

UNIVERSITÉ DE TECHNOLOGIE DE COMPIÈGNE

INITIATION À LA MÉTHODOLOGIE DE LA RECHERCHE

LA MODÉLISATION DES VAGUES : POURQUOI, COMMENT ?

Simon Chabot — Gl01

Sous la direction de :

M.	MOTTELET STÉPHANE	UTC	(Enseignant référent)
M.	VILLON PIERRE	UTC	(Correspondant du jury final)

Automne 2011

Document publié sous licence *Creative-Commons BY-NC*

*« Les plus belles découvertes cesseraient
de me plaire si je devais les garder pour moi. »*

Sénèque

Document publié sous licence *Creative-Commons BY-NC*

Vous êtes libres de :

- *Partager* : reproduire, distribuer et communiquer l'œuvre.
- *Remixer* : adapter l'œuvre.

Selon les conditions suivantes :

- *Attribution* : Vous devez attribuer l'œuvre de la manière indiquée par l'auteur de l'œuvre ou le titulaire des droits (mais pas d'une manière qui suggérerait qu'ils vous soutiennent ou approuvent votre utilisation de l'œuvre).
- *Pas d'utilisation commerciale* : Vous n'avez pas le droit d'utiliser cette œuvre à des fins commerciales.

<http://creativecommons.org/licenses/by-nc/3.0/fr/>

REMERCIEMENTS

Je tiens tout particulièrement à remercier Stéphane Mottelet pour avoir formulé le sujet à l'origine de cette étude, un sujet qui a été tout autant intéressant que chronophage.

Mais aussi Daniel, pour ses relectures attentives et ses critiques. Je le remercie également pour l'oreille attentive qu'il m'a offerte tout au long de ce semestre. Je le soupçonne d'ailleurs d'avoir feint plusieurs fois d'être intéressé, mais cela m'a permis à plusieurs reprises d'explorer des points que je n'aurai pas étudiés sinon.

Je pense également à Marie, sans qui je n'aurai jamais mis les pieds dans ce cours au début du semestre, et donc jamais pu faire cette étude...

TABLE DES MATIÈRES

TABLE DES MATIÈRES	viii
INTRODUCTION	1
MÉTHODOLOGIE	3
1 INTRODUCTION À LA DESCRIPTION D'UN FLUIDE	5
1.1 DEUX APPROCHES BIEN DIFFÉRENTES	5
1.1.1 Lagrangienne	5
1.1.2 Eulérienne	6
1.2 LA DÉRIVÉE PARTICULAIRE	7
1.3 RENDRE COMPTE DU MOUVEMENT D'UN FLUIDE	8
1.3.1 Équations de Navier - Stokes pour les fluides incompressibles	8
1.3.2 L'équation continue de Boltzmann	10
1.4 BILAN	11
2 MODÉLISATION D'UNE VAGUE	13
2.1 LES ÉQUATIONS DE SAINT-VENANT	13
2.1.1 La méthode de Lax-Wendroff	14
2.1.2 Application de la méthode Lax-Wendroff aux équations de Saint-Venant dans le cas 1D	15
2.2 LA MÉTHODE DE BOLTZMANN SUR RÉSEAU	16
2.2.1 Discrétisation de l'équation de Boltzmann	16
2.2.2 Description de la méthode	18
2.2.3 Un pas vers la modélisation d'un fluide à surface libre	19
2.2.4 Exemple et conclusions sur LBM	21
2.3 SMOOTHED PARTICLE HYDRODYNAMICS	22
2.3.1 <i>Integral and summation interpolant</i>	22
2.3.2 Calcul de la position des particules	24
2.4 BILAN	27
3 COÛTS ET OPTIMISATIONS	29
3.1 QUELQUES OPTIMISATIONS POUR LBM	29
3.1.1 Vers une parallélisation de LBM	29
3.1.2 Autres optimisations	29
3.2 UNE MEILLEURE STRUCTURE DE DONNÉES POUR SPH	30
3.2.1 Recherche dans un <i>arbre-kd</i>	31
3.2.2 Recherche dans une table de hashage	31
3.3 BILAN	34

CONCLUSION GÉNÉRALE	35
A LA MÉTHODE DE BOLTZMANN SUR RÉSEAU	37
A.1 OBTENTION DE LA DISTRIBUTION D'ÉQUILIBRE DE LA FONCTION DE BOLTZMANN	37
A.2 IMPLÉMENTATION DE LBM	37
B SMOOTHED PARTICLE HYDRODYNAMICS	41
B.1 CONVERGENCE VERS $\delta(x)$	41
B.2 EXEMPLE DE NOYAUX UTILISÉS POUR SPH	41
B.2.1 Simulation <i>normale</i>	42
B.2.2 Simulation à faible résolution	42
C DOCUMENTS NÉCESSAIRES À L'UV	45
C.1 NOTE DE CLARIFICATION	45
C.1.1 Contexte, motivation et problématique	45
C.1.2 Démarche employée	45
C.1.3 Calendrier prévisionnel	46
C.2 PLAN PROVISOIRE	46
C.2.1 Introduction à la modélisation d'un fluide	46
C.2.2 Modélisation d'une vague	47
C.2.3 Applications de ces modélisations	50
BIBLIOGRAPHIE	53
LISTE DES FIGURES	54

INTRODUCTION

L'étude du mouvement d'un fluide a toujours été un sujet de recherche très actif. Il y a quelques années, seule la description mathématique du mouvement était possible. Aujourd'hui, avec la puissance croissante des ordinateurs, il est possible d'approcher numériquement la solution de ces équations mathématiques et donc de simuler le mouvement d'un fluide. Plusieurs domaines s'intéressent à cela notamment le génie civil et maritime pour qui l'étude de l'interaction fluide/structure est cruciale. Depuis que le pont de Tacoma s'est écroulé en 1940, des recherches ont été faites du côté de la simulation numérique. En effet, cet effondrement a montré que l'étude sur des maquettes (en soufflerie, pour le cas du pont de Tacoma) n'était pas suffisante. Plus récemment, la simulation de fluide a suscité l'intérêt d'un autre domaine : l'animation. Beaucoup plus axé sur un rendu réaliste que sur l'étude des interactions, le monde de l'animation a fait d'énormes recherches afin de produire une simulation convaincante et réaliste pour saisir le spectateur. Le domaine du jeu vidéo est également intéressé par une animation réaliste, mais il faut en plus que celle-ci puisse se produire en temps réel. Comme nous le verrons, simuler le mouvement d'un fluide demande beaucoup de calculs, la simulation en temps réel est donc un véritable enjeu.

Ce document est ainsi l'occasion de présenter quelques méthodes couramment utilisées pour simuler le mouvement d'un fluide. Dans un premier temps on s'intéressera aux outils mathématiques et équations permettant de décrire ce mouvement. Nous présenterons les deux approches existantes : Eulérienne et Lagrangienne. Dans un second temps, trois méthodes permettant d'approcher la solution des équations vues au chapitre 1 seront étudiées. On s'intéressera à la résolution des équations de Saint - Venant, à la méthode de Boltzmann sur réseau ainsi qu'à la méthode *Smoothed Particle Hydrodynamics* (SPH). Nous finirons par voir quelques techniques permettant d'optimiser ces deux dernières.

MÉTHODOLOGIE

La production de ce document a été pour moi l'occasion de faire un peu connaissance avec la mécanique des fluides. En effet, avant de faire cette UV, je n'avais jamais étudié ce domaine. J'ai donc dû commencer par lire des cours assez généraux afin de mieux comprendre les équations que j'allais être amené à manipuler par la suite. En parallèle de cela, j'ai mené des recherches sur les méthodes utilisées par l'animation. C'est ainsi que j'en suis arrivé à la méthode de Boltzmann sur réseau, utilisée par le moteur de rendu de fluide du logiciel d'animation 3D Blender. À cette même période, M. Mottelet m'a orienté sur la méthode SPH. J'avais alors à ma disposition deux méthodes de simulation que j'allais pouvoir étudier. Afin de saisir le fonctionnement de ces deux méthodes, j'ai commencé par étudier les notes de cours et les polycopiés de certaines écoles offrant des cours sur ce sujet, notamment (ADAMS et WICKE 2009) qui est un polycopié sur la méthode SPH. Au fur et à mesure que je comprenais les méthodes et outils employés, j'ai utilisé de plus en plus de rapports techniques ou des thèses comme (MÜLLER et al. b) ou (MARIÉ 2008).

