreux intérêts que présente la modélisation multirésolution. La déformation d'objets complexes avec une grande quantité de détails s'avère souvent difficile et coûteuse en calculs. Dans un cadre multirésolution, de tels objets peuvent être édités à une échelle précise, ce qui permet principalement deux effets: premièrement la modification de points de contrôle à basse résolution suivie de la réintégration des détails modifie la forme globale de l'objet sans en changer les détails caractéristiques. Deuxièmement, la modification de détails fins modifie le caractère de la courbe sans changer sa forme globale.

L'édition multirésolution et les techniques de déformations de courbes paramétriques ont été étudiées en détail par Finkelstein et Salesin [9] avec l'utilisation des B-splines. Mais l'édition multirésolution sous contraintes n'a fait l'objet que de peu de recherches, bien que de nombreux domaines tels que la CAD/CAM ou l'informatique graphique ont besoin de tels outils. Des travaux utilisant l'intégration numérique d'équations différentielles physiques ont été effectués, mais dans les cas où les temps de calculs sont prioritaires sur le réalisme, l'utilisation de contraintes géométriques est un avantage certain. D'autres domaines d'application sont envisageables, tels qu'en modélisation géométrique ou en vision par ordinateur.

Nous présentons ici un environnement d'édition et de déformation de courbes multirésolution qui permet la conservation de l'aire enfermée dans une courbe plane périodique, contrainte non linéaire. Des travaux ont déjà été faits dans ce sens par G. Elber [6] concernant la linéarisation de la contrainte d'aire par décomposition du déplacement en de nombreux pas en x et en y . Dans [7] il développe une version multirésolution de son algorithme en utilisant l'insertion de noeuds, ce qui permet de localiser les déformations. La particularité de notre travail est de présenter un cadre multirésolution uniforme permettant la décomposition et l'utilisation de différents schémas d'ondelettes. Les formules que nous présentons peuvent s'appliquer à tous les schémas multirésolution basés sur des ondelettes. Pour la mise en pratique nous utilisons le schéma de subdivision de Chaïkin, étroitement lié aux ondelettes B-spline quadratiques. Notons enfin que tous nos algorithmes ont été développés dans un souci constant de simplicité et d'efficacité pour en faire un outil interactif.

Cet article est organisé comme suit: la section 2 rappelle les idées de base de la multirésolution. La section 3 explique comment calculer efficacement l'aire enfermée par une courbe définie dans une base multirésolution. La section 4 présente notre méthode de déformation de courbes sous contrainte d'aire. La section 5 conclut et donne des ouvertures sur des travaux futurs.

2. Courbes Multirésolution

Durant les dernières années les ondelettes ont trouvé des applications dans des domaines très variés : analyse numérique [1], analyse du signal [13], traitement d'images [12], visualisation et informatique graphique. Le schéma multirésolution basé sur des ondelettes que nous allons présenter brièvement dans ce chapitre est légèrement plus général que celui introduit dans [12]. Pour plus de détails, voir [16, 9].

Soit E un espace fonctionnel et $V^j \subset E$ des *espaces d'approximation* linéaires imbriqués avec $V^0 \subset V^1 \subset \dots \subset V^n$. Dans le cadre de cet article nous allons nous intéresser aux courbes fermées, ces espaces seront donc de dimension finie. Soit V^j engendré par un ensemble de fonctions de base $\varphi^j = [\varphi_1^j, \dots, \varphi_m^j]^T$, appelées *fonctions d'échelle*. L'espace W^j , complémentaire de V^j dans V^{j+1} , est appelé *espace de détails*. Ses fonctions de base $\psi^j = [\psi_1^j, \dots, \psi_{N-m}^j]^T$ couplées avec les φ^j forment une base de V^{j+1} . Les fonctions ψ_i^j sont appelées *ondelettes*. L'espace V^n peut donc être décomposé comme suit:

$$V_n = V_{n-1} \bigoplus W_{n-1} = V_{n-2} \bigoplus_{j=n-2}^{n-1} W_j = \dots = V_0 \bigoplus_{j=0}^{n-1} W_j. \quad (1)$$

