

**SIMULATION D'UN FLUIDE À LA SURFACE D'UN OBJET PAR
LA MÉTHODE « SMOOTHED PARTICLE HYDRODYNAMICS »**

par

Gilles-Philippe Paillé

Mémoire présenté au Département d'informatique
en vue de l'obtention du grade de maître ès sciences (M.Sc.)

FACULTÉ DES SCIENCES

UNIVERSITÉ DE SHERBROOKE

Sherbrooke, Québec, Canada, 17 décembre 2009



Library and Archives
Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*
ISBN: 978-0-494-61447-1
Our file *Notre référence*
ISBN: 978-0-494-61447-1

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

Le 17 décembre 2009

*le jury a accepté le mémoire de Monsieur Gilles-Philippe Paillé
dans sa version finale.*

Membres du jury

Professeur Richard Egli
Directeur de recherche
Département d'informatique

Professeur Djemel Ziou
Membre
Département d'informatique

Professeure Virginie Charette
Président rapporteur
Département de mathématiques

Sommaire

Dans ce mémoire, je présente une méthode par particules pour la simulation de l'écoulement d'un fluide contraint à se déplacer à la surface d'un objet. Plus précisément, la méthode *Smoothed Particle Hydrodynamics* sera utilisée pour discrétiser le fluide ainsi que pour l'animer. Un maillage approximant l'objet sur lequel le fluide s'écoule sera utilisé pour le supporter et diriger son déplacement. Contrairement aux méthodes précédentes, une méthode par particules permet l'indépendance du temps de calcul et la complexité du maillage supportant le fluide. Elle permet également de suivre l'évolution de l'interface liquide/gaz et liquide/liquide sans procédure supplémentaire. Des temps de calcul de l'ordre du temps réel peuvent donc être obtenus sur des objets très complexes.

Remerciements

Je tiens à remercier tous les membres de ma famille pour leur support tout au long de mes études et qui m'ont encouragé à entreprendre une maîtrise. J'apprécie vraiment qu'ils soient venus me rendre visite lors de mon séjour en France. Je remercie également tous les gens que j'ai connus à Limoges, avec qui j'ai voyagé et passé de très beaux moments.

Je remercie mon directeur de recherche, le professeur Richard Egli, pour ses judicieux conseils, ses idées et son ouverture d'esprit quant à l'exploration de nouvelles pistes ayant mené à la réussite du projet. Je tiens également à souligner l'aide de Benoît Crespin, mon directeur de recherche lors de mon année à l'Université de Limoges, qui a grandement contribué à l'élargissement mon réseau de contacts en Europe et qui m'a accordé sa confiance dans le développement de mon projet de stage à Limoges.

Je remercie tous mes collègues, à Sherbrooke et à Limoges, pour les bons moments passés en leur compagnie. En particulier, je remercie Khalid Djado pour tous les échanges que nous avons eus, ses conseils et son sens de l'humour.

Table des matières

| | |
|---|------------|
| Sommaire | i |
| Remerciements | ii |
| Table des matières | iii |
| Introduction | 1 |
| 1 Notions Préliminaires | 4 |
| 1.1 Smoothed Particle Hydrodynamics | 4 |
| 1.2 Géométrie Riemannienne | 8 |
| 2 Surface SPH | 14 |
| Conclusion | 25 |
| A Force géodésique | 27 |
| A.1 Cas planaire | 27 |

TABLE DES MATIÈRES

A.2 Cas sphérique 29

Introduction

L'écoulement d'un fluide à la surface d'un objet est un phénomène souvent rencontré dans la vie de tous les jours. Un premier exemple est la bulle de savon. La tension de surface étant très élevée, le liquide contenu à l'intérieur de la mince couche de savon est contraint à se déplacer dans un environnement s'apparentant à un espace en deux dimensions. Un deuxième exemple est le déplacement des nuages à la surface de la terre. Malgré l'épaisseur des nuages, il est intéressant de voir que leurs déplacements ne correspondent pas à un écoulement en trois dimensions traditionnel. Ils sont plutôt contraints à se déplacer dans l'atmosphère terrestre. Il en est de même pour tous les autres astres ayant une atmosphère.

Cette contrainte nous permet d'appliquer une simplification importante avant de tenter de simuler ces phénomènes. Étant donné que le mouvement du fluide dans une des dimensions est négligeable, il est possible de réduire le problème en éliminant un degré de liberté. Cette réduction résulte en un ensemble d'équations s'apparentant aux équations d'écoulement d'un fluide en deux dimensions. Cependant, comme nous le verrons plus tard, quelques détails doivent être considérés lorsque l'espace dans lequel le fluide se déplace possède une courbure. Se situant au niveau théorique, cette simplification permet d'économiser du temps de calcul, peu importe la méthode de simulation utilisée.

Malgré sa grande présence, ce type de phénomène est très peu exploité dans le domaine de l'infographie. Pour les méthodes développées jusqu'à présent [15][14][13][5][4], le temps de calcul reste très élevé et dépend fortement de la complexité du maillage représentant la surface, c'est-à-dire le domaine de simulation. Dans ce mémoire, une méthode de simulation par particules est présentée pour résoudre les équations d'écoulement d'un fluide contraint à se déplacer à la surface d'un objet. Cette méthode, nommée *Smoothed*

INTRODUCTION

Particle Hydrodynamics (SPH), permet l'indépendance entre le temps de simulation et la complexité du maillage de la surface. Nous pouvons donc obtenir des temps de calcul de l'ordre du temps réel malgré la complexité de l'objet.

La surface servant à la simulation doit subir un prétraitement afin de simplifier l'application des outils nécessaires à la méthode SPH. En effet, une étape de paramétrisation est nécessaire afin d'adoucir les grandes variations géométriques de la surface. Cette étape transforme la surface complexe en une surface régulière, comme une sphère [8][10] ou un plan [9][7], où il est beaucoup plus facile d'appliquer les opérateurs différentiels tels que définis par la formulation SPH. Les distorsions créées par une telle transformation devront, par contre, être considérés lors des calculs.

Pour cette raison, une paramétrisation conservant localement les angles a été sélectionnée. Très utile, cette propriété nous permet une modification minimale des équations afin qu'elles soient valides dans ce nouveau domaine. Ces transformations, nommées *applications conformes*, sont définies par un étirement infinitésimalement uniforme à chaque point de la surface, ce qui nous garantit qu'un cercle infinitésimal défini sur la surface restera un cercle dans le domaine paramétrique. Cette deuxième propriété nous sera très utile lors de la recherche de voisinage qui s'effectue traditionnellement par la définition d'un voisinage circulaire autour d'un point.

Le problème étant transposé dans ce nouveau domaine, il est maintenant simple de simuler l'écoulement directement dans cet espace. L'application des forces et le déplacement des particules seront ainsi effectués dans un seul et même espace, facilitant l'implémentation et la gestion des variables. La position de chaque particule pourra ensuite être transposée sur la surface à l'aide de l'application inverse, pour la visualisation de l'écoulement.

Les contributions sont l'utilisation d'une méthode par particules pour la simulation de l'écoulement d'un fluide à la surface d'un objet et l'indépendance de la simulation vis-à-vis la complexité du maillage de l'objet sous-jacent. De plus, en tenant compte de la trajectoire géodésique empruntée par une particule se déplaçant dans une espace courbe, la distorsion induite par la paramétrisation de l'objet, présente dans les méthodes par paramétrisation [15][13], est éliminée. Une simulation par particules permet d'autres formes de visualisation qui nécessiteraient des procédures supplémentaires en utilisant une simulation par

INTRODUCTION

maillage. Un de ces types de visualisation est la simulation d'interfaces liquide/liquide ou liquide/gaz, qui s'effectue de façon naturelle avec la méthode SPH.

Le document est organisé de la façon suivante : Le chapitre 1 introduit les notions importantes à connaître avant d'entreprendre la lecture du travail présenté. Il sera notamment question de la formulation originale de la méthode SPH et de la description mathématique des concepts de base de la géométrie riemannienne. Le chapitre 2 présente l'article **Smoothed Particle Hydrodynamics on Triangle Mesh** qui a été soumis à la conférence *Eurographics* où les articles acceptés sont publiés dans le journal *Computer Graphics Forum*. Quelques dérivations techniques se trouvent en annexes pour ne pas encombrer inutilement le texte, les références étant faites directement dans le texte aux endroits appropriés.

Chapitre 1

Notions Préliminaires

Ce chapitre traitera des notions nécessaires à la lecture de l'article au chapitre 2. La section 1.1 est une brève introduction à la méthode *Smoothed Particle Hydrodynamics* dans l'espace euclidien alors que la section 1.2 est une présentation des concepts clés de la géométrie riemannienne.

1.1 Smoothed Particle Hydrodynamics

La méthode SPH est tout d'abord une méthode qui, à partir d'un ensemble discret de points, permet d'évaluer les valeurs d'une fonction à chacun de ces points afin d'obtenir une approximation de la fonction en tous points de l'espace. Cette approche nous permet alors d'utiliser le calcul différentiel sur la fonction nouvellement définie. Elle a été indépendamment introduite par Gingold & Monaghan [6] et Lucy [12] en 1977.

Avant de considérer un ensemble discret, prenons une fonction f continue de \mathbb{R}^n . L'égalité 1.1 démontre que nous pouvons obtenir la valeur en un point en utilisant les valeurs au voisinage de ce point [6].