Ce mémoire est construit selon l'ordre logique où les questions me sont venues. Ainsi, j'ai commencé par m'interroger sur la description mathématique du mouvement d'un fluide et les équations qui le régissent, c'est-à-dire les équations de Navier-Stokes et l'équation de Boltzmann. Une fois ces éléments en mains, j'ai voulu répondre à la question de la résolution de ces équations. J'ai donc étudié trois méthodes différentes permettant cela. Finalement, la dernière partie de ce mémoire est la suite logique de la partie 2. C'est-à-dire, une fois que l'on a un moyen d'approcher numériquement la solution des équations, comment faire pour accélérer ce calcul ? En effet, on se rend très vite compte que résoudre ces équations est une opération très gourmande en ressources.

INTRODUCTION À LA DESCRIPTION D'UN FLUIDE

1

Un fluide est un corps dont l'état physique est liquide ou gazeux. Il ne possède pas de forme propre. Un fluide, au repos, épouse la forme du récipient qui le contient. Lorsque le fluide n'est pas au repos, il est possible de décrire mathématiquement son mouvement. Ce chapitre vise à introduire les deux types de descriptions possibles pour un fluide : la description *Eulérienne* et la description *Lagrangienne*. Par la suite, nous présenterons les équations de Navier-Stokes et l'équation de Boltzmann, outils mathématiques définissant ce mouvement.

Pour décrire le mouvement d'un fluide, quelque soit la description choisie, il est nécessaire d'avoir une échelle. L'échelle utilisée en mécanique des fluides est l'échelle *mésoscopique*. Située entre l'échelle microscopique et l'échelle macroscopique, elle permet de considérer des particules de fluides comme ponctuelles et le milieu comme continu. On ne fait donc pas une étude de toutes les molécules. On appelle une *particule de fluide* un groupe de molécules de fluide. Ainsi, il est possible d'associer à chaque particule de fluide des grandeurs comme une masse, une vitesse ou encore une pression. Ces grandeurs pouvant varier au cours du temps.

1.1 DEUX APPROCHES BIEN DIFFÉRENTES

1.1.1 Lagrangienne

La description Lagrangienne consiste à suivre le mouvement de chaque particule de fluide à chaque instant à partir de l'instant initial. Les particules de fluide sont donc individualisées. Soit une particule de fluide qui en t_0 est à la position $M_0 = (x_0, y_0, z_0)$, alors sa position à un instant t est donné par trois fonctions paramétriques :

$$\begin{cases} x &= f(M_0, t) \\ y &= g(M_0, t) \\ z &= h(M_0, t) \end{cases}$$

La vitesse de cette particule est alors :

$$\mathbf{u}(M_0, t) = \begin{pmatrix} \frac{\partial f}{\partial t} \\ \frac{\partial g}{\partial t} \\ \frac{\partial h}{\partial t} \end{pmatrix}$$



FIGURE 1.1 – Visualisation d'une trajectoire (Guy PATRY — LAL 1.3 — <http://www.freemages.fr/browse/photo-1457-voie-berge-nuit.html>)

La description Lagrangienne permet de voir naturellement la trajectoire d'une particule de fluide, c'est-à-dire le chemin suivi par une particule de fluide. Expérimentalement, les trajectoires sont visibles en mettant du colorant dans le fluide puis en prenant une photographie avec un temps de pause relativement long. La figure 1.1 illustre la description Lagrangienne d'un fluide dans le sens où chaque particule de fluide est assimilée à un véhicule se déplaçant sur la route. La photo prise avec un temps de pause long illustre ainsi la notion de trajectoire.

1.1.2 Eulérienne

La description Eulérienne ne s'intéresse pas au suivi des particules de fluide prises individuellement au cours du temps, mais au mouvement de l'ensemble des particules de fluides à un moment donné. Schématiquement, l'observateur est placé un point fixe O et regarde les particules de fluides devant lui. Ainsi, à deux instants différents, $\overrightarrow{OM}(t)$ ne représente pas la même particule de fluide. La description Eulérienne décrit le champ de vitesse à un instant t du fluide en tout point de celui-ci.

La description Eulérienne est par exemple utilisée pour décrire la carte des vents de la météo 1.2.

Ainsi, en tout point M du fluide et pour tout instant t est défini le vecteur vitesse $\mathbf{u}$ qui a pour composantes :

$$\begin{cases} u(x, y, z, t) \\ v(x, y, z, t) \\ w(x, y, z, t) \end{cases}$$

Les approches Eulérienne et Lagrangienne, bien que différentes au premier abord, sont équivalentes. Il est possible de passer de l'une à l'autre et vice-versa. Le choix d'une description par rapport à l'autre se fait en

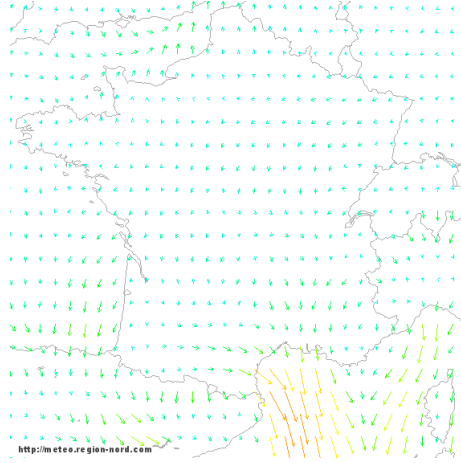


FIGURE 1.2 – Description Eulerienne du vent en France

fonction du milieu étudié, de ce que l'on souhaite représenter. La représentation Lagrangienne, bien que sans doute plus intuitive, présente le défaut majeur qu'il est souvent difficile de suivre le déplacement d'une particule de fluide. Si l'on voulait faire un petit comparatif *technique* entre une simulation Eulerienne et une simulation Lagrangienne, le tableau ci-dessous devrait convenir :

Simulation Eulerienne	Simulation Lagrangienne
Discrétisation de l'espace	Discrétisation du fluide
Maillage requis	Maillage optionnel
Conditions aux limites difficiles	Conditions aux limites simples

1.2 LA DÉRIVÉE PARTICULAIRE

Soit $A(x, y, z, t)$ une grandeur liée à une particule de fluide située en $M(x, y, z)$ à l'instant t . On s'intéresse alors au taux de variation de cette grandeur par rapport au temps que l'on note $\frac{DA}{Dt}$.

$$\begin{aligned} \frac{DA}{Dt} &= \lim_{dt \rightarrow 0} \frac{A(x + u_x dt, y + u_y dt, z + u_z dt, t + dt) - A(x, y, z, t)}{dt} \\ &= \lim_{dt \rightarrow 0} \frac{A(x, y, z, t) + u_x \frac{\partial A}{\partial x} dt + u_y \frac{\partial A}{\partial y} dt + u_z \frac{\partial A}{\partial z} dt - A(x, y, z, t)}{dt} \end{aligned}$$

D'où

$$\frac{DA}{Dt} = \frac{\partial A}{\partial t} + (\mathbf{u} \cdot \vec{\nabla})A \quad (1.1)$$

On appelle $\frac{DA}{Dt}$ la dérivée particulière (en description Lagrangienne) de A par rapport à t . $\frac{\partial A}{\partial t}$ est la dérivée partielle, au sens classique, (en description Eulerienne) de A par rapport à t . La relation 1.1 traduit le fait qu'en description Eulerienne à deux instants différents, $A(x, y, z, t)$ ne caractérise pas la même particule de fluide. Le terme $(\mathbf{u} \cdot \vec{\nabla})A$ est lié à la variation spatiale et s'appelle *terme convectif*. Le terme $\frac{\partial A}{\partial t}$ correspond à la variation de A des particules de fluides passant au même endroit.