La condition (1) implique que les fonctions d'échelle sont subdivisibles, i.e. pour tout $j \in [1, n]$ il doit exister des matrices P^j et Q^j telles que les *équations de subdivision* suivantes sont vérifiées

$$\begin{aligned} \varphi^{j-1} &= (P^j)^T \varphi^j \\ \psi^{j-1} &= (Q^j)^T \varphi^j. \end{aligned} \quad (2)$$

D'autre part, les fonctions d'échelle "fines" φ^j peuvent être construites à partir des fonctions d'échelles "grossières" et des ondelettes à l'aide de matrices A^j et B^j :

$$\varphi^j = (A^j)^T \varphi^{j-1} + (B^j)^T \psi^{j-1}. \quad (3)$$

Notons que $[P^j \mid Q^j]$ et $\begin{bmatrix} A^j \\ B^j \end{bmatrix}$ sont des matrices carrées, et que

$$\begin{bmatrix} P^j & Q^j \end{bmatrix} \begin{bmatrix} A^j \\ B^j \end{bmatrix} = I. \quad (4)$$

(4) est généralement appelée *condition de reconstruction*. Le choix des fonctions d'échelle détermine la structure des matrices P^j, Q^j, A^j , et B^j . Pour la plupart des applications il est préférable que ces matrices soient creuses, ce qui est vrai pour les ondelettes à support

compact.

A partir de ce schéma multirésolution, soit une *courbe multirésolution* $\mathbf{c}(t)$, $t \in I$ définie comme élément d'un certain espace fonctionnel V^n de dimension $D2^n$. $\mathbf{c}(t)$ peut être représentée comme suit:

$$\mathbf{c}(t) = \sum_{i=0}^{D2^n-1} x_i^n \varphi_i^n = (\mathbf{x}^n)^T(\boldsymbol{\varphi}^n), \quad (5)$$

où \mathbf{x}^n est un vecteur colonne de $D2^n$ points de contrôle $x_0^n, \dots, x_{D2^n-1}^n \in \mathbb{R}^2$. De manière générale le vecteur \mathbf{x}^n peut être un signal avec $n = D2^n$ échantillons.

Les relations (2) et (3) permettent maintenant de créer un signal basse résolution \mathbf{x}^{n-1} avec moins d'échantillons en utilisant le filtre passe bande A^n :

$$\mathbf{x}^{n-1} = A^n \mathbf{x}^n,$$

où A^n est de taille $(D2^{n-1} \times D2^n)$. Les détails qui ont été perdus dans ce filtrage peuvent être récupérés dans un autre signal \mathbf{d}^{n-1} en utilisant un filtre passe haut B^n :

$$\mathbf{d}^{n-1} = B^n \mathbf{x}^n,$$

où B^n est une matrice $(D2^{n-1} \times D2^n)$. Ce processus de séparation d'un signal \mathbf{x}^n en un signal grossier \mathbf{x}^{n-1} et des détails \mathbf{d}^{n-1} est appelé **décomposition** ou *analyse*. Les matrices A^n et B^n s'appellent *filtres de décomposition (d'analyse)*. La décomposition peut être répétée récursivement sur le nouveau signal \mathbf{x}^{n-1} . Finalement, le signal original sera décomposé en un signal basse résolution \mathbf{x}^0 et des détails $\mathbf{d}^0, \dots, \mathbf{d}^{n-1}$. Ce processus récursif est connu sous le nom *filter bank* [12], voir figure 1.

Le signal original \mathbf{x}^n peut être reconstruit en utilisant les *matrices de reconstruction (de synthèse)* P^j et Q^j (2)

$$\mathbf{x}^j = P^j \mathbf{x}^{j-1} + Q^j \mathbf{d}^{j-1} \quad \text{pour } j = 1, \dots, n.$$

Ce processus s'appelle **reconstruction**.

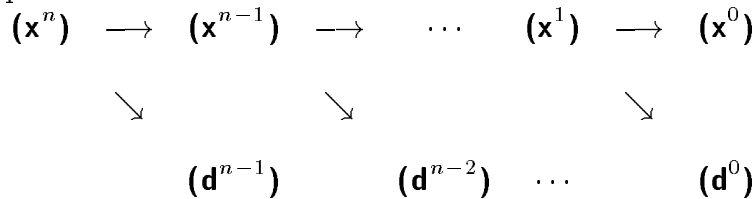


Figure 1: L'algorithme filter bank.