1.1. SMOOTHED PARTICLE HYDRODYNAMICS

$$f(\vec{x}) = \lim_{h \rightarrow 0} \int_R f(\vec{x}') W(\vec{x} - \vec{x}', h) dR \quad (1.1)$$

où le noyau W satisfait aux conditions suivantes :

$$\begin{aligned} \int_R W(\vec{x} - \vec{x}', h) dR &= 1 \\ \lim_{h \rightarrow 0} W(\vec{x} - \vec{x}', h) &= \delta(\vec{x} - \vec{x}') \end{aligned}$$

où

$$\delta(\vec{x} - \vec{x}') = \begin{cases} 1 & \text{si } \vec{x} - \vec{x}' = \vec{0} \\ 0 & \text{sinon} \end{cases}$$

En pratique, on doit ajouter quelques conditions au noyau étant donné qu'il sera nécessaire que les calculs de dérivées se comportent de façon satisfaisante [11] :

$$\begin{aligned} \int_R (\vec{x} - \vec{x}') W(\vec{x} - \vec{x}', h) dR &= 0 && \text{Symétrie} \\ W(\vec{x} - \vec{x}', h) |_{\|\vec{x} - \vec{x}'\| < h} &> 0 && \text{Positivité} \\ W(\vec{x} - \vec{x}', h) |_{\|\vec{x} - \vec{x}'\| \geq h} &= 0 && \text{Support compact} \end{aligned}$$

où la condition de support compact est seulement introduite pour réduire l'ordre de complexité de l'algorithme. Tel qu'illustré dans la figure 1.1, un exemple de noyau satisfaisant à ces conditions est le suivant :

$$W(\vec{r}, h) = \frac{4}{\pi h^8} \begin{cases} (h^2 - \|\vec{r}\|^2)^3, & \text{si } \|\vec{r}\| < h \\ 0, & \text{sinon} \end{cases}$$

1.1. SMOOTHED PARTICLE HYDRODYNAMICS

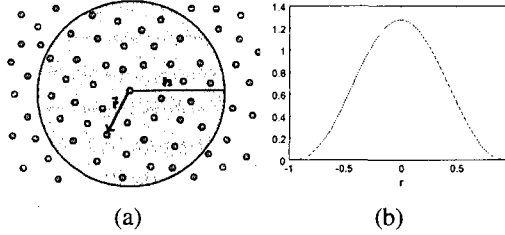


figure 1.1 – (a) Voisinage circulaire d'une particule. (b) Exemple d'un noyau ($h = 1$)

D'une façon similaire, nous pouvons estimer la valeur du gradient et du laplacien par les égalités suivantes.

$$\begin{aligned}\nabla f(\vec{x}) &= \lim_{h \rightarrow 0} \int_R \nabla f(\vec{x}') W(\vec{x} - \vec{x}', h) dR \\ \Delta f(\vec{x}) &= \lim_{h \rightarrow 0} \int_R \Delta f(\vec{x}') W(\vec{x} - \vec{x}', h) dR\end{aligned}$$

Mais ceci nécessite de connaître à l'avance ∇f et Δf . L'idéal serait de pouvoir obtenir le même résultat en ne connaissant que f . Étant donné que nous avons imposé la condition du support compact, il est possible d'utiliser l'intégration par parties pour obtenir les égalités suivantes :

$$\nabla f(\vec{x}) = \lim_{h \rightarrow 0} \int_R f(\vec{x}') \nabla W(\vec{x} - \vec{x}', h) dR \quad (1.2)$$

$$\Delta f(\vec{x}) = \lim_{h \rightarrow 0} \int_R f(\vec{x}') \Delta W(\vec{x} - \vec{x}', h) dR \quad (1.3)$$

Nous avons maintenant tous les ingrédients nécessaires pour définir les équivalents discrets de ces formules. Nous devons tout d'abord laisser tomber la limite étant donné que nous devons avoir un support compact d'aire finie positive. Cette première modification fera des équations 1.1, 1.2 et 1.3 des approximations. La deuxième modification sera de ne considérer qu'une quantité finie de points à l'intérieur du support. Il faudra donc considérer

1.1. SMOOTHED PARTICLE HYDRODYNAMICS

que chacun des points p_j sont sélectionnés au centre d'un volume V_j , pour en arriver aux formules suivantes :

$$\begin{aligned}\langle f(\vec{x}) \rangle &= \sum_n f_j W(\vec{x} - \vec{x}_j, h) V_j \\ \nabla \langle f(\vec{x}) \rangle &= \sum_n f_j \nabla W(\vec{x} - \vec{x}_j, h) V_j \\ \Delta \langle f(\vec{x}) \rangle &= \sum_n f_j \Delta W(\vec{x} - \vec{x}_j, h) V_j\end{aligned}\tag{1.4}$$

où $\langle f(\vec{x}) \rangle$ signifie l'approximation de la fonction f .

Le seul détail restant à expliquer est la manière de calculer le volume d'une particule. Nous pouvons exploiter le fait que la méthode sera appliquée à un phénomène physique pour faire le parallèle avec la masse et la masse volumique d'une particule. En effet, pour un objet de masse m et de masse volumique ρ uniforme, nous avons que

$$V = \frac{m}{\rho}\tag{1.5}$$

En assignant une masse m_j à chaque point p_j de notre ensemble discret, il est possible de calculer son volume par l'équivalent discret de la formule 1.5 :

$$V_j = \frac{m_j}{\rho_j}$$

Par cette démarche, nous avons simplement repoussé le problème car il nous reste à connaître la masse volumique ρ_j de chaque particule p_j . Heureusement, une astuce nous permet de l'approximer par la formule 1.4 :

1.2. GÉOMÉTRIE RIEMANNIENNE

$$\begin{aligned}\langle \rho(\vec{x}) \rangle &= \sum_n \rho_j W(\vec{x} - \vec{x}_j, h) V_j \\ &= \sum_n \rho_j W(\vec{x} - \vec{x}_j, h) \frac{m_j}{\rho_j} \\ &= \sum_n m_j W(\vec{x} - \vec{x}_j, h)\end{aligned}$$

À ce stade, nous sommes en mesure d'utiliser la méthode SPH pour des applications physiques. Nous n'avons qu'à assigner une certaine masse à chaque point, calculer la masse volumique de chaque point pour ensuite effectuer des calculs sur un ensemble de valeurs quelconques définies à chaque point.

1.2 Géométrie Riemannienne

Informellement, la géométrie riemannienne est l'étude des espaces munis d'une notion de distance et des fonctions définies sur ceux-ci. Nous pouvons déjà séparer ce sujet en deux sous-domaines, le premier étant l'étude de la structure de ces espaces et le deuxième étant la révision du calcul différentiel des objets définis dans ceux-ci. Cette section est une simple introduction intuitive au sujet, de telle sorte que le lecteur puisse avoir une bonne idée des concepts qui se cachent derrière les équations énoncées dans l'article. Pour une introduction plus détaillée au domaine, le livre de do Carmo [2] est conseillé.

Le principal objet de la géométrie riemannienne est l'espace muni de sa notion de distance, qui est appelé *variété riemannienne*. Un exemple simple d'une variété riemannienne est la sphère dans \mathbb{R}^3 , sa notion de distance étant définie par la longueur d'arc euclidienne d'une courbe tracée sur la sphère. Pour alléger la présentation et mettre en évidence l'application de la théorie au problème posé plus haut, cette section traitera seulement des surfaces dans \mathbb{R}^3 , c'est-à-dire les variétés de dimension 2 plongées dans \mathbb{R}^3 .

D'un point de vue topologique, les surfaces ne sont pas toutes équivalentes. La figure 1.2 démontre trois surfaces de *genres* différents. Informellement, le genre d'une surface

1.2. GÉOMÉTRIE RIEMANNIENNE

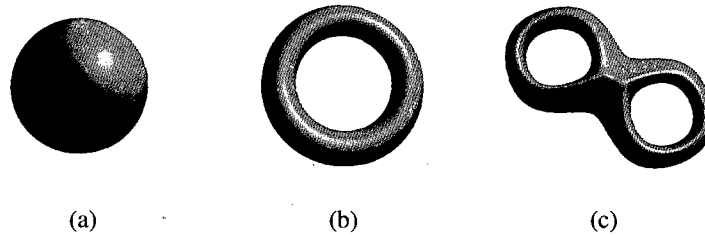


figure 1.2 – Trois surfaces topologiquement différentes. Surfaces de genre 0 (a), 1 (b) et 2 (c).

S est le nombre de poignées qu'il faut ajouter à une sphère pour obtenir une surface de topologie équivalente à S .

Pour étudier une surface S , la méthode classique est d'utiliser des cartes locales. Une *carte locale* u est une simple assignation de coordonnées de \mathbb{R}^2 à chaque point d'un ouvert U de S . Ce processus d'assignation de coordonnées est appelée une *paramétrisation*. Nous pouvons ainsi recouvrir S avec un ensemble de cartes locales pour obtenir un *atlas*. Une carte u et son inverse peuvent être écrits sous la forme mathématique suivante :

$$u^1 = u^1(x^1, x^2, x^3)$$

$$u^2 = u^2(x^1, x^2, x^3)$$

$$x^1 = x^1(u^1, u^2)$$

$$x^2 = x^2(u^1, u^2)$$

$$x^3 = x^3(u^1, u^2)$$

où les exposants représentent des indices.

Un vecteur \vec{v} tangent à la surface en un point p fait partie de l'espace tangent en ce point, c'est-à-dire $\vec{v} \in T_p$. Cet espace vectoriel peut être généré par deux vecteurs linéairement indépendants $\vec{g}_1, \vec{g}_2 \in T_p$. Ayant défini une carte locale u dans un voisinage local de p , une façon simple est de choisir \vec{g}_1 et \vec{g}_2 comme étant les vecteurs tangents aux lignes de

1.2. GÉOMÉTRIE RIEMANNIENNE

coordonnées.

$$\begin{aligned}\vec{g}_1 &= \left(\frac{\partial x^1}{\partial u^1}, \frac{\partial x^2}{\partial u^1}, \frac{\partial x^3}{\partial u^1} \right) \\ \vec{g}_2 &= \left(\frac{\partial x^1}{\partial u^2}, \frac{\partial x^2}{\partial u^2}, \frac{\partial x^3}{\partial u^2} \right)\end{aligned}$$

Il est alors possible de trouver deux nombres réels v^1 et v^2 tels que $\vec{v} = v^1 \vec{g}_1 + v^2 \vec{g}_2$. Nous pouvons maintenant croire qu'il serait possible de travailler seulement avec les composantes (v^1, v^2) sans jamais faire référence au vecteur original \vec{v} qui est défini dans \mathbb{R}^3 . Premièrement, voyons comment calculer le produit scalaire de deux vecteurs tangents \vec{v} et \vec{w} :

$$\begin{aligned}\vec{v} \cdot \vec{w} &= (v^1 \vec{g}_1 + v^2 \vec{g}_2) \cdot (w^1 \vec{g}_1 + w^2 \vec{g}_2) \\ &= v^1 w^1 (\vec{g}_1 \cdot \vec{g}_1) + v^1 w^2 (\vec{g}_1 \cdot \vec{g}_2) + v^2 w^1 (\vec{g}_2 \cdot \vec{g}_1) + v^2 w^2 (\vec{g}_2 \cdot \vec{g}_2) \\ &= \begin{bmatrix} v^1 & v^2 \end{bmatrix} \begin{bmatrix} \vec{g}_1 \cdot \vec{g}_1 & \vec{g}_1 \cdot \vec{g}_2 \\ \vec{g}_2 \cdot \vec{g}_1 & \vec{g}_2 \cdot \vec{g}_2 \end{bmatrix} \begin{bmatrix} w^1 \\ w^2 \end{bmatrix} \\ &= \begin{bmatrix} v^1 & v^2 \end{bmatrix} G \begin{bmatrix} w^1 \\ w^2 \end{bmatrix}\end{aligned}$$

où $(G)_{ij} = g_{ij} = \vec{g}_i \cdot \vec{g}_j$ est appelé le tenseur métrique. On note les composantes de l'inverse de G par $(G^{-1})^{ij} = g^{ij}$.