Ainsi, si on s'intéresse à l'accélération d'une particule de fluide on aura :

$$\vec{a} = \frac{D\mathbf{u}}{Dt} = \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \vec{\nabla})\mathbf{u} \quad (1.2)$$

$\frac{D\mathbf{u}}{Dt}$ est donc l'accélération de la particule de fluide étudiée (comme en mécanique du point) et $\frac{\partial \mathbf{u}}{\partial t}$ est la variation des vitesses des *différentes* particules de fluide passant au point observé.

1.3 RENDRE COMPTE DU MOUVEMENT D'UN FLUIDE

Traduire en termes mathématiques le mouvement observé d'un fluide est le but de cette section. Deux approches seront vues :

1. les équations de Navier - Stokes pour les fluides incompressibles ;
2. l'équation continue de Boltzmann.

Ces deux moyens théoriques que nous aurons ensuite à notre disposition seront utilisés dans le chapitre suivant pour simuler numériquement le mouvement d'un fluide.

1.3.1 Équations de Navier - Stokes pour les fluides incompressibles

Les équations de Navier - Stokes sont des équations aux dérivées partielles qui décrivent le mouvement des fluides Newtoniens.

Définition 1.1 *On appelle fluide newtonien un fluide dont le coefficient de viscosité ne varie pas avec la vitesse.*

L'eau ou l'huile sont par exemple des fluides Newtoniens, alors que le dentifrice non. Le sable mouillé n'est également pas un fluide Newtonien, puisque lorsqu'on frappe le sable avec une vitesse élevée, il se comporte comme un solide (viscosité élevée) alors que si l'on appuie doucement, il se comporte comme une pâte.

Définition 1.2 *Un fluide est dit incompressible lorsque son volume reste constant indépendamment des forces externes pouvant lui être appliqué.*

Mathématiquement, cela signifie que la masse volumique est constante :

$$\frac{D\rho}{Dt} = 0$$

L'étude d'un fluide incompressible est plus simple que l'étude d'un fluide compressible, notamment puisque cela permet de se dispenser de l'équation de la conservation de l'énergie du système pour obtenir la pression et la vitesse (ce qui nous intéresse au final dans cette étude), et aussi de ne pas tenir compte des tenseurs de contraintes qui interviendrait sinon dans les équations. Il reste donc deux équations au lieu de trois.

Un fluide Newtonien incompressible peut être décrit par les équations de Navier - Stokes :

$$\frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot (\rho \mathbf{u}) = 0 \quad (1.3a)$$

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \vec{\nabla})\mathbf{u} \right) = -\vec{\nabla} P + \mu \nabla^2 \mathbf{u} + \mathbf{F} \quad (1.3b)$$

L'équation 1.3a est appelée *équation de continuité* et l'équation 1.3b est appelée *équation de bilan de la quantité de mouvement*. Sachant que le fluide est incompressible, et donc que ρ est une constante pour le problème étudié, il est possible de simplifier la première équation. Ce qui donne :

$$\vec{\nabla} \cdot \mathbf{u} = 0 \quad (1.4)$$

Cette relation est appelée *équation d'incompressibilité*.

Interprétation des équations 1.3b et 1.4

Équation de continuité Aussi appelée *équation de conservation de la masse*, elle traduit la célèbre loi d'Antoine Lavoisier :

« Rien ne se perd, rien ne se crée, tout se transforme »

Montrons rapidement comment l'obtenir.

On considère une partie de fluide de masse volumique ρ délimitée par une surface S et de volume V . La masse de cette partie de fluide est naturellement $m = \iiint_V \rho \, dV$ et le débit massique sortant de la surface S est égal à $\iint_S \rho \mathbf{u} \cdot \vec{S}$.

Ainsi, la conservation de la masse s'écrit, si l'on considère qu'il n'y a ni puits, ni source :

$$-\iint_S \rho \mathbf{u} \cdot \vec{S} = \iiint_V \frac{\partial \rho}{\partial t} \, dV$$

Le théorème d'Ostrogradsky, affirme que $\iint_S \rho \mathbf{u} \cdot \vec{S} = \iiint_V \vec{\nabla} \cdot (\rho \mathbf{u}) \, dV$. D'où

$$\iiint_V \left(\vec{\nabla} \cdot (\rho \mathbf{u}) + \frac{\partial \rho}{\partial t} \right) \, dV = 0$$

Cette propriété devant être vérifiée pour tout V , on en déduit l'équation de continuité 1.3a.

Dans le cas où ρ est constante, on aboutit très simplement à l'équation d'incompressibilité $\vec{\nabla} \cdot \mathbf{u} = 0$.

Équation de bilan de la quantité de mouvement Cette équation traduit quand à elle la seconde loi de Newton :

$$m\vec{a} = \sum \vec{F}$$

Ainsi, le terme de droite de l'équation fait intervenir les forces qui s'appliquent à une particule de fluide considérée :

- La force de pression : $-\vec{\nabla} P$.
- La force de viscosité : $\mu \nabla^2 \mathbf{u}$. Ce qui correspond à la résistance au fluide à être déformé.
- Les autres forces extérieures : $\mathbf{F}$. Telle que la force de poids, etc.

1.3.2 L'équation continue de Boltzmann

Considérons un domaine $\mathcal{R}^3$ dans lequel sont confinées N particules de fluide. Ces particules, toutes identiques, voyagent à vitesse constante entre les collisions. Elles rebondissent élastiquement entre elles et contre les parois du domaine. Notons $f(\mathbf{r}, \mathbf{u}, t)$ la fonction de distribution de ces particules. Autrement dit,

$$dN = f(\mathbf{r}, \mathbf{u}, t) d\mathbf{r} d\mathbf{u}$$

représente le nombre de particules situées à l'instant t dans un domaine $d\mathbf{r}$ autour du point $\mathbf{r}$ ayant une vitesse $\mathbf{u}$ définie à $d\mathbf{u}$ près.

Si l'on suppose qu'il n'y a pas de collision, on a :

$$f(\mathbf{r} + \mathbf{u} dt, \mathbf{u} + \frac{\mathbf{F}}{m} dt, t + dt) d\mathbf{r} d\mathbf{u} - f(\mathbf{r}, \mathbf{u}, t) d\mathbf{r} d\mathbf{u} = 0$$

Cette équation traduit le fait que le nombre de particules autour du point $\mathbf{r} + \mathbf{u} dt$ ayant une vitesse $\mathbf{u} + \frac{\mathbf{F}}{m} dt$ est égal au nombre de particules qu'il y avait autour du point $\mathbf{r}$ ayant une vitesse $\mathbf{u}$ à l'instant t .

Si l'on suppose maintenant qu'il y a des collisions, cette égalité n'est plus vraie. Il faut traduire le nombre de particules perdues et gagnées. Pour cela on introduit un opérateur de collision $\Omega(f)$ qui exprime cet échange, nous avons :

$$f(\mathbf{r} + \mathbf{u} dt, \mathbf{u} + \frac{\mathbf{F}}{m} dt, t + dt) d\mathbf{r} d\mathbf{u} - f(\mathbf{r}, \mathbf{u}, t) d\mathbf{r} d\mathbf{u} = \Omega(f) d\mathbf{u} d\mathbf{r} dt$$

On divisant le tout par $d\mathbf{r} d\mathbf{u} dt$ et faisant tendre dt vers 0, nous obtenons l'équation de Boltzmann :

$$\frac{\partial f}{\partial t} + \mathbf{u} \frac{\partial f}{\partial \mathbf{r}} + \frac{\mathbf{F}}{m} \frac{\partial f}{\partial \mathbf{u}} = \Omega(f) \quad (1.5)$$

Le membre de gauche de l'équation 1.5 représente « l'avancée du flux » et le membre de droite est ce que l'on appelle l'opérateur de collision. L'opérateur de collision proposé par Boltzmann ne fonctionnait que pour des gaz peu denses (MARIÉ 2008) et était extrêmement compliqué puisqu'il décrivait les collisions de toutes les particules. Il faudra attendre 1954 pour que des mathématiciens donnent un opérateur de collision simple et qui puisse s'appliquer à la mécanique des fluides. Cet opérateur porte le nom d'opérateur BGK, du nom de ces créateurs Bhatnagar, Gross et Krook.