Une courbe multirésolution (5), peut être représentée à un certain *niveau de résolution* $L \in [0, n]$ par des points de contrôle grossiers \mathbf{x}^L qui forment une approximation du

polygone de contrôle initial et des coefficients de détail $\mathbf{d}^L, \dots, \mathbf{d}^{n-1}$ en appliquant le filter bank. On obtient:

$$\mathbf{c}(t) = (\mathbf{x}^L)^T(\varphi^L) + (\mathbf{d}^L)^T(\psi^L) + \dots + (\mathbf{d}^{n-1})^T(\psi^{n-1}), \quad L = 0, \dots, n. \quad (6)$$

Tout ce qui sera développé dans cet article est indépendant de la représentation particulière choisie pour la courbe multirésolution. Les exemples de résultats ont été obtenus en implémentant les ondelettes de Chaïkin. Elles sont basées sur le schéma de subdivision de Chaïkin [2] et sont également connues comme ondelettes B-spline quadratiques. La décomposition d'une courbe à $D2^n$ points de contrôle $x_0^n, \dots, x_{D2^n-1}^n$ est donnée par

$$\begin{aligned} x_i^{n-1} &= \frac{1}{4}(-x_{2i-2}^n + 3x_{2i-1}^n + 3x_{2i}^n - x_{2i+1}^n) \\ d_i^{n-1} &= \frac{1}{4}(x_{2i-2}^n - 3x_{2i-1}^n + 3x_{2i}^n - x_{2i+1}^n), \end{aligned} \quad (7)$$

où tous les indices sont pris modulo le nombre de points de contrôle. La formule de construction pour Chaïkin est:

$$\begin{aligned} x_{2i}^n &= \frac{3}{4}(x_i^{n-1} + d_i^{n-1}) + \frac{1}{4}(x_{i+1}^{n-1} - d_{i+1}^{n-1}) \\ x_{2i+1}^n &= \frac{1}{4}(x_i^{n-1} + d_i^{n-1}) + \frac{3}{4}(x_{i+1}^{n-1} + d_{i+1}^{n-1}). \end{aligned} \quad (8)$$

La figure 2 illustre les approximations successives d'une courbe de subdivision de Chaïkin obtenues par la décomposition (7).

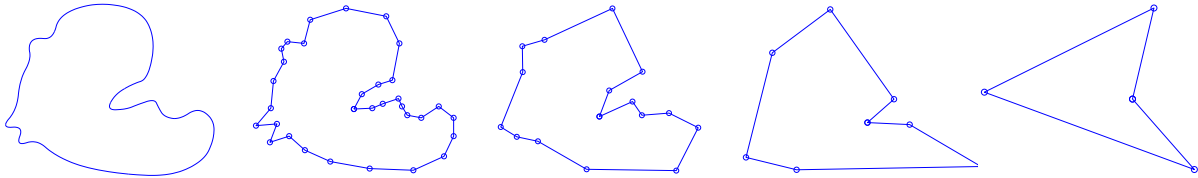


Figure 2: De gauche à droite: une courbe de Chaïkin définie par 32 points de contrôle, son polygone de contrôle (niveau 5), puis les polygones de contrôle des approximations correspondant au niveaux 4, 3 et 2.

3. Calcul de l'aire d'une courbe multirésolution

Nous allons dans ce chapitre développer un outil de déformation préservant l'aire enfermée dans une courbe multirésolution plane périodique. Pour cela il est nécessaire de pouvoir évaluer l'aire de la courbe à n'importe quel niveau de résolution. La section (3.1) présente les formules multirésolution de la fonction d'aire. La section (3.2) explique comment le calcul peut être effectué de manière efficace en développant une relation de récurrence.

3.1 L'aire enfermée dans une courbe multirésolution

Soit $\mathbf{c}(t) = (x(t), y(t))$ une courbe paramétrique plane et périodique sans auto-intersections et suffisamment régulière pour que toutes les dérivées nécessaires puissent être définies. L'aire incluse dans $\mathbf{c}(t)$ est donnée par le théorème de Green [6, 5]:

$$\mathcal{A} = \frac{1}{2} \oint x(t)y'(t) - x'(t)y(t)dt. \quad (9)$$

Quelques notations sont nécessaires pour la suite. Introduisons la forme bilinéaire

$$I(\varphi, \psi) := \begin{pmatrix} I(\varphi_i, \psi_j)_{ij} \end{pmatrix}, \quad (10)$$

où φ et ψ sont deux vecteurs de fonctions et $I(\varphi_i, \psi_j) = \oint \varphi_i(t)\psi'_j(t) - \varphi'_i(t)\psi_j(t)dt$.