À ce stade, il devient utile d'utiliser la convention de sommation d'Einstein. Il s'agit simplement d'alléger la notation en rendant implicite la sommation lorsqu'un indice se répète.

1.2. GÉOMÉTRIE RIEMANNIENNE

$$\begin{aligned}v^i \vec{g}_i &= \sum_i v^i \vec{g}_i \\ &= v^1 \vec{g}_1 + v^2 \vec{g}_2\end{aligned}$$

Pour le moment, nous avons trouvé une façon de définir un vecteur tangent par rapport à une base donnée, de telle sorte qu'on puisse calculer le produit scalaire en ne connaissant que les coordonnées par rapport à cette base. Ce calcul est effectué à l'aide du tenseur métrique.

Considérons maintenant un champ de vecteurs \vec{v} tangents défini dans un voisinage de p . Il est possible de définir une dérivée directionnelle d'un champ de vecteurs, nommée *dérivée covariante*, telle que le vecteur résultant soit toujours dans le plan tangent de p . Nous devons considérer deux phénomènes. Le premier est que le vecteur normal à la surface peut varier, ce qui entraînerait un vecteur qui n'est pas nécessairement tangent. Le deuxième est que la base vectorielle peut aussi changer de point en point, ce qui entraînerait une variation des composantes du vecteur sans que le vecteur n'ait de variations réelles. La dérivée usuelle dans la direction d'une ligne de coordonnée est définie de la façon suivante, sans considérer le premier phénomène.

$$\begin{aligned}\frac{\partial \vec{v}}{\partial u^j} &= \frac{\partial (v^i \vec{g}_i)}{\partial u^j} \\ &= \frac{\partial v^i}{\partial u^j} \vec{g}_i + v^i \frac{\partial \vec{g}_i}{\partial u^j}\end{aligned}\tag{1.6}$$

Nous pouvons maintenant définir la dérivée covariante comme étant la projection de la dérivée directionnelle usuelle dans le plan tangent T_p . Le premier terme de l'équation 1.6 étant déjà un vecteur tangent, nous pouvons le laisser intact. Cependant, le deuxième terme peut produire une composante normale. On pose donc Γ_{ij}^k , nommés les symboles de Christoffel, comme étant les composantes telles que la projection du deuxième terme s'écrive en fonction des vecteurs de la base vectorielle de T_p .

1.2. GÉOMÉTRIE RIEMANNIENNE

$$P_{T_p} \left\{ \frac{\partial \vec{g}_i}{\partial w^j} \right\} = \Gamma_{ij}^k \vec{g}_k$$

où P_{T_p} est la projection sur T_p .

Nous avons donc une formulation mathématique complète de la dérivée covariante dans la direction d'une ligne de coordonnée, notée $\nabla_{\vec{g}_j}$, par l'équation suivante :

$$\begin{aligned} \nabla_{\vec{g}_j} \vec{v} &= P_{T_p} \left\{ \frac{\partial \vec{v}}{\partial w^j} \right\} \\ &= \frac{\partial v^i}{\partial w^j} \vec{g}_i + v^i P_{T_p} \left\{ \frac{\partial \vec{g}_i}{\partial w^j} \right\} \\ &= \frac{\partial v^i}{\partial w^j} \vec{g}_i + v^i \Gamma_{ij}^k \vec{g}_k \\ &= \left(\frac{\partial v^k}{\partial w^j} + v^i \Gamma_{ij}^k \right) \vec{g}_k \end{aligned}$$

Par linéarité, nous obtenons la dérivée covariante dans une direction quelconque \vec{w} de la façon suivante :

$$\begin{aligned} \nabla_{\vec{w}} \vec{v} &= \nabla_{w^j \vec{g}_j} \vec{v} \\ &= w^j \nabla_{\vec{g}_j} \vec{v} \\ &= \left(w^j \frac{\partial v^k}{\partial w^j} + w^j v^i \Gamma_{ij}^k \right) \vec{g}_k \end{aligned}$$

Les symboles de Christoffel encodent les variations de la base vectorielle et la projection sur le plan tangent. Étant donné qu'on se situe au niveau infinitésimal, la projection peut être vue comme une rotation définie par la variation du vecteur normal dans la direction de \vec{w} . Bien que la définition à l'aide de la rotation soit plus naturelle pour la notion du

1.2. GÉOMÉTRIE RIEMANNIENNE

transport parallèle d'un vecteur, elle porte à croire qu'il est nécessaire d'avoir la connaissance de données extrinsèques à la surface, les vecteurs normaux, alors que la dérivée covariante peut s'effectuer en ne connaissant que le tenseur métrique, qui est une donnée intrinsèque. La raison pour laquelle il est idéal d'avoir une définition intrinsèque de nos objets et de nos opérations est de pouvoir effectuer tous nos calculs sur une carte locale sans jamais faire référence à la surface originale.

L'obtention des symboles de Christoffel étant plutôt technique et n'ajoutant pas à la compréhension des concepts introduits dans cette section, ils seront simplement donnés sans preuve par l'équation suivante :

$$\Gamma_{jk}^i = \frac{g^{im}}{2} \left(\frac{\partial g_{mj}}{\partial u^k} + \frac{\partial g_{mk}}{\partial u^j} - \frac{\partial g_{jk}}{\partial u^m} \right)$$

Le dernier concept à voir avant de présenter l'article est le *transport parallèle*. Il s'agit d'une extension de la notion de parallélisme de l'espace euclidien aux surfaces. Pour le définir mathématiquement, il nous faut tout d'abord sélectionner une courbe γ quelconque sur notre surface. Pour chaque point de la courbe, sélectionnons un vecteur \vec{v} tangent à la surface en ce point. On dit que le champ de vecteur ainsi défini est parallèle s'il satisfait à l'équation suivante :

$$\nabla_{\dot{\gamma}(t)} \vec{v} = 0$$

On peut voir qu'il s'agit simplement de la généralisation du fait qu'un champ de vecteurs parallèles dans l'espace euclidien ne varie pas de point en point. En particulier, si la courbe satisfait à la condition $\nabla_{\dot{\gamma}(t)} \dot{\gamma}(t) = 0$, on dit qu'il s'agit d'une *géodésique*. Elle peut être décrite comme étant une courbe ayant une accélération nulle dans le plan tangent, généralisant la ligne droite pour les surfaces.

Chapitre 2

Méthode particulière pour la simulation de l'écoulement d'un fluide sur un maillage triangulaire

Cette partie du mémoire présente l'article **Smoothed Particle Hydrodynamics on Triangle Meshes**. Il s'agit d'un problème proposé par le professeur Richard Egli dans le but de généraliser la méthode particulière *Smoothed Particle Hydrodynamics* à une surface de genre 0 ou 1 représentée par un maillage triangulaire. Sous la supervision de Richard Egli, j'ai eu à développer la méthode au niveau théorique, à l'implémenter et à rédiger l'article qui constitue ce chapitre.

Nous proposons une méthode permettant la simulation d'un fluide à la surface d'un objet en temps réel. Le principe est de transformer la surface en un espace plus régulier, comme la sphère ou le plan, et de résoudre les équations de Navier-Stokes sur ces nouveaux domaines en considérant les déformations induites par la transformation. L'espace régulier choisi dépendra de la topologie de la surface originale.

L'article est divisé en deux parties. La première est la présentation des méthodes de paramétrisation d'une surface. Une brève description des méthodes existantes permet de

bien définir l'estimation des métriques, des connexions et des vecteurs caractérisant la périodicité du domaine paramétrique. La deuxième partie constitue le coeur de l'article. Une brève revue de la méthode SPH classique est présentée avant de la généraliser aux surfaces de genre 0 ou 1. Une attention particulière est portée au calcul des distances et au calcul de la trajectoire géodésique qu'emprunte une particule se déplaçant dans le domaine paramétrique. La section traitant de la force géodésique présente des formules sans dérivation détaillée. L'annexe A présente donc le calcul détaillé de ces formules pour le cas du plan et de la sphère.

Smoothed Particle Hydrodynamics on Triangle Meshes

Gilles-Philippe Paillé, Richard Egli

Centre MOIVRE, Département d'Informatique
Université de Sherbrooke, Sherbrooke, Québec, CANADA

Abstract

We present a particle-based method for solving Navier-Stokes equations on the surface of objects of genus 0 and 1. When the mesh is conformally mapped to a domain of constant curvature, the problem is greatly simplified due to the regularity of the new domain and the uniform stretching property of conformal maps. We adapt the Smoothed Particle Hydrodynamics method to work in this framework and we reformulate the momentum equation so that it takes into account the geodesic path taken by the particles. A particle-based method allows the simulation time to be independent of the underlying mesh resolution and offers a natural way to compute free-surface flows. We therefore achieve real-time simulation without artifacts caused by the parameterization of the triangle mesh.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

1. Introduction

Fluid flows on the surface of an object can be observed at very different scales. Clouds moving across the surface of the earth can be well approximated by a fluid flowing on a sphere. At a smaller scale, soap films are another example of liquid constrained to move on a curved domain. But the complexity of such phenomena can be simplified by considering that the fluid is moving on a curved 2D domain instead of a small subset of a full 3D Euclidian domain. Such simplifications allow us to remove a degree of freedom, simplifying equations and saving computation time.

Mesh methods for solving Navier-Stokes equations on an object represented by a triangular mesh have been developed, but they suffer from being very expensive to compute and depend heavily on the underlying mesh. We propose a particle approach, namely the Smoothed Particle Hydrodynamics method (SPH), to solve these equations in real time without explicit dependency on the triangle mesh. The resulting method is very simple and requires only minor modification of a standard 2D SPH simulation to be suitable for curved objects of genus 0 and 1.

The simulation uses the uniform stretching property of conformal mapping to simplify equations in the SPH formulation. By first mapping the mesh conformally to a regular

domain, the computations are entirely done in the parameter domain and the particles can be easily transposed to the original object space for visualization.

Our contribution is the use of particle-based real-time simulation for fluid flow over a surface, allowing the simulation to be independent of the complexity of the mesh. By taking into account the geodesic path of a moving particle, we remove distortions resulting from the parameterization of the mesh. A particle-based simulation allows for simple surface tracking, which offers new possibilities for other forms of visualization, like free surface flows, all in a single unified method.

The paper is organized as follows : Section 2 reviews existing work on conformal parameterization, SPH and flow on surfaces. Section 3 explains the offline computations that need to be performed before doing the simulation. Section 4 recalls the basics of the SPH method that works on Euclidean domains. Then, Section 5 extends the SPH method to work on non-Euclidean domains, which is our main contribution. Results are then presented in Section 6 showing the independence of the mesh resolution and free-surface flows on mesh. Our conclusions are given in Section 7.