L'opérateur BGK est le suivant :

$$\Omega(f) = \frac{1}{\lambda} (f^{eq} - f) \quad (1.6)$$

Cet opérateur traduit le fait que f va converger vers f^{eq} qui est la fonction de distribution des particules à l'équilibre¹. Il décrit l'effet global des collisions. λ est appelé *temps caractéristique* et dépend de la viscosité du fluide.

Ainsi, f décrit le mouvement des particules de fluide naviguant dans l'espace $\mathcal{R}^3$. Dans le chapitre suivant, nous allons approximer cette fonction afin de rendre compte du mouvement des particules de fluide.

1. Par *équilibre*, il ne faut pas comprendre que $\mathbf{u}$ est nulle, mais que la fonction de distribution f ne change pas si elle a cette valeur. f^{eq} est détaillée en annexe A.1.

1.4 BILAN

Décrire le mouvement d'un fluide est théoriquement possible. En revanche, afin de conserver des équations simples, nous avons dû faire plusieurs hypothèses sur les fluides étudiés (milieu continu, fluide newtonien et incompressible, etc). On s'est donc limité à la description d'une classe restreinte de fluides. Cette classe de fluide est néanmoins suffisamment riche pour que l'on puisse s'y intéresser, puisqu'elle comporte notamment des fluides comme l'eau ou l'huile.

Le chapitre suivant est consacré à la simulation numérique d'un fluide, c'est à dire, compte tenu des équations données ici, comment approcher une solution rendant compte du mouvement d'un fluide.

MODÉLISATION D'UNE VAGUE

2

Intéressons nous maintenant à la modélisation proprement dite. Ce chapitre, sans aucun doute le plus long des trois, présentera trois méthodes différentes utilisées pour simuler le mouvement d'un fluide.

2.1 LES ÉQUATIONS DE SAINT-VENANT

Les équations de Saint-Venant, dites les équations d'eaux peu profondes sont des équations dérivées¹ des équations de Navier-Stokes dans le cas où la profondeur de l'eau est négligeable face à la largeur du domaine étudié. Le problème est alors simplifié puisque les équations ne font plus qu'intervenir la hauteur et la vitesse de l'eau. Les équations de Saint Venant se trouvent sous deux formes : conservatives ou non-conservatives. Je ne me suis intéressé qu'au cas conservatif, ce qui permet de faire abstraction de la force de Coriolis par exemple.

On a donc dans le cas 2D :

$$\begin{cases} \frac{\partial h}{\partial t} + \frac{\partial hu}{\partial x} + \frac{\partial hv}{\partial y} = 0 \\ \frac{\partial hu}{\partial t} + \frac{\partial hu^2 + \frac{1}{2}gh^2}{\partial x} + \frac{\partial huv}{\partial y} = 0 \\ \frac{\partial hv}{\partial t} + \frac{\partial huv}{\partial x} + \frac{\partial hv^2 + \frac{1}{2}gh^2}{\partial y} = 0 \end{cases} \quad (2.1)$$

où :

- h est la hauteur de l'eau par rapport au sol
- u est la vitesse selon l'axe des abscisses ;
- v est la vitesse selon l'axe des ordonnées ;
- g est l'accélération liée à la gravité.

Pour plus de clarté, posons (BURKARDT 2010) :

$$U = \begin{pmatrix} h \\ hu \\ hv \end{pmatrix} \quad F(U) = \begin{pmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \\ huv \end{pmatrix}$$

$$G(U) = \begin{pmatrix} hv \\ huv \\ hv^2 + \frac{1}{2}gh^2 \end{pmatrix}$$

Les équations précédentes peuvent alors être ramenées à :

$$\frac{\partial U}{\partial t} + \frac{\partial F(U)}{\partial x} + \frac{\partial G(U)}{\partial y} = 0 \quad (2.2)$$

1. Le passage des équations de Navier-Stokes aux équations de Saint-Venant est donné en annexe A de (MÜLLER et al. a), et admis dans le cadre de ce mémoire.

Cette équation est régie par une loi de conservation hyperbolique, dont la solution peut être approchée grâce la méthode de Lax-Wendroff, basée sur les différences finies.

Dans la suite de cette section, nous nous intéresserons seulement au cas 1D de ces équations afin de ne pas alourdir les écritures, la méthode présentée restant valable dans le cas 2D.

Dans le cas 1D, l'équation 2.2 se ramène à :

$$\frac{\partial U}{\partial t} + \frac{\partial F(U)}{\partial x} = 0 \quad (2.3)$$

avec

$$U = \begin{pmatrix} h \\ hu \end{pmatrix} \quad F(U) = \begin{pmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \end{pmatrix}$$

2.1.1 La méthode de Lax-Wendroff

Soit une équation du type :

$$\frac{\partial f(x,t)}{\partial t} = \frac{\partial g(f(x,t))}{\partial x}$$

dont on souhaite approcher la solution. On se donne alors une discrétisation en temps et en espace. On note f_i^n l'approximation de $f(i\Delta x, n\Delta t)$, et g_i^n celle de $g(f_i^n)$. On suppose connues les conditions initiales.

L'idée de la méthode de Lax-Wendroff est de calculer une estimation de $f_{i+\frac{1}{2}}^{n+\frac{1}{2}}$ qui va servir à calculer chaque valeur de f_i^{n+1} .

La méthode de Lax-Wendroff donne deux relations :

$$\frac{f_{i+\frac{1}{2}}^{n+\frac{1}{2}} - \frac{f_i^n + f_{i+1}^n}{2}}{\frac{1}{2}\Delta t} = \frac{g_{i+1}^n - g_i^n}{\Delta x} \quad (2.4a)$$

$$\frac{f_i^{n+1} - f_i^n}{\Delta t} = \frac{g_{i+\frac{1}{2}}^{n+\frac{1}{2}} - g_{i-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x} \quad (2.4b)$$

De 2.4a on peut tirer l'estimation intermédiaire :

$$f_{i+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{f_i^n + f_{i+1}^n}{2} + \frac{\Delta t}{2\Delta x} (g_{i+1}^n - g_i^n) \quad (2.5)$$

Puis de 2.4b, on peut sortir la valeur de f_i^{n+1} :

$$f_i^{n+1} = f_i^n + \frac{\Delta t}{\Delta x} \left(g_{i+\frac{1}{2}}^{n+\frac{1}{2}} - g_{i-\frac{1}{2}}^{n+\frac{1}{2}} \right) \quad (2.6)$$

Comme on connaît $f_{i+\frac{1}{2}}^{n+\frac{1}{2}}$ et $f_{i-\frac{1}{2}}^{n+\frac{1}{2}}$ et que l'on a $g_i^n = g(f_i^n)$, on connaît $g_{i+\frac{1}{2}}^{n+\frac{1}{2}}$ et $g_{i-\frac{1}{2}}^{n+\frac{1}{2}}$, ce qui rend le calcul possible.

Cette approximation est en $O(\Delta t^2, \Delta x^2)$.

2.1.2 Application de la méthode Lax-Wendroff aux équations de Saint-Venant dans le cas 1D

Nous avons :

$$\frac{\partial U}{\partial t} = -\frac{\partial F(U)}{\partial x} \quad (2.7)$$

L'idée est d'exprimer le vecteur $F(U)$ en fonction des composantes de U uniquement. Ainsi, la quantité hu^2 que l'on retrouve dans le vecteur $F(U)$ sera exprimée comme $\frac{(hu)^2}{h}$, autrement dit, la deuxième composante de U au carré sur la première composantes de U , ce qui permet d'avoir la quantité hu^2 , à l'aide des composantes de U uniquement. On fera donc l'hypothèse que la quantité h n'est pas nulle, pour tout x et tout t .