L'aire (9) d'une courbe multirésolution (6) peut maintenant être évaluée à n'importe quel niveau de résolution L . On obtient l'équation bilinéaire

$$2\mathcal{A} = (\mathbf{X}^L) \begin{bmatrix} M^L \end{bmatrix} (\mathbf{Y}^L)^T, \quad \forall L \in \{0, \dots, n\}, \quad (11)$$

où X^L et Y^L sont des vecteurs ligne contenant respectivement les coordonnées x et y des $D2^n$ coefficients de la représentation multirésolution de la courbe, i.e.

$$\begin{pmatrix} \mathbf{X}^L \\ \mathbf{Y}^L \end{pmatrix} = (\mathbf{x}^L, \mathbf{d}^L, \mathbf{d}^{L+1}, \dots, \mathbf{d}^{n-1}), \quad (12)$$

et

$$M^L = \begin{bmatrix} I(\varphi^L, \varphi^L) & I(\varphi^L, \psi^l)_{l=L}^{n-1} \\ I(\psi^k, \varphi^L)_{k=L}^{n-1} & I(\psi^k, \psi^l)_{k,l=L}^{n-1} \end{bmatrix} = \begin{bmatrix} A & B \\ -B^T & C \end{bmatrix}. \quad (13)$$

La matrice d'aire M^L est une matrice anti-symétrique de taille $D2^n \times D2^n$ contenant 4 blocs principaux. Le bloc A (jaune en fig. 3) est de taille $D2^L \times D2^L$, le bloc B (rouge) est de taille $D2^L \times (D2^n - D2^L)$, et le bloc C (vert) est de taille $(D2^n - D2^L) \times (D2^n - D2^L)$, voir fig. 3.

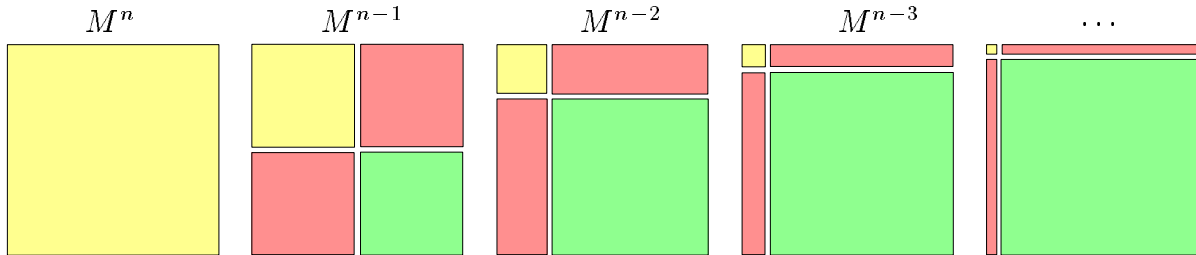


Figure 3: Les matrices d'aire M^L pour $L \in \{0, \dots, n\}$ ont la même taille $D2^n \times D2^n$ mais leur décomposition en quatre blocs est différente selon le niveau de résolution L .

3.2 Calcul efficace des matrices d'aire

Les matrices d'aire doivent être calculées pour tous les niveaux de résolution. Chaque élément de ces matrices est l'intégrale d'un produit de fonctions (9). En posant $N = 2^n$, le calcul de toutes les matrices nécessitera $O(\log(N) \cdot N^2)$ évaluations éventuellement coûteuses de ces intégrales. Nous allons montrer dans ce chapitre que dans le cas de courbes multirésolution, seule l'évaluation des éléments de la matrice "mère" M^n est nécessaire. Par bilinéarité de la forme (10), les matrices des niveaux $n-1, \dots, L$ peuvent ensuite être obtenues en utilisant de manière récursive les équations de décomposition (2).