2. Related Work

2.1. Conformal parameterization

Discrete conformal parameterization has been extensively studied in the past decade. We will concentrate on methods that map genus-0 objects to the unit sphere and genus-1 objects to the plane.

Let us first enumerate sphere parameterization methods. [AHTK99] and [HAT*00] describe a technique in which the object is first mapped to the plane, then stereographically projected to the sphere to obtain the conformal mapping. [GtY02], [Gu03] and [GWC*03] describe a technique in which the Gauss map is taken as the initial approximation, then updated with a diffusion process to minimize an energy functional. The resulting map is then conformal. [LH07] upgrade this method with a new initial map and a modified diffusion process that speeds up convergence.

Global conformal parameterization of genus-1 objects is done by [GtY02], [Gu03] and [GY03] using differential one-forms defined on each edge of the mesh to solve two linear systems. The resulting set of one-forms can then be optimized for some desired properties using the method described in [JWYG04]. Cutting the mesh along the homology basis, it can then be unfolded to a plane homeomorphic to a disk by integrating one-forms.

2.2. Smoothed Particle Hydrodynamics

Smoothed Particle Hydrodynamics was introduced by [GM77] and [Luc77] but the first application of this method in the computer graphics community was made by [DG96] for simulation of deformable bodies. [SAC*99] then used it to simulate lava flows, including heat transfer and non-constant viscosity. [MCG03] used SPH to simulate fluid interactive time, including surface tension force. We have chosen to use this last formulation of SPH due to the simplicity of the formulas and the stability of the overall simulation.

2.3. Surface flows

The concept of fluid flows on surfaces was introduced to the computer graphics community by [Sta03], who computed a local parameterization for each quadrilateral patch on which he solved the Navier-Stokes equations. Transition functions were defined between each pair of neighboring patches to transfer fluid properties across the entire mesh. Distortions are created by a misinterpretation of the advection term, resulting in a fluid that does not follow a geodesic path. [SY04] did the simulation directly on the mesh using an adaptation of the semi-Lagrangian method introduced by [Sta99], eliminating distortion with parallel transport of the velocity vector in the backtracing step. Their method simulates incompressible and inviscid fluid flow by solving the Poisson equation and advecting fluid properties at each iteration.

[LWC05] conformally parameterized the mesh to solve

the Navier-Stokes equations on the parameter domain. The same misinterpretation of the advection term as in [Sta03] deviates the fluid from the geodesic path. [ETK*07] used the simplicial complex and the chain complex paradigm to translate equations into chain operations. [FZKH05] adapted the Lattice-Boltzmann method by using the tangent space at each vertex to project neighboring lattices on it.

In computer animation with particles, [DE09] used a particle system to visualize a velocity field resulting from a fluid simulation. The particles need some velocity correction to avoid interpenetration and two different steps are needed to do the simulation and the visualization. The simulation step is Eulerian and therefore also depends on the resolution of the mesh.

The computational time of these simulations depends heavily on the complexity of the mesh, which makes them difficult to use for real-time applications. Particle simulation removes this dependency. In addition, it allows for new visualization techniques that would require additional processing steps using mesh simulation methods.

2.4. Contribution

Below is a list of the contributions made by the method we present in this paper.

- Particle-based method for surface flows
- Correct treatment of the advection term
- Simulation independent of the mesh resolution

3. Parameterization

Particle-based simulation would be very hard to perform directly on the mesh because of the curvature. The solution that we propose is to transform the mesh to a more regular shape, like a plane or a sphere, via a parameterization process. Many different mapping techniques exist. A subset of these techniques, called conformal maps, uniformly stretch an infinitesimal neighborhood of each point of a surface. This ensures angle preservation and avoids shearing transformation, which are very nice properties to have when working in the parameter domain. Conformal maps are preferred over exponential maps defining local isometries because the latter approach requires too many different maps to cover the mesh efficiently, whereas the former only needs one map.

When expressed in a conformal domain, Navier-Stokes equations only need to be slightly modified to remain valid in the new domain. The uniform stretching also ensures that a small circular neighborhood is kept circular in the new domain, which will be very useful for the neighbor search process of SPH.

We will only use global mapping, since it removes the need to do a mesh segmentation and compute patch transition functions. Genus-0 surfaces are mapped to a sphere

while genus-1 surfaces are mapped to a periodic plane. We will first describe the mapping process, then compute some metric information about these mappings that is needed during the fluid simulation. It is to be noted that everything in this section is done offline.

For what follows, M is a mesh with vertices $\{v_i\}$, edges $\{e_{ij}\}$ and triangles $\{f_{ijk}\}$.

3.1. Genus-0 surfaces

Since a genus-0 surface is homeomorphic to a sphere, a bijective map can be computed between the two surfaces. This is done by the method described in [GWC*03] and [LH07], which we will briefly review here.

A conformal map between two genus-0 surfaces is equivalent to a harmonic map. Since the latter can be computed using a simple iterative minimization procedure, we can use this same procedure to compute a conformal map. The mapping is done by first assigning an initial position $\varphi_0(\vec{v}_i)$ to each vertex of the mesh on a sphere, then iteratively updating the new position to finally get a conformal map. The conformal map is obtained when the mapping minimizes an energy functional of φ on M , described in Equation 1 below.

$$E(M) = \int_M |\nabla\varphi(\vec{x})|^2 dM \quad (1)$$

The energy is minimized when the equation $\Delta\varphi = 0$ is satisfied. The iteration process is conceptually the same as heat diffusion on the surface. This diffusion is done until we reach a steady state of the diffusion equation (Equation 2).

$$\frac{\partial\varphi(v_i)}{\partial t} = -\Delta\varphi(v_i) \quad (2)$$

The initial map can be simply the Gauss map or a stereographic projection of a planar graph representation of the mesh, the latter eliminating fold-over problems. From this initial guess, we iteratively move each vertex in the direction of the tangential component of the Laplacian Δ^T until it converges. The Laplacian on a mesh is defined by Equation 3.

$$\Delta^T\varphi(v_i) = \mathbf{P}_{\varphi(v_i)} \sum_{e_{ij}} k_{ij}(\varphi(v_i) - \varphi(v_j)) \quad (3)$$

where $\mathbf{P}_{\varphi(v_i)}$ is the projection operator on the tangent plane of the sphere at position $\varphi(v_i)$ and k_{ij} is defined by the cotangent formula (Equation 4).

$$k_{ij} = \frac{1}{2} (\cot\alpha_{ij} + \cot\beta_{ij}) \quad (4)$$

where α_{ij} and β_{ij} denote the angles opposite to the edge e_{ij} in the two triangles adjacent to e_{ij} .

At every iteration, we need to normalize $\varphi(v_i)$ to ensure that the whole map stays on the sphere.

3.2. Genus-1 surfaces

For genus-1 surfaces, we need to proceed differently. We will briefly review the technique described by [GY03] for genus-1 surfaces. The mesh M will be cut in a way that allows it to be unfolded to a flat plane, obtaining the fundamental domain. The new position of each vertex on the plane is found by first solving linear systems of one-forms defined on each edge, then integrating the resulting one-forms on the fundamental domain. A one-form ω is an assignment of a real value ω_{ij} to each edge e_{ij} of M such that $\omega_{ij} = -\omega_{ji}$. By identifying the cut edges, we obtain a periodic plane that is topologically the same as the original object.

Obtaining the fundamental domain is done by a collapsing procedure, see [Gib77]. Selecting an arbitrary initial triangle of the mesh and labeling it as visited, all neighbor triangles are recursively visited, with no prescribed order, until every triangle has been labeled. During this procedure, some neighbors will already be visited. In this case, the edge adjacent to the two triangles is marked as a potential fundamental domain border. We only collapse triangles because we need the original edges to build the fundamental domain border. After the collapsing procedure, we obtain a set of potential edges. Dangling edges are recursively removed until none remain. The edges are then cut to create the fundamental domain to be unfolded. The resulting border is not necessarily minimal and does not need to be minimal.

Every genus-1 surface has two generators for a homology basis B_0 and B_1 that can be found from the set of border edges using standard computational homology methods [KMM01]. The basis $\{B_0, B_1\}$ is simply a set of edges representing loops that cannot be contracted to a point on the surface.

The goal is to find a pair of one-forms (σ, τ) so that we can find the new parameters (s_i, t_i) associated with every vertex v_i .

The next step is to solve the two linear systems in Equation 5, the unknowns being the two one-forms ω_{jk}^0 and ω_{jk}^1 defined on each edge e_{jk} of the mesh.

$$\begin{aligned} \omega_{jk}^i + \omega_{kl}^i + \omega_{li}^i &= 0, \forall f_{jkl} \in M \\ \sum_{e_{jk} \in M} k_{jk} \omega_{jk}^i &= 0, \forall v_j \in M \\ \sum_{e_{jk} \in B_l} \omega_{jk}^i &= \delta_l^i, \forall B_l \end{aligned} \quad (5)$$

Choosing $\sigma_{jk} = \omega_{jk}^0$ as being the one-form corresponding to the first parameter domain coordinate, we are ready

to compute the second coordinate one-form τ_{jk} , which is a linear combination of the two known one-forms.

$$\tau_{jk} = \sum_{0 \leq i \leq 1} \gamma_i \omega_{jk}^i$$

The coefficients γ_i are found by solving a small linear system of equations, defined by Equation 6.

$$W\vec{\gamma} = \vec{c} \quad (6)$$

where \vec{c} and W are defined respectively by equations 7 and 8.

$$W_{ij} = \frac{1}{6} \sum_{f_{klm} \in M} \begin{vmatrix} \omega_{kl}^i & \omega_{lm}^i & \omega_{mk}^i \\ \omega_{kl}^j & \omega_{lm}^j & \omega_{mk}^j \\ 1 & 1 & 1 \end{vmatrix} \quad (7)$$

$$c_i = \sum_{f_{klm} \in M} [\omega_{kl}^i \quad \omega_{lm}^i \quad \omega_{mk}^i] M_{klm} \begin{bmatrix} \sigma_{kl} \\ \sigma_{lm} \\ \sigma_{mk} \end{bmatrix} \quad (8)$$

where M_{klm} is defined by Equation 9 [Gu03].

$$M_{klm} = \frac{1}{24A} \begin{pmatrix} 2(b^2 + c^2) & a^2 + b^2 - c^2 & a^2 + c^2 - b^2 \\ b^2 + a^2 - c^2 & 2(c^2 + a^2) & b^2 + c^2 - a^2 \\ c^2 + a^2 - b^2 & c^2 + b^2 - a^2 & 2(a^2 + b^2) \end{pmatrix} \quad (9)$$

where A is the area of the triangle f_{klm} , $a = \|e_{kl}\|$, $b = \|e_{lm}\|$ and $c = \|e_{mk}\|$.