On a, d'après la relation 2.5 :

$$\begin{aligned} U_{i+\frac{1}{2}}^{n+\frac{1}{2}} &= \frac{U_i^n + U_{i+1}^n}{2} - \frac{\Delta t}{2\Delta x} (F_{i+1}^n - F_i^n) \\ \begin{pmatrix} h_{i+\frac{1}{2}}^{n+\frac{1}{2}} \\ (hu)_{i+\frac{1}{2}}^{n+\frac{1}{2}} \end{pmatrix} &= \frac{1}{2} \begin{pmatrix} h_i^n + h_{i+1}^n \\ (hu)_i^n + (hu)_{i+1}^n \end{pmatrix} \\ &\quad - \frac{\Delta t}{2\Delta x} \left(\begin{pmatrix} (hu)_{i+1}^n - (hu)_i^n \\ \left[\frac{((hu)_{i+1}^n)^2}{h_{i+1}^n} + \frac{1}{2}gh_{i+1}^n \right] - \left[\frac{((hu)_i^n)^2}{h_i^n} + \frac{1}{2}gh_i^n \right] \end{pmatrix} \right) \end{aligned}$$

Puis, à l'aide de la relation 2.6, on en déduit la valeur de U_i^{n+1} , soit :

$$\begin{aligned} \begin{pmatrix} h_i^{n+1} \\ (hu)_i^{n+1} \end{pmatrix} &= \begin{pmatrix} h_i^n \\ (hu)_i^n \end{pmatrix} \\ &\quad - \frac{\Delta t}{\Delta x} \left(\begin{pmatrix} (hu)_{i+\frac{1}{2}}^{n+\frac{1}{2}} - (hu)_{i-\frac{1}{2}}^{n+\frac{1}{2}} \\ \left[\frac{((hu)_{i+\frac{1}{2}}^{n+\frac{1}{2}})^2}{h_{i+\frac{1}{2}}^{n+\frac{1}{2}}} + \frac{1}{2}gh_{i+\frac{1}{2}}^{n+\frac{1}{2}} \right] - \left[\frac{((hu)_{i-\frac{1}{2}}^{n+\frac{1}{2}})^2}{h_{i-\frac{1}{2}}^{n+\frac{1}{2}}} + \frac{1}{2}gh_{i-\frac{1}{2}}^{n+\frac{1}{2}} \right] \end{pmatrix} \right) \end{aligned}$$

Ainsi, compte tenu du fait que l'on connaisse les conditions initiales h_i^0 et u_i^0 , on peut calculer pour n'importe quel instant la quantité h_i^n .

Pour que la modélisation soit complète il faut expliciter les conditions aux limites. La méthode de Lax-Wendroff permet de calculer les valeurs de U seulement pour les points intérieurs au domaine étudié, pas les points limites. Il est par exemple impossible de calculer h_0^1 , puisqu'il faudrait connaître $h_{-1/2}^{1+1/2}$, qui n'existe pas. Plusieurs choix sont alors possibles. Le choix des conditions aux limites est fait en fonction du problème physique étudié.

Nous ferons la distinction entre deux possibilités :

Réflexif c'est à dire que la vague va se heurter à un mur et repartir dans l'autre sens. Mathématiquement cela se traduit par :

$$\begin{aligned} u_0^n &= -u_1^n & h_0^n &= h_1^n \\ u_I^n &= -u_{I-1}^n & h_I^n &= h_{I-1}^n \end{aligned}$$

où I représente le dernier nœud de la discrétisation.

Libre c'est à dire que la vague va continuer dans sa direction sans rencontrer d'obstacle aux limites du domaine étudié. C'est le choix fait pour simuler une vague dans un océan par exemple. Mathématiquement, on a :

$$\begin{aligned} u_0^n &= u_1^n & \left| \right. & h_0^n = h_1^n \\ u_I^n &= u_{I-1}^n & \left| \right. & h_I^n = h_{I-1}^n \end{aligned}$$

Étant donné la forme de la solution $h = f(x, t)$ dans le cas 1D et $h = f(x, y, t)$ dans le cas 2D, il est impossible de voir des éclaboussures, de l'écume ou simplement un rouleau se former. Ce modèle est donc adapté dans le cas d'une simulation présentant peu ou pas de perturbations. On peut voir une vague se former puis avancer, mais pas l'éclatement de la vague. Afin de voir réellement l'eau *éclater*, les gouttes d'eau, de voir réellement le rouleau, nous allons maintenant étudier deux méthodes particulières.

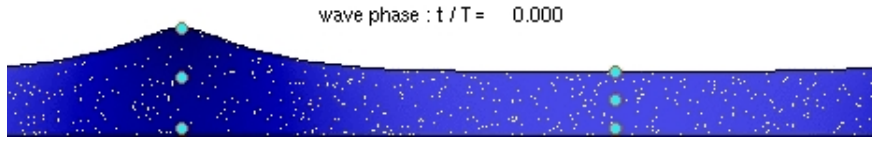


FIGURE 2.1 – Shallow water 1D (BURKARDT 2010)

2.2 LA MÉTHODE DE BOLTZMANN SUR RÉSEAU

La méthode de Boltzmann sur réseau (ou *Lattice Boltzmann Method* en anglais, que l'on abrègera LBM) est une méthode qui utilise un maillage où chaque cellule va interagir avec — et seulement avec — ses cellules voisines. Cette méthode, au lieu de discrétiser les équations de Navier-Stokes, discrétise l'équation de Boltzmann.

2.2.1 Discrétisation de l'équation de Boltzmann

On rappelle que l'équation de Boltzmann, avec l'opérateur BGK est la suivante :

$$\frac{\partial f}{\partial t} + \mathbf{u} \frac{\partial f}{\partial \mathbf{r}} + \frac{\mathbf{F}}{m} \frac{\partial f}{\partial \mathbf{u}} = \frac{1}{\lambda} (f^{eq} - f)$$

Par la suite, nous considérerons que le système n'est soumis à aucune force extérieure. Ainsi, l'équation se simplifie et on obtient :

$$\frac{\partial f}{\partial t} + \mathbf{u} \frac{\partial f}{\partial \mathbf{r}} = \frac{Df}{Dt} = \frac{1}{\lambda} (f^{eq} - f)$$

Dans le modèle LBM, les vitesses sont discrètes. Il est courant de noter α la dimension de l'espace dans lequel on travaille et β le nombre de direction des vitesses discrètes. On appelle alors $D_\alpha Q_\beta$ le maillage associé.

On travaille fréquemment avec un maillage D_2Q_9 en deux dimensions et D_3Q_{19} en trois dimensions. Pour faire un rendu en temps réel en deux dimensions, certains auteurs utilisent un maillage D_2Q_4 pour accélérer les calculs. En trois dimensions, le maillage D_3Q_{19} semble être un bon compromis entre stabilité, rapidité de calculs et consommation de mémoire (MÜLLER et al. a).

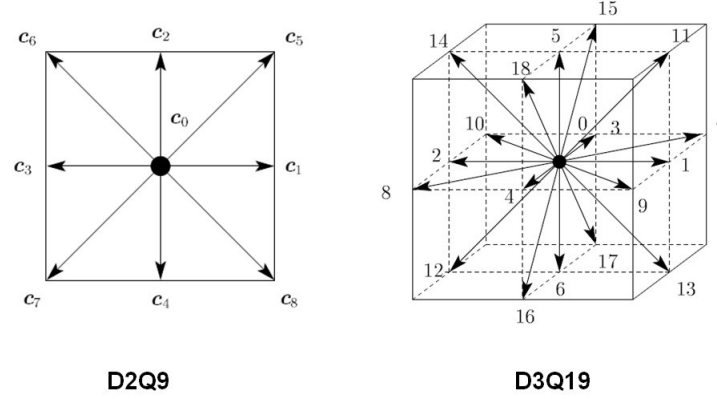


FIGURE 2.2 – Maillages couramment utilisés

La figure 2.2 illustre la discrétisation des vitesses : à gauche une maille D_2Q_9 où l'on voit les 9 vitesses possibles et à droite une maille D_3Q_{19} .

Étant donné que les vitesses sont discrètes ($\mathbf{e}_i$), on note :

$$f_i(\mathbf{r}, t) = f(\mathbf{r}, \mathbf{e}_i, t) \quad i \in [0; \beta - 1]$$

Ainsi, l'équation discrète est :

$$\underbrace{f_i(\mathbf{r} + \mathbf{e}_i \Delta t, t + \Delta t) - f_i(\mathbf{r}, t)}_{\text{Advection}} = \underbrace{\frac{\Delta t}{\lambda} (f_i^{eq}(\mathbf{r}, t) - f_i(\mathbf{r}, t))}_{\text{Collision}}$$

Par commodité, on prendra $\Delta t / \Delta x = 1$ et on notera $\omega = \frac{\Delta t}{\lambda}$. L'équation discrète peut donc s'écrire sous la forme suivante :

$$f_i(\mathbf{r} + \mathbf{e}_i \Delta t, t + \Delta t) = (1 - \omega) f_i(\mathbf{r}, t) + \omega f_i^{eq}(\mathbf{r}, t)$$

Dans la suite de ce chapitre, on s'intéressera au modèle D_3Q_{19} et D_2Q_9 . Les schémas seront donnés pour le modèle D_2Q_9 pour que cela soit plus clair.