Dans le cas où un argument de la forme bilinéaire I reste fixe (indice k), on obtient les relations récursives suivantes. Nous les appellerons filtres (P) et (Q) en fonction du filtre de décomposition utilisé:

$$\text{(P) - filter:} \quad I(\varphi^{L-1}, \psi^k) = (P^L)^T I(\varphi^L, \psi^k) \quad (14)$$

$$\text{(Q) - filter:} \quad I(\psi^{L-1}, \psi^k) = (Q^L)^T I(\varphi^L, \psi^k). \quad (15)$$

Par symétrie on obtient des formules analogues dans le cas où le premier argument est fixe. Dans le cas où les deux arguments ont besoin d'être calculés de manière récursive, les équations de décomposition seront utilisées sous forme produit tensoriel:

$$\text{(PP) - filter:} \quad I(\varphi^{L-1}, \varphi^{L-1}) = (P^L)^T I(\varphi^L, \varphi^L) (P^L) \quad (16)$$

$$\text{(PQ) - filter:} \quad I(\varphi^{L-1}, \psi^{L-1}) = (P^L)^T I(\varphi^L, \varphi^L) (Q^L) \quad (17)$$

$$\text{(QQ) - filter:} \quad I(\psi^{L-1}, \psi^{L-1}) = (Q^L)^T I(\varphi^L, \varphi^L) (Q^L) \quad (18)$$

L'algorithme qui découle de ces formules, pour calculer successivement les matrices $M^{n-1}, M^{n-2}, \dots, M^L$, est schématisé en fig. 4.

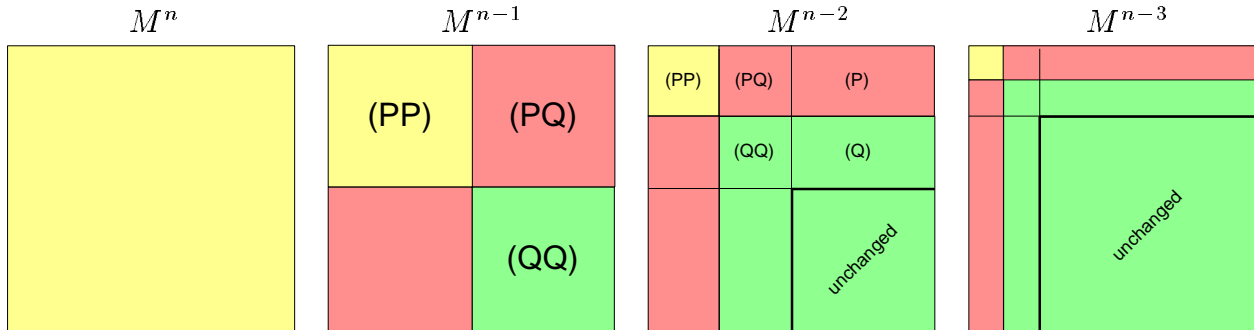


Figure 4: Calcul récursif de la matrice d'aire: M^{n-1} à partir de M^n , M^{n-2} à partir de M^{n-1} etc.

4. DEFORMATION MULTIREOLUTION SOUS CONTRAINTE D'AIRES

Des contraintes linéaires peuvent facilement être incorporées dans un système de design interactif de courbes et surfaces NURBS, e.g. positions, tangentes, orthogonalité. Des contraintes non-linéaires comme l'aire, le volume, la convexité ou des contraintes d'origine physique, sont difficiles à traiter, en particulier à cause d'un effort numérique considérable.

Dans ce chapitre, nous présentons une méthode de déformation de courbes multirésolution planes et périodiques à n'importe quel niveau de résolution tout en préservant l'aire enfermée. De manière élégante il va être possible de :

- préserver les détails en modifiant globalement la forme de la courbe, et
- déformer la courbe à différents niveaux de résolution

tout en conservant l'aire enfermée par la courbe.

4.1 Principe

La méthode de déformation est composée de trois étapes principales :

- **décomposition** : La courbe est décomposée jusqu'à une échelle qui a été définie l'utilisateur. La courbe est alors représentée par un polygone de contrôle et un ensemble de coefficients de détail (voir formule (6)).
- **déformation** : L'utilisateur définit une déformation en déplaçant un ou plusieurs points du polygone de contrôle.
- **conservation d'aire** : Une nouvelle courbe est calculée en respectant la déformation définie par l'utilisateur et qui a la même aire intérieure que la courbe initiale. Cette étape constitue le problème que nous nous proposons de résoudre.