Having obtained a one-form corresponding to each parameter, we are ready to compute the final parameterization by integrating the one-forms over the fundamental domain of M . By choosing an arbitrary initial vertex to be at the origin $(0, 0)$, we find the coordinates of the neighbor vertices by Equation 10, using a flooding procedure similar to the fundamental domain computation.

$$(s_j, t_j) = (s_i, t_i) + (\sigma_{ij}, \tau_{ij}) \quad (10)$$

for every unknown vertex of coordinates (s_j, t_j) neighboring a known vertex of coordinates (s_i, t_i) .

3.2.1. Fundamental parallelogram

The unfolding of the fundamental domain can send two neighboring points very far apart. The periodicity of the fundamental domain is thus an important concept to take into account when computing distances on it, see section 5.2. For

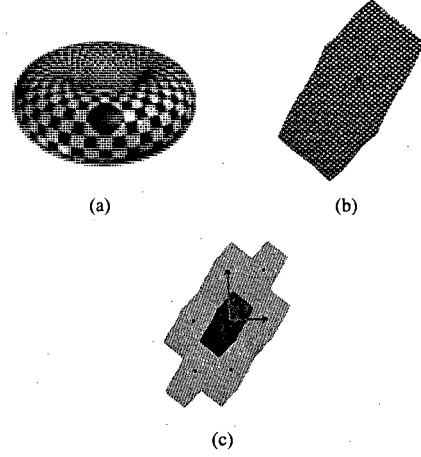


Figure 1: Periodicity of the domain. a) The mapped mesh. b) The fundamental domain. c) The periodic structure of the domain with fundamental parallelogram vectors joining equivalent points.

this purpose, we will define two vectors forming the fundamental parallelogram [GY03]. These are a pair of independent vectors representing the shortest non-null translations needed to get back to an equivalent point on the parameter domain (see Figure 1).

This equivalence relation is expressed by Equation 11: two points are equivalent if they are separated by an integral linear combination of the fundamental parallelogram vectors \vec{b}_0 and \vec{b}_1 .

$$\vec{p} \equiv \vec{p} + i\vec{b}_0 + j\vec{b}_1, \forall i, j \in \mathbb{Z} \quad (11)$$

These vectors are simply computed by integrating one-forms over the homology bases using Equation 12.

$$\vec{b}_i = \sum_{e_{jk} \in B_i} (\sigma_{jk}, \tau_{jk}) \quad (12)$$

3.3. Metrics

A mapping generally induces local deformations that need to be taken into account in doing computations on the parameter domain. A metric g is a bilinear function which takes as input two tangent vectors and returns a scalar, generalizing the dot product in Euclidean space. Since a bilinear function can be represented by a matrix, we can build a 2×2 matrix with components g_{ij} called the *metric tensor*. We denote the components of the inverse of the metric tensor as g^{ij} .

Since a conformal map only locally induces a uniform stretch, the metric tensor is proportional to the metric tensor of the parameter domain, which is, in our case, the identity matrix. The constant of proportionality is called the conformal factor and is denoted by λ^2 . It is generally different for every point on the object and it is the only value that will be needed to compute lengths and areas in the parameter domain.

In the discrete setting of an object represented by a triangle mesh, a conformal factor will be computed for every vertex v_i of M . Concretely, λ represents the ratio of a unit of length on the mesh to a unit of length on the parameter domain. They are computed once the parameterization procedure is done using the formula in Equation 13 from [JWYG04].

$$\lambda_i = \frac{1}{n} \sum_{e_{ij} \in M} \frac{\|v_i - v_j\|}{\|(\sigma_{ij}, \tau_{ij})\|} \quad (13)$$

where n is the number of incident edges to the vertex v_i .

3.4. Affine connections

Differential calculus on manifolds involves the notion of parallel transport for vector quantities. Parallelism on a curved surface needs to be defined using the affine connection of the metric [dC92]. This is a map that encodes changes in the basis used to represent a vector from point to point. They are needed to compute directional derivatives of a vector quantity like the velocity of a particle, see section 5.3. For a surface admitting a Riemannian metric g , as in our case, the connection is represented with the Christoffel symbols, as defined in Equation 14. We make use of the Einstein summation convention where repeated indices are implicitly summed over. For clarity, derivatives are denoted by $\frac{\partial g_{mj}}{\partial k} = \partial_k g_{mj}$.

$$\Gamma_{jk}^i = \frac{g^{im}}{2} (\partial_k g_{mj} + \partial_j g_{mk} - \partial_m g_{jk}) \quad (14)$$

For a conformal map, the expression of the connection is pretty simple. The components of Γ being a combination of the components of $\nabla \ln \lambda$, we only need to compute this gradient at each vertex in the parameter domain. For a vertex v_i having n neighboring vertices $\{w_j\}$, we can approximate the gradient with the pseudoinverse formula in Equation 15:

$$\nabla \ln \lambda_{v_i} = (W^T W)^{-1} W^T \vec{d} \quad (15)$$

where the j th row W_j of the matrix W and \vec{d} are defined as in equations 16 and 17.

$$W_j = w_j - v_i, 0 \leq j < n \quad (16)$$

$$\vec{d}_j = \ln \lambda_{w_j} - \ln \lambda_{v_i}, 0 \leq j < n \quad (17)$$

4. SPH Basics

SPH is an approximation method that defines a continuous function from an unorganized set of points. To each point in the set is associated a mass m_j and a density ρ_j . To approximate the function at a position \vec{r} , a convolution is done with the surrounding points inside a radius h , the *smoothing length*. The value of a property A contained at each point is weighted by the *kernel function* W and summed over a local neighborhood.

$$\langle A(\vec{r}) \rangle = \sum_{j \in N(\vec{r})} \frac{m_j A_j}{\rho_j} W(\vec{r} - \vec{r}_j, h)$$

The masses m_j are given while the densities ρ_j need to be computed using Equation 18 before being able to compute any other property.

$$\begin{aligned} \rho_i &= \langle \rho(\vec{r}_i) \rangle \\ &= \sum_{j \in N(\vec{r}_i)} \frac{m_j \rho_j}{\rho_j} W(\vec{r}_i - \vec{r}_j, h) \\ &= \sum_{j \in N(\vec{r}_i)} m_j W(\vec{r}_i - \vec{r}_j, h) \end{aligned} \quad (18)$$

4.1. Derivatives

Having defined a continuous function over the whole space, it is now easy to define the gradient and the Laplacian of any property A . After some manipulations, one can show that the discretized gradient and Laplacian are given by equations 19 and 20.

$$\nabla \langle A(\vec{r}) \rangle = \sum_{j \in N(\vec{r})} \frac{m_j A_j}{\rho_j} \nabla W(\vec{r} - \vec{r}_j, h) \quad (19)$$

$$\Delta \langle A(\vec{r}) \rangle = \sum_{j \in N(\vec{r})} \frac{m_j A_j}{\rho_j} \Delta W(\vec{r} - \vec{r}_j, h) \quad (20)$$

4.2. Kernels

The kernel functions used in the constitutive equations of SPH need to follow some rules to give a good approximation of the desired result. These properties are as follows:

- $\int W(\vec{r}, h) dV = 1$ Unity
- $\int W(\vec{r}, h) \vec{r} dV = 0$ Symmetry
- $W(\vec{r}, h)|_{\|\vec{r}\| < h} > 0$ Non-negativity
- $W(\vec{r}, h)|_{\|\vec{r}\| \geq h} = 0$ Null outside neighborhood

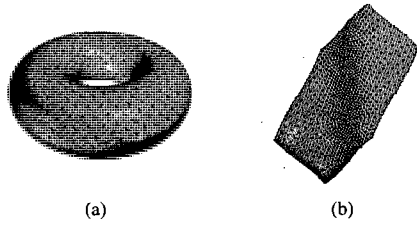


Figure 2: Particle distribution. A uniform particle distribution on the surface (a) induces a non-uniform distribution in the parameter domain (b) due to the distortions caused by the parameterization.

where the last condition is only introduced to get an algorithm of linear complexity.

The kernel functions need not be the same for every computation. For instance, three different kernels will be used for each property. These functions are chosen for their simplicity, speed of computation and stability. Density computation uses W_{poly6} while the pressure and viscosity computations use W_{spiky} and W_{visc} , respectively.

$$\begin{aligned}
 W_{poly6}(\vec{r}, h) &= \frac{4}{\pi h^8} \begin{cases} (h^2 - r^2)^3, & r < h \\ 0, & \text{otherwise} \end{cases} \\
 W_{spiky}(\vec{r}, h) &= \frac{10}{\pi h^5} \begin{cases} (h - r)^3, & r < h \\ 0, & \text{otherwise} \end{cases} \\
 W_{visc}(\vec{r}, h) &= \frac{20}{\pi h^5} \begin{cases} -\frac{r^3}{2h^3} + \frac{r^2}{h^2} + \frac{h}{2r} - 1, & r < h \\ 0, & \text{otherwise} \end{cases}
 \end{aligned}$$

where $r = \|\vec{r}\|$.

The coefficients in front of each kernel are necessary to ensure the unity property. They are only valid in 2D and thus are valid for our application on surfaces.

5. SPH on a surface

The first thing to do in order to simulate fluid flow on the surface of an object is to adapt the Navier-Stokes equations. Since the particles will be moving on the parameter domain, the distortions caused by the parameterization must be considered (see Figure 2). They appear in the computation of the gradient and the Laplacian, and of the path taken by particles.

From now on, we need to be careful with the properties we are manipulating, since there are two different spaces in which they can "live": the surface and the parameter domain. Properties that live on the parameter domain will be written A and those that live on the surface will be written \tilde{A} to avoid any confusion.

The new set of equations (see [ARI89]) benefits from the nice properties of the global conformal mapping.

$$\tilde{\rho} \frac{D\tilde{u}}{Dt} = \underbrace{-\frac{1}{\lambda^2} \nabla \tilde{p}}_{\text{pressure}} + \underbrace{\mu \frac{1}{\lambda^2} \Delta \tilde{u}}_{\text{viscosity}} + \underbrace{G(\tilde{u})}_{\text{geodesic}} + \underbrace{\tilde{F}}_{\text{external}} \quad (21)$$

We see that the only differences with the original Navier-Stokes equations are the use of the conformal factors λ^2 and the addition of the geodesic force. This force comes from the parallel transport of the velocity to ensure that a particle will follow a geodesic path on the surface. Details about this force will be given in Section 5.3.