Chaque maillage impose ses propres vitesses. Ainsi on a pour la maille D_2Q_9 :

$$\mathbf{e}_i = \begin{cases} (0, 0), & (i = 0) \\ (\pm 1, 0), (0, \pm 1), & (i = 1 \dots 4) \\ (\pm 1, \pm 1), & (i = 5 \dots 8) \end{cases}$$

et pour la maille D_3Q_{19} :

$$\mathbf{e}_i = \begin{cases} (0, 0, 0), & (i = 0) \\ (\pm 1, 0, 0), (0, \pm 1, 0), (0, 0, \pm 1), & (i = 1 \dots 6) \\ (\pm 1, \pm 1, 0), & (i = 7 \dots 10) \\ (0, \pm 1, \pm 1), & (i = 11 \dots 14) \\ (\pm 1, 0, \pm 1) & (i = 15 \dots 18) \end{cases}$$

2.2.2 Description de la méthode

La méthode est basée sur deux étapes distinctes :

1. L'avancée du fluide (ou *streaming - step*). Étape dans laquelle les quantités f_i sont mises à jour en supposant qu'il n'y a pas de collisions.
2. La collision (ou *collide - step*). Étape dans laquelle les quantités précédemment calculées sont mises à jour pour tenir compte des collisions.

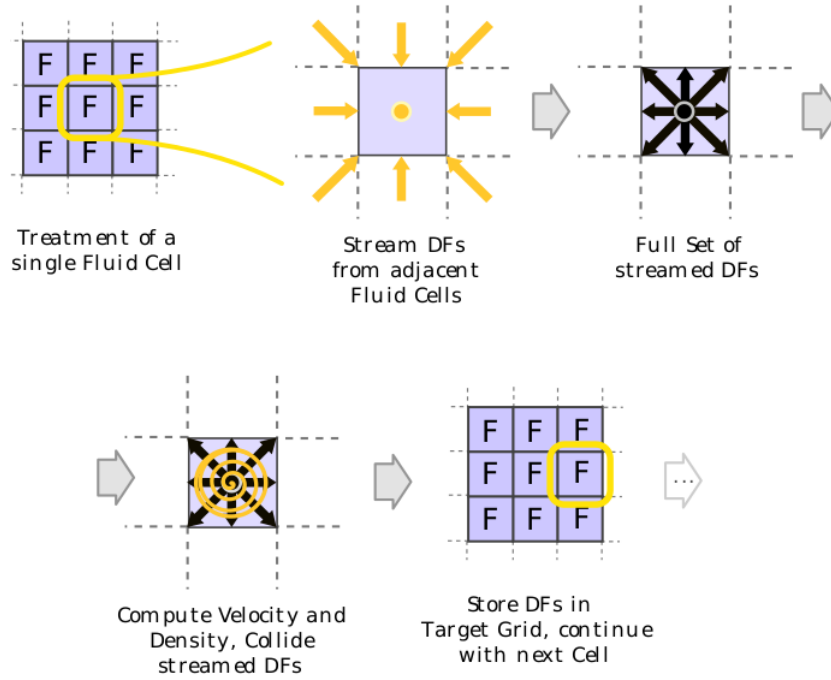


FIGURE 2.3 – Visualisation de la méthode appliquée sur une maille (MÜLLER et al. a)

La figure 2.3 est donnée pour illustrer les explications qui sont à suivre.

Chaque maille du domaine est prise une à une. L'étape 1 consiste à définir les fonctions de distribution f'_i en fonction des f_i en supposant qu'il n'y a pas eu de collisions pour la maille considérée. On a donc :

$$f'_i(\mathbf{r}, t + \Delta t) = f_i(\mathbf{r} - \mathbf{e}_i \Delta t, t) \quad \forall i$$

À l'étape 2, les collisions entrent en jeu. Pour prendre en compte les collisions, il est nécessaire de calculer² f_i^{eq} . On a :

$$f_i^{eq} = w_i \rho \left[1 + 3(\mathbf{e}_i \cdot \mathbf{u}) - \frac{3}{2} \mathbf{u}^2 + \frac{9}{2} (\mathbf{e}_i \cdot \mathbf{u})^2 \right] \quad (2.8)$$

où :

- ρ est la densité du fluide présente dans la maille considérée (qui dépend donc des f_i).
- $\mathbf{u}$ est la vitesse locale du fluide dans la maille considérée (qui dépend également des f_i).

2. L'obtention de cette formule est donnée en annexe A.1.

- w_i sont des coefficients qui sont liés au maillage. On a :
 - Pour D_3Q_{19} : $w_0 = \frac{1}{3}$, $w_{1,\dots,6} = \frac{1}{18}$, $w_{7,\dots,18} = \frac{1}{36}$.
 - Pour D_2Q_9 : $w_0 = \frac{4}{9}$, $w_{1,\dots,3} = \frac{1}{9}$, $w_{4,\dots,8} = \frac{1}{36}$.

La densité ρ et la vitesse locale du fluide $\mathbf{u}$ sont obtenues de la façon suivante :

$$\rho = \sum_{i=0}^{\beta-1} f_i \quad \mathbf{u} = \frac{1}{\rho} \sum_{i=0}^{\beta-1} \mathbf{e}_i f_i$$

Une fois ces calculs effectués, on peut calculer f_i^{eq} et mettre à jour chaque f_i grâce à l'équation discrète de Boltzmann :

$$f_i(\mathbf{r}, t + \Delta t) = (1 - \omega) f_i'(\mathbf{r}, t + \Delta t) + \omega f_i^{eq}$$

La quantité ω varie entre $]0;2]$ et est un paramètre qui permet de contrôler la viscosité du fluide. Plus ω tend vers 0 plus le fluide est visqueux et inversement, plus il tend vers 2 plus le fluide est turbulent.

Pour gérer simplement les conditions limites et les obstacles avec la méthode de Boltzmann sur réseau, on peut utiliser la condition de type *réflectif*. C'est à dire que si $\mathbf{r} + \mathbf{e}_i \Delta t$ représente un obstacle, alors on affecte la valeur de la distribution de direction opposée. C'est à dire :

$$f_i'(\mathbf{r}, t + \Delta t) = f_{invert(i)}(\mathbf{r}, t)$$

Où $invert(i)$ est une fonction qui retourne l'indice du vecteur vitesse de direction opposée au vecteur $\mathbf{e}_i$. Il s'agit d'une rotation de π radians.

2.2.3 Un pas vers la modélisation d'un fluide à surface libre

Le plus souvent, on s'intéresse à la modélisation d'un fluide à surface libre, c'est-à-dire qui possède une partie en contact avec un gaz, comme de l'air ou autre. Pour le moment, ce que nous avons vu avec LBM ne permet pas de le faire : nous avons montré comment modéliser le fluide au sein du fluide. Nous allons donc voir maintenant l'idée qu'il y a derrière la modélisation d'un fluide à surface libre avec LBM.

Dans la méthode jusque là décrite, chaque maille était soit un obstacle, soit du fluide. Nous allons maintenant ajouter deux autres types (THÜREY 2007) : *gaz* et *interface*. Les mailles de type *gaz* seront les mailles sans fluide (rien ne sera donc calculé ici) et les mailles de type *interface* seront celles qui contiennent à la fois du fluide et du gaz. L'utilisation de cette interface permet de définir un contour fermé autour du fluide et de s'assurer, via un calcul, que l'équation de continuité est bien respectée.

Chaque cellule possède une masse, en fonction de son type :

$$M(\mathbf{r}, t) = \begin{cases} 0 & \text{Si de type } gaz \\ \rho(\mathbf{r}, t) & \text{Si de type } fluide \\ \rho(\mathbf{r}, t)\epsilon(\mathbf{r}, t) & \text{Si de type } interface \end{cases}$$

où $\epsilon(\mathbf{r}, t)$ représente la quantité de fluide dans la cellule considérée (avec $0 < \epsilon < 1$).