Etant donné une courbe multirésolution $\mathbf{c}(t)$ et un niveau de décomposition L auquel est appliquée la déformation, prenons les notations suivantes, en négligeant les indices L et n :

- \mathcal{A}_{ref} est l'aire de référence que nous voulons conserver
- $\begin{pmatrix} X_0 \\ Y_0 \end{pmatrix}$ sont les coordonnées après déformation du polygone de contrôle, et avant les corrections pour conserver l'aire
- $\mathcal{A}_0 = \frac{1}{2} X_0 M Y_0^T$ est l'aire correspondante qui doit être corrigée
- $\begin{pmatrix} X \\ Y \end{pmatrix}$ sont les coordonnées après correction
- $\mathcal{A} = \frac{1}{2} X M Y^T$ est la nouvelle aire

Dans ces termes, nous pouvons reformuler les trois étapes :

- exprimer $\mathbf{c}(t)$ dans une base multirésolution V^L , voir formule (6)

- déformer le polygone de contrôle pour obtenir X_0 et Y_0
- calculer X et Y "proches de" X_0 de Y_0 tels que $\mathcal{A} = \mathcal{A}_{ref}$

4.2 Critère de lissage

Pour obtenir une courbe "lisse", i.e. qui a une forme visuelle plaisante, nous utilisons la minimisation d'une version linéarisée de l'énergie de tension, classique en design variationnel :

$$\mathcal{E} = \int |c''(t)|^2 dt = \int x''(t)^2 + y''(t)^2 dt$$

Dans le cadre particulier d'une courbe multirésolution, nous pouvons l'exprimer de façon similaire à la formule (11) pour un niveau $L \in \{0, \dots, n\}$:

$$\mathcal{E}(X, Y) = \frac{1}{2} X^L H^L (X^L)^T + \frac{1}{2} Y^L H^L (Y^L)^T \quad (19)$$

où H^L est une matrice symétrique calculée par le même processus que la matrice M^L .

4.3 Conservation de l'aire

Afin d'assurer que X et Y seront proches de X_0 de Y_0 , on introduit un critère de distance, et notre problème se reformule ainsi en un problème d'optimisation convexe sous contrainte :

$$\begin{aligned} \min_{X, Y} \quad & \frac{1}{2} X H X^T + \frac{1}{2} Y H Y^T + \alpha (\|X - X_0\|^2 + \|Y - Y_0\|^2) \\ \text{sous} \quad & \mathcal{A} - \mathcal{A}_{ref} = 0. \end{aligned} \quad (20)$$

$\alpha \geq 0$ est un scalaire qui permet de choisir l'équilibre entre une courbe plus lisse ou une courbe qui respecte mieux la déformation définie par l'utilisateur.

Pour résoudre ce système, nous utilisons une linéarisation de la contrainte d'aire et les multiplicateurs de Lagrange, ce qui nous ramène à trouver le point stationnaire d'une fonction quadratique, c'est à dire à résoudre un système linéaire.

4.4 Exemples

Dans toutes les illustrations nous utiliserons un code de couleurs qui nous permettra d'identifier la courbe initiale (en bleu), la courbe déformée (en vert), et la courbe finale vérifiant la contrainte d'aire (en rouge). Les polygones de contrôle présents sur la plupart des figures sont les coefficients d'échelle au niveau L , donc les polygones que peut manier

l'utilisateur.

La figure 5 illustre les 3 étapes de notre algorithme, et montre le rôle du coefficient α dans la formule (20). La courbe originale a 32 points de contrôle ($n = 5$). Après deux décompositions on obtient 8 coefficients d'échelle qui forment le polygone de contrôle au niveau $L = 3$ et 24 coefficients de détail. La déformation est définie en déplaçant un point de contrôle, ce qui, grâce la représentation multirésolution, déforme la courbe sur une large portion. La reconstruction à ce moment donne une courbe qui ne vérifie pas la contrainte d'aire. La courbe finale est calculée par la formule (20) dans son expression multirésolution puis reconstruite. Nous constatons que le choix d'une petite valeur pour α favorise une courbe régulière mais qui s'éloigne de la courbe déformée (en vert), alors qu'une valeur plus grande donne une courbe plus conforme aux souhaits de l'utilisateur.