Information on metrics and connections is only known at vertices of the mesh. Since a particle is necessarily contained in a triangle of the mesh, these properties can be found by doing barycentric interpolation of the values contained at each vertex.

5.1. Corrected formulas

First, we have already pointed out that a conformal parameterization maps circles to circles of different radius. This means that we can still use a circular neighborhood in the parameter domain, but we need to change the smoothing length by some factor. Noting that a unit of length in the parameter domain is stretched by a factor of λ to be valid on the surface, we need to define the new radius by Equation 22.

$$h_j = \frac{\tilde{h}}{\lambda_j} \quad (22)$$

where \tilde{h} is the global smoothing length, λ_j is the stretch factor at the position of particle j and h_j is the local smoothing length of particle j .

We can also point out that if lengths are stretched by a factor λ , areas will be stretched by a factor λ^2 . For this reason, the area occupied by a particle in the parameter domain is computed by Equation 23.

$$V_j = \frac{\tilde{m}_j}{\tilde{\rho}_j \lambda_j^2} \quad (23)$$

As we wish to do every computation in only one domain and Equation 21 is written in such a way that the differential operators are to be applied in the parameter domain, the new SPH formulation will be described by Equation 24.

$$\langle A(\vec{r}) \rangle = \sum_{j \in \mathcal{N}(\vec{r})} \frac{\tilde{m}_j A_j}{\tilde{\rho}_j \lambda_j^2} W \left(\tilde{d}(\vec{r}, \vec{r}_j), \frac{\tilde{h}}{2} \left(\frac{1}{\lambda(\vec{r})} + \frac{1}{\lambda_j} \right) \right) \quad (24)$$

where $\vec{d}(\vec{r}, \vec{r}_j)$ means the vector representing the shortest straight path between \vec{r} and \vec{r}_j . This subtlety will be fully discussed in Section 5.2.

Given this new formulation, it is easy to define the density, gradient and Laplacian on the parameter domain using equations 25, 26 and 27.

$$\tilde{\rho}_i = \sum_{j \in N(\vec{r})} \frac{\tilde{m}_j}{\lambda_j^2} W(\vec{d}(\vec{r}_i, \vec{r}_j), h_{ij}) \quad (25)$$

$$\frac{1}{\lambda_i^2} \nabla \langle \tilde{p}(\vec{r}_i) \rangle = \frac{1}{\lambda_i^2} \sum_{j \in N(\vec{r})} \frac{\tilde{m}_j \tilde{p}_j}{\tilde{\rho}_j \lambda_j^2} \nabla W(\vec{d}(\vec{r}_i, \vec{r}_j), h_{ij}) \quad (26)$$

$$\frac{1}{\lambda_i^2} \Delta \langle \tilde{u}(\vec{r}_i) \rangle = \frac{1}{\lambda_i^2} \sum_{j \in N(\vec{r})} \frac{\tilde{m}_j}{\tilde{\rho}_j \lambda_j^2} (\tilde{u}_j - \tilde{u}_i) \Delta W(\vec{d}(\vec{r}_i, \vec{r}_j), h_{ij}) \quad (27)$$

where $h_{ij} = (h_i + h_j)/2$ and $\tilde{p}_j = k\tilde{\rho}_j$, with k being a constant.

5.2. Distance computation

As explained in Section 3.2.1, the distance between two points on the global conformal domain needs to take into consideration the periodic structure of the domain. Due to this periodicity, multiple straight paths can be drawn between two points on the parameter domain, as illustrated in Figure 1. Only the shortest path is relevant to the computation and this is the path that will be used for the SPH formulation.

For genus-0 surfaces, the mesh is mapped to a sphere. In this case, the Euclidean distance is a good approximation of the real distance and naturally gives the shortest distance.

For genus-1 surfaces, the mesh is mapped to a periodic flat domain. We need to look for the closest translated copy of a particle \tilde{p}_j around the particle position \tilde{p}_i . This is done by translating \tilde{p}_j by integer values of the fundamental parallelogram vectors defined in Section 3.2.1 and taking the difference relative to the closest translated position of the particle. Mathematically, this path vector is defined by Equation 28.

$$\vec{d}(\tilde{p}_1, \tilde{p}_2) = \vec{d}_{ij}, \quad \|\vec{d}_{ij}\| = \min_{k,l \in \{-1,0,1\}} \|\vec{d}_{kl}\| \quad (28)$$

where $\vec{d}_{ij} = \tilde{p}_2 - \tilde{p}_1 + i\vec{b}_1 + j\vec{b}_2$

5.3. Geodesic force

The path of a free particle moving on a manifold without force acting on it will have zero geodesic curvature since, by definition, the geodesic curvature is the acceleration of a curve in the tangent space of the manifold. This implies that if there is no force, i.e. no acceleration, the free particle must follow a geodesic path.

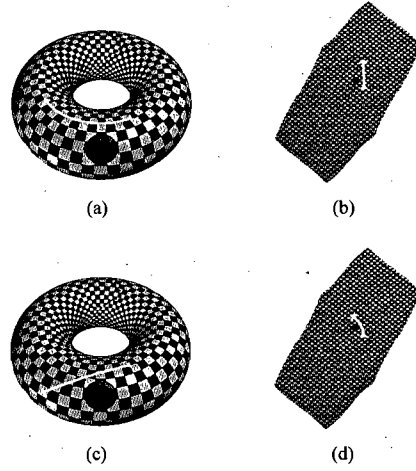


Figure 3: Effect of the geodesic force. a) and b) show the path taken by a particle without geodesic force while c) and d) show the path taken by a particle taking geodesic force into account.

This intuitive fact is expressed in mathematical terms by $\frac{\tilde{D}\tilde{u}}{\tilde{D}t} = 0$ with the material derivative on the surface $\frac{\tilde{D}}{\tilde{D}t}$ defined as in Equation 29 for a vector quantity [dC92].

$$\begin{aligned} \frac{\tilde{D}u^\alpha}{\tilde{D}t} &= \frac{\partial u^\alpha}{\partial t} + \tilde{\nabla}_{\tilde{u}} u^\alpha \\ &= \frac{\partial u^\alpha}{\partial t} + (\tilde{u} \cdot \nabla) u^\alpha + \Gamma_{\beta\gamma}^\alpha u^\beta u^\gamma \\ &= \frac{Du^\alpha}{Dt} + \Gamma_{\beta\gamma}^\alpha u^\beta u^\gamma \end{aligned} \quad (29)$$

for $\alpha \in \{1, 2\}$, where $\frac{D}{Dt}$ is the material derivative on the parameter domain and $\tilde{\nabla}_{\tilde{u}}$ is the covariant derivative.

What Equation 29 means is that if we want to move the particles directly on the parameter domain, we have to add an artificial force that corresponds to the parallel transport of the velocity. This results in particles following a geodesic path. Figure 3 shows the difference in the path taken by a particle with and without the geodesic force.

The geodesic force is thus a correction of the path taken by the particle on the parameter domain. This force is expressed by Equation 30.

$$G(\tilde{u}_i)^\alpha = -\rho_i \Gamma_{\beta\gamma}^\alpha u^\beta u^\gamma \quad (30)$$

Once again, the conformal mapping simplifies this fairly complex equation. The close relationship between conformal maps and complex analysis lets us derive a very simple

| Nb triangles | Steps/second | Simulation time |
|--------------|--------------|-----------------|
| 1250 | 189 | 5.29ms |
| 12800 | 182 | 5.49ms |
| 31250 | 182 | 5.49ms |

Table 1: Simulation time of 1000 particles on a torus of different resolutions.

equation for the geodesic force on a planar domain. Using the definition of the Christoffel symbols and some algebraic manipulations, we get Equation 31.

$$G_c^{plane}(\vec{u}_i) = -\rho_i F_c u_c^2 \quad (31)$$

where $F_c = (\nabla \ln \lambda)_x - i(\nabla \ln \lambda)_y$ and $u_c = u_x + iu_y$.

The above equation is computed on the parameter domain and is only valid for a planar domain. But genus-0 surfaces are mapped to a sphere, which requires us to change our perspective on the problem. By performing a simple change of basis, we obtain the new velocity \vec{v} in the new basis B built to be aligned with $\nabla \ln \lambda$. With these values defined, we can finally compute the geodesic force that is valid for spherical domains by Equation 32.

$$G_c^{sphere}(\vec{u}_i) = -\rho_i (\nabla \ln \lambda)^2 B \begin{bmatrix} \vec{v}_x^2 - \vec{v}_y^2 \\ 2\vec{v}_x \vec{v}_y \\ 0 \end{bmatrix} \quad (32)$$

where B and \vec{v} are defined as below.

$$B = \begin{pmatrix} (\nabla \ln \lambda)_x & (\vec{n} \times \nabla \ln \lambda)_x & (\vec{n})_x \\ (\nabla \ln \lambda)_y & (\vec{n} \times \nabla \ln \lambda)_y & (\vec{n})_y \\ (\nabla \ln \lambda)_z & (\vec{n} \times \nabla \ln \lambda)_z & (\vec{n})_z \end{pmatrix}$$

$$\vec{v} = B^{-1} \vec{u}$$

6. Results

The method is now applied to different situations, to confirm that simulation time is in fact independent of the mesh resolution, and to demonstrate free-surface flows. Various other examples will also be presented to complete the section. All results presented in this section were obtained with a 2.6GHz Intel Core 2 Quad processor.

The resolution independence is demonstrated in Table 1 where the simulation time for a single frame is displayed for every resolution of a given mesh. A torus is used as the simulation domain in this example.

Figure 4 demonstrates free-surface flows over a torus and

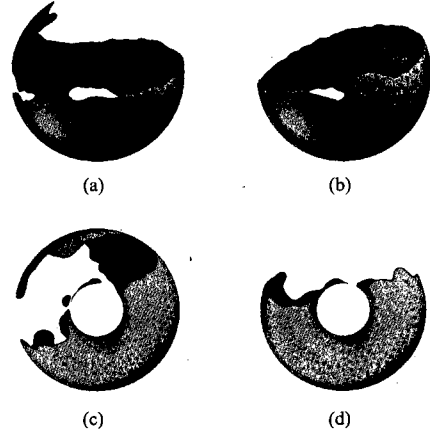


Figure 4: Free-surface flow over a sphere (a,b) and a torus (c,d).

a sphere. The gravitational force is transposed to the parameter domain and is applied to each particle. Surface tension is added using the color field from [MCG03] adapted to the formalism presented in Section 5.1. In this example, the visualization is done by sampling the density of the fluid on each vertex of the mesh and using a marching squares-like method to extract the interface of the fluid.