Les cellules de type *fluide* et *gaz* ne peuvent se transformer qu'en cellule de type *interface*. Alors que les cellules d'*interface* peuvent devenir des

cellules *gaz* ou *fluide*. Durant les collisions, chaque cellule gagne ou perd de la masse qu'elle échange avec ses cellules voisines. Il faut donc, au cours de la simulation, faire des changements. Une cellule de type *fluide* va devenir de type *interface* lorsqu'au moins une de ces cellules voisines est de type *gaz*. Une cellule de type *interface* deviendra de type *gaz* si sa masse M est négative ou nulle, ou bien de type *fluide* si masse devient supérieure ou égale à ρ .

Si on appelle $\Delta M_i(\mathbf{r}, t)$ l'échange de masse entre une cellule de type *interface* et la cellule située à $\mathbf{r} + \mathbf{e}_i \Delta t$, on a :

$$\Delta M_i(\mathbf{r}, t + \Delta t) = \begin{cases} 0 & \text{if } \mathbf{r} + \mathbf{e}_i \Delta t \text{ is of type fluid or gas} \\ f_i(\mathbf{r} - \mathbf{e}_i \Delta t, t) - f_i(\mathbf{r}, t) & \text{if } \mathbf{r} + \mathbf{e}_i \Delta t \text{ is of type interface} \end{cases}$$

Si $\mathbf{r} + \mathbf{e}_i \Delta t$ est de type *gaz*, *fluide* ou *interface* respectivement. Puis on calcule :

$$M(\mathbf{r}, t + \Delta t) = M(\mathbf{r}, t) + \sum_{i=0}^{\beta-1} \Delta M_i(\mathbf{r}, t + \Delta t)$$

ce qui permet de faire les transformations *interface* / *gaz* / *fluide* nécessaires.

L'illustration ci-dessus schématise le raisonnement à tenir :

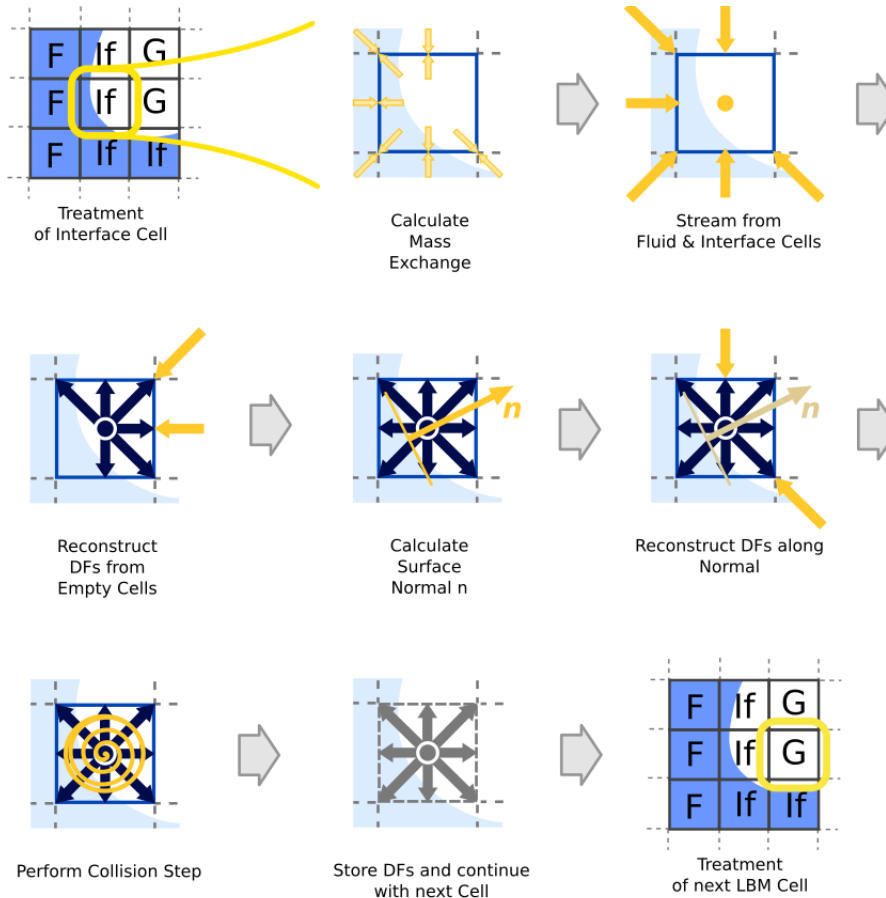


FIGURE 2.4 – Illustration des différentes étapes du processus (THÜREY 2007)

2.2.4 Exemple et conclusions sur LBM

Finissons cette technique de modélisation avec un petit exemple.

La figure 2.5 est le rendu de l'implémentation³ que j'ai faite de cette méthode, avec les forces extérieures négligées et sans surface libre. Le maillage utilisé est D_2Q_9 . Sur chaque image le fluide est représenté deux fois au même instant. Sur la gauche on visualise la densité du fluide (ce qui peut être assimilé à l'onde qui se déplace) et sur la droite la vitesse locale $\mathbf{u}$ du fluide au point considéré. Le rouge représentant une vitesse selon l'axe des abscisses et le bleu selon l'axe des ordonnées, la graduation se faisant selon l'intensité. Les contours verts sont des obstacles. La condition aux limites utilisée est de type réfléchitif.

Initialement, pour que le fluide ne bouge pas, on pose $f_i = w_i$. Ainsi, il est dans son état de stabilité. Pour démarrer l'animation, il suffit de cliquer avec la souris, ce qui permet d'ajouter un peu de fluide à l'endroit visé, et donc de lancer la simulation. Le phénomène simulé ici est une goutte de fluide ajoutée dans un fluide (de même nature) au repos. Ce qui a pour effet de créer une vague. Chaque clic ajoute une nouvelle goutte et donc perturbe l'état courant du fluide. Petit à petit, si on n'agit pas sur le système, l'état du fluide tend vers l'équilibre.

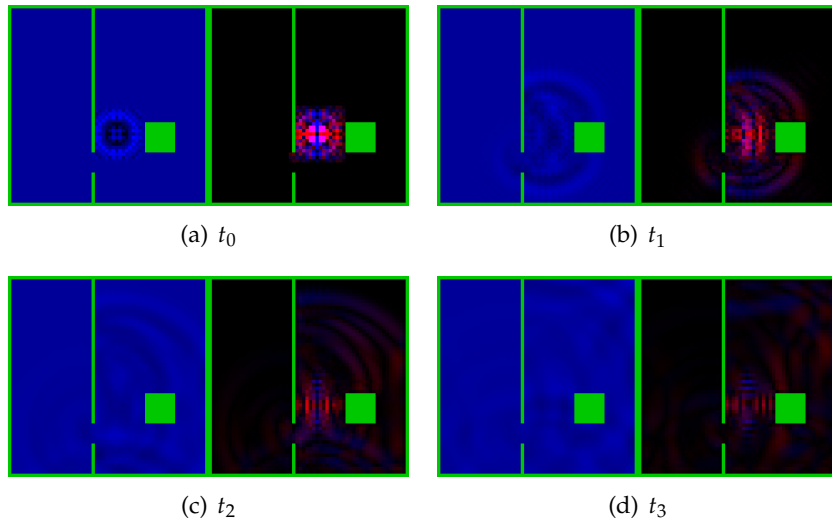


FIGURE 2.5 – LBM D_2Q_9

La méthode LBM permet de simuler avec un taux de réalisme élevé un fluide. La figure 2.5 n'est certes pas très impressionnante, mais l'implémentation de cette méthode est extrêmement simple. C'est notamment cette technique qui est utilisée par le moteur de rendu de fluide de Blender.

Nous avons, pour cette méthode, supposé qu'aucune force extérieure n'agissait sur le fluide. Le lecteur intéressé par la gestion des forces extérieures pourra se référer à [HAYASHI](#).