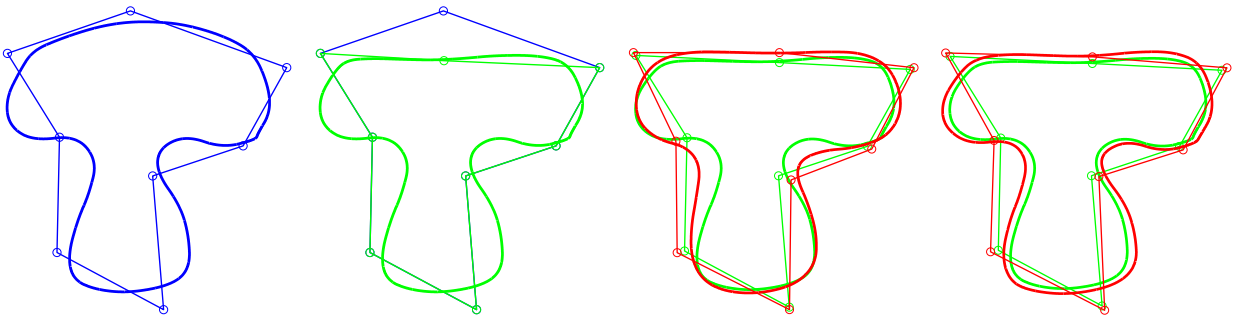


Figure 5: Influence du coefficient α . (a) courbe initiale et son polygone de contrôle (bleu), $n=5$, $L=3$. (b) polygone de contrôle déformé et courbe reconstruite sans conservation d'aire (vert), $L=3$. (c) polygone de contrôle corrigé et courbe vérifiant la contrainte d'aire (rouge), $\alpha = 1$. (d) $\alpha = 100$.

Remarquons que cette méthode est une optimisation globale qui peut d'une part s'avérer coûteuse malgré un système creux, et qui d'autre part ne permet pas un contrôle précis du résultat : on voudrait pouvoir déformer la courbe localement, et utiliser pleinement les fonctionnalités de l'analyse multirésolution en fixant les coefficients de détails durant l'optimisation, afin de modifier la forme globale sans changer le caractère de la courbe.

Ce double but est atteint facilement en effectuant la minimisation (20) seulement sur certains coefficients (d'échelle et/ou de détails). La figure 6 illustre une déformation pour laquelle l'aire a été compensée seulement sur les points voisins, pour différentes étendues. On peut ainsi borner la déformation sur un voisinage facile à contrôler.

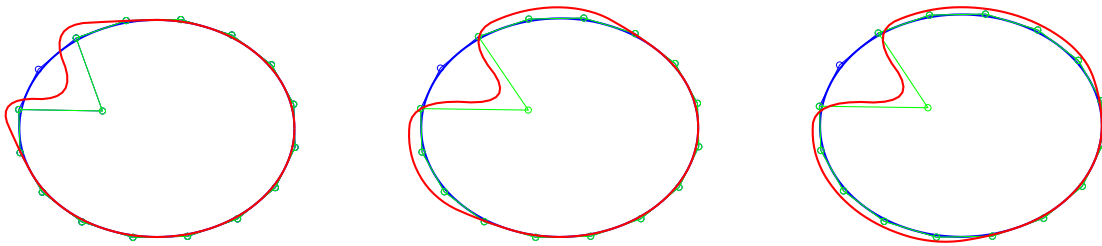


Figure 6: *Contrôle local. $n = 7$ and $L = 4$. De gauche à droite: 1, 2 et 5 voisins.*

Le cas le plus intéressant est celui de la figure 7: l'aire est adaptée en ne corrigeant que les deux voisins directs sur le polygone de contrôle, et sans toucher aux détails. On remarque que la forme globale est modifiée pour l'adaptation d'aire, mais que tous les détails qui caractérisent la courbe sont soigneusement conservés. En d'autres termes, l'expression du problème (20) dans la base multirésolution nous permet de corriger la surface de la courbe avec seulement quelques coefficients d'échelle, ce qui a le double avantage de permettre un contrôle du résultat et d'alléger considérablement les calculs.