Various other examples are shown in Figure 5. External forces are added by dragging the mouse over the mesh or over the parameter domain.

7. Conclusion

In this paper, we have presented a particle-based method for simulating fluid flows on the surface of an object of genus 0 or 1. The idea is to conformally map the triangle mesh to a constant curvature domain and solve the Navier-Stokes equations on the parameter domain. The uniform stretching property of conformal maps ensures minimal changes to these equations. The reformulation of the advection term lets the particles follow the geodesic path without distortions caused by the parameterization.

Results show that the simulation time is independent of the resolution of the mesh, so real-time performance is achieved even with high resolution triangle mesh. The method was used to simulate free-surface flows by taking advantage of the natural surface tracking that a particle-based simulation offers.

References

[AHTK99] ANGENENT S., HAKER S., TANNENBAUM A., KIKINIS R.: Conformal geometry and brain flattening. In *MIC-*

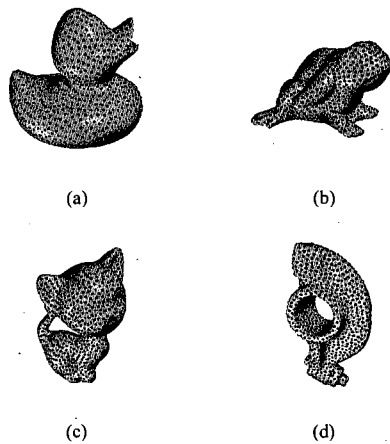


Figure 5: Particle simulation over various complex objects.

- CAI '99: *Proceedings of the Second International Conference on Medical Image Computing and Computer-Assisted Intervention* (London, UK, 1999), Springer-Verlag, pp. 271–278. 17
- [ARI89] ARIS R.: *Vectors, tensors and the basis equations of fluid mechanics*. Dover, New-York, 1989. 21
- [dC92] DO CARMO M. P.: *Riemannian Geometry*. Birkhäuser, Boston, MA, 1992. 20, 22
- [DE09] DJADO K., EGLI R.: Particle-based fluid flow visualization on meshes. In *AFRIGRAPH '09: Proceedings of the 6th International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa* (New York, NY, USA, 2009), ACM, pp. 65–72. 17
- [DG96] DESBRUN M., GASCUEL M.-P.: Smoothed particles: A new paradigm for animating highly deformable bodies. In *Proceedings of the Eurographics Workshop on Computer Animation and Simulation '96* (New York, NY, USA, 1996), Springer-Verlag New York, Inc., pp. 61–76. 17
- [ETK*07] ELCOTT S., TONG Y., KANSO E., SCHRÖDER P., DESBRUN M.: Stable, circulation-preserving, simplicial fluids. *ACM Trans. Graph.* 26, 1 (2007), 4. 17
- [FZKH05] FAN Z., ZHAO Y., KAUFMAN A., HE Y.: Adapted unstructured lbm for flow simulation on curved surfaces. In *SCA '05: Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2005), ACM, pp. 245–254. 17
- [Gib77] GIBLIN P.: *Graphs, Surfaces and Homology*. Chapman & Hall, London, 1977. 18
- [GM77] GINGOLD R. A., MONAGHAN J. J.: Smoothed particle hydrodynamics - Theory and application to non-spherical stars. *Royal Astronomical Society, Monthly Notices* 181 (Nov. 1977), 375–389. 17
- [GtY02] GU X., TUNG YAU S.: Computing conformal structures of surfaces. *Communications in Information and Systems* 2 (2002), 121–146. 17
- [Gu03] GU X.: *Parametrization for surfaces with arbitrary topologies*. PhD thesis, Harvard University, Cambridge, MA, USA, 2003. Adviser-Gortler, Steven J. 17, 19
- [GWC*03] GU X., WANG Y., CHAN T. F., THOMPSON P. M., TUNG YAU S.: Genus zero surface conformal mapping and its application to brain surface mapping. *IEEE Transactions on Medical Imaging* 23 (2003), 949–958. 17, 18
- [GY03] GU X., YAU S.-T.: Global conformal surface parameterization. In *SGP '03: Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing* (Aire-la-Ville, Switzerland, Switzerland, 2003), Eurographics Association, pp. 127–137. 17, 18, 19
- [HAT*00] HAKER S., ANGENENT S., TANNENBAUM A., KIKINIS R., SAPIRO G., HALLE M.: Conformal surface parameterization for texture mapping. *IEEE Transactions on Visualization and Computer Graphics* 6, 2 (2000), 181–189. 17
- [JWYG04] JIN M., WANG Y., YAU S.-T., GU X.: Optimal global conformal surface parameterization. In *VIS '04: Proceedings of the Conference on Visualization '04* (Washington, DC, USA, 2004), IEEE Computer Society, pp. 267–274. 17, 20
- [KMM01] KACZYNSKI T., MISCHAIKOW K., MROZEK M.: *Computing Homology*. Springer, 2001. 18
- [LH07] LI H., HARTLEY R.: Conformal spherical representation of 3d genus-zero meshes. *Pattern Recogn.* 40, 10 (2007), 2742–2753. 17, 18
- [Luc77] LUCY L. B.: A numerical approach to the testing of the fission hypothesis. *Astronomical Journal* 82 (Dec. 1977), 1013–1024. 17
- [LWC05] LUI L. M., WANG Y., CHAN T. F.: Solving pdes on manifolds with global conformal parametrization. In *VLISM* (2005), pp. 307–319. 17
- [MCG03] MÜLLER M., CHARYPAR D., GROSS M.: Particle-based fluid simulation for interactive applications. In *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland, 2003), Eurographics Association, pp. 154–159. 17, 23
- [SAC*99] STORA D., AGLIATI P.-O., CANI M.-P., NEYRET F., GASCUEL J.-D.: Animating lava flows. In *Graphics Interface, GI '99, June, 1999* (Kingston, Ontario, Canada, June 1999), Mackenzie I. S., Stewart J., (Eds.), Canadian Human-Computer Communications Society, pp. 203–210. 17
- [Sta99] STAM J.: Stable fluids. In *SIGGRAPH '99: Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1999), ACM Press/Addison-Wesley Publishing Co., pp. 121–128. 17
- [Sta03] STAM J.: Flows on surfaces of arbitrary topology. In *SIGGRAPH '03: ACM SIGGRAPH 2003 Papers* (New York, NY, USA, 2003), ACM, pp. 724–731. 17
- [SY04] SHI L., YU Y.: Inviscid and incompressible fluid simulation on triangle meshes: Research articles. *Comput. Animat. Virtual Worlds* 15, 3-4 (2004), 173–181. 17

Conclusion

Dans ce mémoire, une méthode particulière de simulation de l'écoulement d'un fluide à la surface d'un objet de genre 0 ou 1 a été présentée. Il est montré qu'en calculant une paramétrisation conforme de la surface, il est possible de résoudre les équations de Navier-Stokes directement sur le domaine paramétrique en temps réel à l'aide d'une généralisation de la méthode *Smoothed Particle Hydrodynamics*.

La propriété principale d'une paramétrisation conforme est l'étirement local uniforme, permettant la conservation des angles. Elle permet de simplifier grandement les opérateurs différentiels transposés sur la surface de l'objet. L'étirement uniforme permet aussi de considérer qu'un cercle sur la surface restera un cercle dans le domaine paramétrique. Ceci permet une modification minimale de la formulation SPH originale. La première partie du travail a été d'implémenter les méthodes de paramétrisation d'un objet de genre 0 sur la sphère et d'un objet de genre 1 dans le plan. La deuxième partie a été de généraliser la méthode de simulation particulière *Smoothed Particle Hydrodynamics* pour considérer les déformations induites par la paramétrisation. Le terme d'advection a été correctement calculé en considérant la trajectoire géodésique empruntée par une particule, ce qui avait causé des distorsions dans quelques méthodes précédentes [15] [13].

Une méthode particulière permet l'indépendance du temps de simulation et de la résolution du maillage de l'objet sur lequel le fluide s'écoule. Cette propriété permet d'obtenir des temps de calcul de l'ordre du temps réel, ce qui la différencie des autres méthodes qui ne fonctionnent qu'en temps différé et qui dépendent de la résolution du maillage de l'objet [15][14][13][5][4]. Elle permet aussi un traitement naturel de l'interface liquide/liquide et liquide/gaz, ce qui n'est pas possible avec les autres méthodes [15][14][13][5][4][3] sans

CONCLUSION

procédure supplémentaire. La méthode génère son propre champ de vitesse et ne permet pas aux particules de s'éloigner de la surface étant donné qu'elles se déplacent dans le domaine paramétrique, contrairement à la visualisation par particules décrite dans [3].

Annexe A

Force géodésique

Dans cette annexe, les dérivations des équations 31 et 32 de l'article seront présentées. Il s'agit de la force géodésique appliquée à chaque particule. La section A.1 démontre que les équations 30 et 31 sont équivalentes alors que la section A.2 est la dérivation détaillée du changement de base utilisé dans l'article.

A.1 Cas planaire

Montrons que les équations 30 et 31 sont équivalentes. On doit d'abord noter que selon la définition de la métrique et des connexions affines, les symboles de Christoffel sont les suivants :

$$\begin{aligned}\Gamma_{11}^1 &= \partial_1 \ln \lambda & \Gamma_{12}^1 &= \partial_2 \ln \lambda & \Gamma_{21}^1 &= \partial_2 \ln \lambda & \Gamma_{22}^1 &= -\partial_1 \ln \lambda \\ \Gamma_{11}^2 &= -\partial_2 \ln \lambda & \Gamma_{12}^2 &= \partial_1 \ln \lambda & \Gamma_{21}^2 &= \partial_1 \ln \lambda & \Gamma_{22}^2 &= \partial_2 \ln \lambda\end{aligned}$$

Pour illustrer le calcul de ces symboles, prenons l'exemple de Γ_{11}^1 . La métrique est définie comme étant $g_{ij} = \lambda^2 \delta_{ij}$ et son inverse étant $g^{ij} = \frac{1}{\lambda^2} \delta^{ij}$. Selon l'équation 14 de l'article, on a que

A.1. CAS PLANAIRE

$$\begin{aligned}
\Gamma_{11}^1 &= \frac{g^{1m}}{2} (\partial_1 g_{m1} + \partial_1 g_{m1} - \partial_m g_{11}) \\
&= \frac{\delta^{1m}}{2\lambda^2} (2\delta_{m1}\partial_1\lambda^2 - \delta_{11}\partial_m\lambda^2) \\
&= \frac{1}{2\lambda^2} (2\partial_1\lambda^2 - \partial_1\lambda^2) \\
&= \frac{1}{2\lambda^2} \partial_1\lambda^2 \\
&= \frac{2\lambda}{2\lambda^2} \partial_1\lambda \\
&= \frac{1}{\lambda} \partial_1\lambda \\
&= \partial_1 \ln \lambda
\end{aligned}$$

où la convention de sommation d'Einstein à été utilisée.

Nous pouvons aussi obtenir les autres composantes Γ_{jk}^i par la même démarche. En utilisant ces valeurs, nous pouvons alors calculer les deux composantes de la force géodésique définie par l'équation 30 de l'article. En définissant $f_i = \partial_i \ln \lambda$, nous obtenons

$$\begin{aligned}
G(\vec{u}_i)^1 &= -\rho_i \Gamma_{\beta\gamma}^1 u^\beta u^\gamma \\
&= -\rho_i (f_1 u^1 u^1 + f_2 u^1 u^2 + f_2 u^2 u^1 - f_1 u^2 u^2) \\
&= -\rho_i (f_1 u^1 u^1 + 2f_2 u^1 u^2 - f_1 u^2 u^2) \\
G(\vec{u}_i)^2 &= -\rho_i \Gamma_{\beta\gamma}^2 u^\beta u^\gamma \\
&= -\rho_i (-f_2 u^1 u^1 + f_1 u^1 u^2 + f_1 u^2 u^1 + f_2 u^2 u^2) \\
&= -\rho_i (-f_2 u^1 u^1 + 2f_1 u^2 u^1 + f_2 u^2 u^2)
\end{aligned}$$

D'autre part, la définition de la force géodésique planaire à l'aide des nombres complexes de l'équation 31 nous permet d'obtenir l'équation A.1. En définissant $F_c = f_1 - i f_2$ et $u_c = u^1 + i u^2$, nous obtenons

A.2. CAS SPHÉRIQUE

$$\begin{aligned} G_c^{plane}(\vec{u}_i) &= -\rho_i F_c u_c^2 \\ &= -\rho_i F_c [(u^1 u^1 - u^2 u^2) + i(2u^1 u^2)] \\ &= -\rho_i [(f_1 u^1 u^1 + 2f_2 u^1 u^2 - f_1 u^2 u^2) + i(-f_2 u^1 u^1 + 2f_1 u^2 u^1 + f_2 u^2 u^2)] \\ &= G(\vec{u}_i)^1 + iG(\vec{u}_i)^2 \end{aligned} \tag{A.1}$$

Ainsi, nous pouvons voir qu'il s'agit simplement de la force géodésique écrite sous sa forme complexe.

□

A.2 Cas sphérique

Avant de s'attaquer au cas de la sphère, regardons comment l'équation de la force géodésique réagit lorsqu'on applique une transformation linéaire dans le plan. Tout d'abord, il serait plus utile de manipuler des matrices. Nous réécrivons donc les symboles de Christoffel de la façon suivante :

$$\Gamma^1 = \begin{pmatrix} f_1 & f_2 \\ f_2 & -f_1 \end{pmatrix} \tag{A.2}$$

$$\Gamma^2 = \begin{pmatrix} -f_2 & f_1 \\ f_1 & f_2 \end{pmatrix} \tag{A.3}$$

où $f_i = \partial_i \ln \lambda$.

Lors d'un changement de repère linéaire représenté par la matrice A , les symboles de Christoffel réagissent de la façon suivante [1] :

A.2. CAS SPHÉRIQUE

$$\tilde{\Gamma}^i = \sum_j (A^{-1})_{ij} (A^T \Gamma^j A) \quad (\text{A.4})$$

Nous pouvons espérer obtenir une formulation simple de ces symboles en sélectionnant soigneusement la base A utilisée. En effet, nous pouvons construire une base orthogonale dont un des vecteurs est $\nabla \ln \lambda$:

$$A = \begin{pmatrix} f_1 & -f_2 \\ f_2 & f_1 \end{pmatrix} \quad (\text{A.5})$$

$$A^{-1} = \frac{1}{\|\nabla \ln \lambda\|^2} \begin{pmatrix} f_1 & f_2 \\ -f_2 & f_1 \end{pmatrix} \quad (\text{A.6})$$

Dans ce nouveau repère, les symboles de Christoffel et la nouvelle vitesse \vec{w} sont les suivants :

$$\tilde{\Gamma}^1 = \|\nabla \ln \lambda\|^2 \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad (\text{A.7})$$

$$\tilde{\Gamma}^2 = \|\nabla \ln \lambda\|^2 \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad (\text{A.8})$$

$$\vec{w} = A^{-1} \vec{u} \quad (\text{A.9})$$

En utilisant cette nouvelle formulation, nous obtenons l'équation de la force géodésique dans le nouveau repère :

A.2. CAS SPHÉRIQUE

$$G^A(\vec{u}_i) = -\rho_i \begin{pmatrix} \vec{w}^T \tilde{\Gamma}^1 \vec{w} \\ \vec{w}^t \tilde{\Gamma}^2 \vec{w} \end{pmatrix} \quad (\text{A.10})$$

$$= -\rho_i \|\nabla \ln \lambda\|^2 \begin{pmatrix} w^1 w^1 - w^2 w^2 \\ 2w^1 w^2 \end{pmatrix} \quad (\text{A.11})$$

Comme nous désirons le vecteur de force dans le repère original, nous multiplions ce résultat par la matrice A .

$$G(\vec{u}_i) = -\rho_i \|\nabla \ln \lambda\|^2 A \begin{pmatrix} w^1 w^1 - w^2 w^2 \\ 2w^1 w^2 \end{pmatrix} \quad (\text{A.12})$$

Au final, nous avons seulement changé la formulation du problème afin de se débarrasser de la dépendance sur une seule coordonnée du vecteur $\nabla \ln \lambda$. De cette façon, nous obtenons une équation n'utilisant que la longueur et la direction de ce vecteur. Cette nouvelle formulation qui nous a laissé voir la géométrie derrière cette force nous permet de définir la force dans un plan tangent de la sphère en définissant une nouvelle base B .

$$B = \begin{pmatrix} (\nabla \ln \lambda)_1 & (\vec{n} \times \nabla \ln \lambda)_1 & (\vec{n})_1 \\ (\nabla \ln \lambda)_2 & (\vec{n} \times \nabla \ln \lambda)_2 & (\vec{n})_2 \\ (\nabla \ln \lambda)_3 & (\vec{n} \times \nabla \ln \lambda)_3 & (\vec{n})_3 \end{pmatrix}$$

où \vec{n} est le vecteur normal au plan tangent.

Cette matrice fait exactement le même travail que la matrice A , c'est-à-dire qu'elle sélectionne un repère orthogonal alignant un vecteur de la base avec le vecteur $\nabla \ln \lambda$. Dans ce nouveau repère, la vitesse de la particule i est décrite par l'équation suivante :

A.2. CAS SPHÉRIQUE

$$\vec{v} = B^{-1}\vec{u}$$

Étant donné que \vec{u} et la force géodésique se situent dans le plan tangent, nous pouvons reprendre l'équation A.10 en considérant la coordonnée normale nulle.

$$G^B(\vec{u}_i) = -\rho_i \|\nabla \ln \lambda\|^2 \begin{pmatrix} v^1 v^1 - v^2 v^2 \\ 2v^1 v^2 \\ 0 \end{pmatrix}$$

Encore une fois, nous désirons obtenir la force dans le plan tangent de la sphère. Nous multiplions par la base B pour finalement obtenir la formulation utilisée dans l'article.

$$G^{sphere}(\vec{u}_i) = -\rho_i \|\nabla \ln \lambda\|^2 B \begin{pmatrix} v^1 v^1 - v^2 v^2 \\ 2v^1 v^2 \\ 0 \end{pmatrix}$$

Bibliographie

- [1] Rutherford ARIS. Vectors, tensors and the basis equations of fluid mechanics. Dover, New-York, 1989.
- [2] Manfredo P. Do CARMO. Differential Geometry of Curves and Surfaces. Prentice-Hall, Englewood Cliffs, NJ, 1976.
- [3] Khalid DJADO et Richard EGLI. « Particle-based fluid flow visualization on meshes ». Dans AFRIGRAPH '09 : Proceedings of the 6th International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa, pages 65–72, New York, NY, USA, 2009. ACM.
- [4] Sharif ELCOTT, Yiyong TONG, Eva KANSO, Peter SCHRÖDER et Mathieu DESBRUN. « Stable, circulation-preserving, simplicial fluids ». ACM Trans. Graph., 26(1) :4, 2007.
- [5] Z. FAN, Y. ZHAO, A. KAUFMAN et Y. HE. « Adapted unstructured LBM for flow simulation on curved surfaces ». Dans SCA '05 : Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pages 245–254, New York, NY, USA, 2005. ACM.
- [6] R. A. GINGOLD et J. J. MONAGHAN. « Smoothed particle hydrodynamics - Theory and application to non-spherical stars ». Royal Astronomical Society, Monthly Notices, 181 :375–389, novembre 1977.
- [7] Xianfeng GU. « Parametrization for surfaces with arbitrary topologies ». Thèse de doctorat, Harvard University, Cambridge, MA, USA, 2003.

BIBLIOGRAPHIE

- [8] Xianfeng GU, Yalin WANG, Tony F. CHAN, Paul M. THOMPSON et Shing tung YAU. « Genus Zero Surface Conformal Mapping and Its Application to Brain Surface Mapping ». IEEE Transactions on Medical Imaging, 23 :949–958, 2003.
- [9] Xianfeng GU et Shing-Tung YAU. « Global conformal surface parameterization ». Dans SGP '03 : Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, pages 127–137, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.
- [10] Hongdong LI et Richard HARTLEY. « Conformal spherical representation of 3D genus-zero meshes ». Pattern Recogn., 40(10) :2742–2753, 2007.
- [11] M. B. LIU et G. R. LIU. « Restoring particle consistency in smoothed particle hydrodynamics ». Appl. Numer. Math., 56(1) :19–36, 2006.
- [12] L. B. LUCY. « A numerical approach to the testing of the fission hypothesis ». Astronomical Journal, 82 :1013–1024, décembre 1977.
- [13] Lok Ming LUI, Yalin WANG et Tony F. CHAN. « Solving PDEs on Manifolds with Global Conformal Parametrization ». Dans VLSM, pages 307–319, 2005.
- [14] Lin SHI et Yizhou YU. « Inviscid and incompressible fluid simulation on triangle meshes : Research Articles ». Comput. Animat. Virtual Worlds, 15(3-4) :173–181, 2004.
- [15] Jos STAM. « Flows on surfaces of arbitrary topology ». Dans SIGGRAPH '03 : ACM SIGGRAPH 2003 Papers, pages 724–731, New York, NY, USA, 2003. ACM.