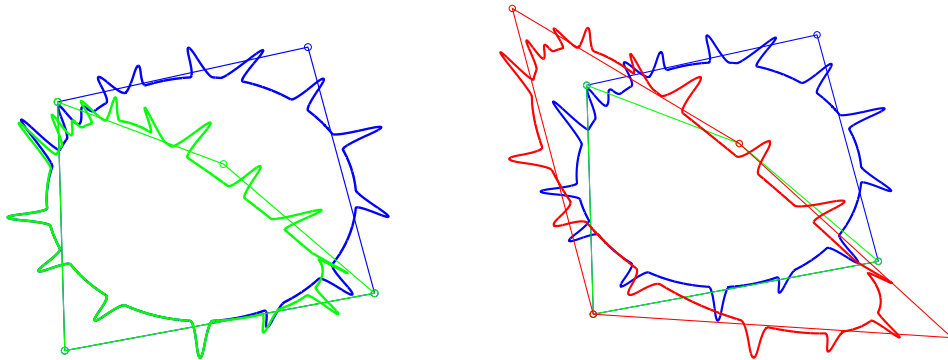


Figure 7: *déformation multirésolution. $n = 7$, $L = 2$. courbe initiale (bleu), courbe déformée au niveau 2 (vert), courbe finale conservant l'aire (rouge).*

5 CONCLUSION ET TRAVAUX FUTURS

Nous avons présenté une méthode de déformation de courbes planes et périodiques. Cette méthode intègre la contrainte non linéaire d'aire grâce à une approximation. De plus, nous avons développé une expression multirésolution de la contrainte d'aire pour des courbes multirésolution basées sur des schémas d'ondelettes. Cette expression se calcule efficacement par une relation de récurrence qui assure un pré-calcul rapide. La méthode d'optimisation employée permet enfin des déformations en temps-réel.

Les travaux actuels concernent l'intégration d'autres contraintes non linéaires dans cet environnement d'édition multirésolution, telle que la conservation de la longueur. Une généralisation aux surfaces est aussi en préparation.

REFERENCES

- [1] Beylkin, G., Coifman R., Rokhlin V., *Fast wavelet transforms and numerical algorithms I*, Communications on Pure and Applied Mathematics XLV (1999), 4-S3.
- [2] Chaikin G., *An algorithm for high speed curve generation*. Computer Graphics and Image Processing **3** (1974), 346-349.
- [3] Cohen A., Matei B., *Compact representations of images by edge adapted multiscale transforms*, IEEE ICIP conference, Tessaloniki, (2001).
- [4] Dahmen W., Micchelli C., *Using the refinement equation for evaluating integrals of wavelets*, SIAM J. Numerical Analysis, **30**, (1993), 507–537.
- [5] Eberly D., Lancaster J., *On gray scale image measurements*, CVGIP: Graphical Models and Image Processing, **53** (6), (1991), 538–549.
- [6] Elber G., *Linearizing the area and volume constraints*, Technical Report CIS-2000-04, (2000).
- [7] Elber G., *Multiresolution curve editing with linear constraints*, 6th ACM/IEEE sump. on Solid Modeling and Applications, Ann Arbor, (2001), 109–119.
- [8] Farin G., *Curves and Surfaces for Computer Aided Geometric Design*, Academic Press, New York, 4th edition, (1997).
- [9] Finkelstein A., Salesin D.H., *Multiresolution curves*, Computer Graphics Proceedings (SIGGRAPH 94), (1994), 261–268.
- [10] Jacob M., Blu T., Unser M., *An exact method for computing the area moments of wavelet and spline curves*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **23** (6), (2001), 633–642.
- [11] Lounsbery M., De Rose T., Warren J., *Multiresolution analysis for surfaces of arbitrary topological type*, ACM Transaction on Graphocs, **16** (1), (1997), 34–73.
- [12] Mallat S., *A theory for multiresolution signal decomposition: the wavelet representation*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **11**, (1989), 674–693.
- [13] Mertins A., *Signal Analysis: Wavelets, Filter Banks, Time-Frequency Transforms and Applications*, John Wiley & Sons., Chichester, ISBN: 0-471-98626-7, (1999).
- [14] Sapiro G., Tannenbaum A., *Area and length preserving geometric invariant scale-spaces*, IEEE Transactions on Pattern Analysis and Machine Intelligence **17** (1), (1995), 67–72.
- [15] Schröder P., Zorin D., *Subdivision for Modeling and Animation*, SIGGRAPH course notes, (1998, 1999).
- [16] Stollnitz E., DeRose T., and Salesin D., *Wavelets for Computer Graphics: Theory and Applications*, Morgan-Kaufmann, (1996).