3. Le détail de l'implémentation est donnée en annexe [A.2](#).

2.3 SMOOTHED PARTICLE HYDRODYNAMICS

Smoothed Particle Hydrodynamics (que l'on abrègera SPH) est une méthode de calcul utilisée pour simuler des fluides à surface libre. C'est une méthode cent pour cent Lagrangienne qui a la particularité de ne pas avoir de maillage — ce qui fait qu'on s'en sert beaucoup dans les applications où la géométrie du domaine est amenée à varier. Cette méthode est sans doute plus simple à visualiser que les précédentes (bien que les outils mathématiques soient plus abstraits). En effet, ici pour simuler un fluide nous allons l'assimiler à un nombre fini de petites billes se déplaçant dans un domaine et voir comment elles réagissent entre elles. Imaginez une piscine remplie de boules de billard et vous perturbez ces billes, en jetant un obstacle dans la piscine par exemple, et vous décrivez le mouvement observé. Chacune des billes portera des informations sur son état, comme sa pression interne, sa vitesse, sa densité etc. Ces informations seront mises à jour grâce aux équations de Navier Stokes, qui doivent être vérifiées en tout point du fluide.

Dans le modèle SPH, on ne se préoccupe pas de l'équation de conservation de masse. En effet, la masse du système est naturellement conservée puisque chaque particule porte une information sur sa propre masse. De fait, si on n'ajoute pas de particule, qu'on n'en supprime pas ou que l'on ne modifie pas leur masse, la masse globale du fluide est conservée. De plus, généralement pour l'animation graphique, on se place dans l'hypothèse où le fluide est incompressible ([ADAMS et WICKE 2009](#)), l'équation de la conservation de l'énergie n'est plus nécessaire pour connaître la pression et la vitesse. Ainsi, dans la suite de ce chapitre, nous nous intéresserons simplement à la résolution de l'équation de bilan de conservation de la quantité de mouvement, qui a été présentée au chapitre 1. Soit :

$$\underbrace{\rho \frac{D \mathbf{u}}{D t}}_{\text{advection}} = \underbrace{-\vec{\nabla} P}_{\text{pression}} + \underbrace{\mu \nabla^2 \mathbf{u}}_{\text{viscosité}} + \underbrace{\mathbf{F}}_{\text{forces ext.}} \quad (2.9)$$

2.3.1 Integral and summation interpolant

Le but de cette sous-section est de montrer comment on peut approximer une fonction, son gradient, son Laplacien en un point à l'aide des points voisins. (D'où le terme *interpolant*). Ces notions seront ensuite utilisées pour approcher numériquement la solution de l'équation 2.9. Nous ferons la distinctions entre deux cas, continu et discontinu. Le premier pour présenter les résultats généraux et le second pour l'appliquer à la méthode SPH.

Cas continu

Sachant que la distribution de Dirac $\delta(x)$ est l'élément neutre du produit de convolution, on a $f * \delta = f$, soit :

$$f(x) = \int f(y) \delta(y - x) dy \quad (2.10)$$

On approxime alors la distribution de Dirac par une fonction “de poids” ou “noyau”. On a donc :

$$\langle f(x) \rangle = \int f(y) W(\|y - x\|, h) \, dy \quad (2.11)$$

où $\langle f(x) \rangle$ est l'évaluation approchée de $f(x)$ et où $W(\|x\|, h)$ la fonction de poids qui est choisie de sorte qu'elle ait les mêmes propriétés que δ . Autrement dit :

$$W(\|x\|, h) \geq 0, \quad \int W(\|x\|, h) \, dx = 1, \quad \text{supp}(W(\|x\|, h)) \subset]-h; h[$$

Toute fonction $W_h \in L^1$ ayant ces propriétés converge vers δ quand h tend vers 0 (La preuve est donnée en annexe B.1). On supposera aussi, pour les besoins de la proposition 2.1, que $W_h \in C^\infty$. Des exemples de noyaux utilisés sont donnés en annexe B.2, aussi bien pour le cas 2D que le cas 3D. Le paramètre h permet donc de contrôler l'influence des points voisins de x . Des valeurs trop grandes introduiront des calculs non-nécessaires, alors que des valeurs trop petites peuvent faire que l'on passe à coté des certains points nécessaires au calcul de l'estimation. Le choix de h est donc crucial pour la simulation.

Afin de pouvoir approcher la solution de l'équation 2.9, nous devons aussi nous intéresser à l'évaluation du gradient et du Laplacien d'une fonction f . Ces opérateurs seront appliqués directement sur $\langle f \rangle$. La proposition 2.1 nous permet de déduire une relation entre les dérivées de $\langle f \rangle$ et celles de W_h .

Proposition 2.1 Soit $f \in L^1, g \in C^\infty$. Si $\forall k = 0 \dots p, g^{(k)}$ est bornée, alors

$$(f * g)^{(k)} = f * g^{(k)}, k = 0 \dots p$$

Le gradient de $\langle f(x) \rangle$ est défini par :

$$\vec{\nabla} \langle f(x) \rangle = \sum_{i=0}^n \frac{\partial (f * W)}{\partial x_i} \mathbf{e}_i$$

En utilisant la proposition 2.1, il vient que

$$\vec{\nabla} \langle f(x) \rangle = \sum_{i=0}^n \left(f * \frac{\partial W}{\partial x_i} \right) \mathbf{e}_i$$

puis par linéarité de la convolution, $\vec{\nabla} \langle f(x) \rangle = f * \vec{\nabla} W$

Le gradient d'une fonction peut donc être calculé ainsi :

$$\vec{\nabla} \langle f(x) \rangle = \int f(y) \vec{\nabla} W(\|y - x\|, h) \, dy \quad (2.12)$$

En utilisant un raisonnement similaire, la proposition 2.1 nous permet aussi de déduire que le Laplacien de f peut être calculé comme ceci :

$$\nabla^2 \langle f(x) \rangle = \int f(y) \nabla^2 W(\|y - x\|, h) \, dy \quad (2.13)$$

De plus, on a :

$$\begin{aligned} \vec{\nabla} W(\|y - x\|, h) &= \frac{y - x}{\|y - x\|} W'(\|y - x\|, h) \\ \nabla^2 W(\|y - x\|, h) &= W''(\|y - x\|, h) \end{aligned}$$

Cas discontinu

Étant donné que l'on assimile le fluide à un nombre fini N de petites billes, nous devons discrétiser les expressions que nous avons définies ci-dessus qui sont continues. On pose :

$$\begin{aligned} f_i &= \langle f(\mathbf{r}_i) \rangle \\ W_h^{ji} &= W(\|\mathbf{r}_j - \mathbf{r}_i\|, h) \\ \Delta V_j &= \frac{m_j}{\rho_j} \quad \text{car } m_j = \Delta V_j \times \rho_j \end{aligned}$$

En appliquant la méthode des rectangles (ERLEBEN 2010), on a

$$f_i = \sum_j^N \frac{m_j}{\rho_j} f_j W_h^{ji} \quad (2.14)$$

$$\vec{\nabla} f_i = \sum_j^N \frac{m_j}{\rho_j} f_j \vec{\nabla} W_h^{ji} \quad (2.15)$$

$$\nabla^2 f_i = \sum_j^N \frac{m_j}{\rho_j} f_j \nabla^2 W_h^{ji} \quad (2.16)$$

2.3.2 Calcul de la position des particules

Nous pouvons écrire l'équation du bilan de quantité de mouvement 2.9 comme ci (ADAMS et WICKE 2009) :

$$\frac{D\mathbf{u}}{Dt} = \frac{1}{\rho} (\mathbf{f}^p + \mathbf{f}^v + \mathbf{f}^e)$$

où on a :

$$\begin{aligned} \mathbf{f}^p &= -\vec{\nabla} P && \text{La force liée à la pression} \\ \mathbf{f}^v &= \mu \nabla^2 \mathbf{u} && \text{La force liée à la viscosité} \\ \mathbf{f}^e &= \mathbf{F} && \text{Les forces extérieures.} \end{aligned}$$

Or, nous venons de voir une méthode pour évaluer une fonction f , son gradient et son Lapacien. Autrement dit, nous pouvons calculer chacune des forces ci-dessus et ainsi en déduire l'accélération. De fait, $\mathbf{u}$ peut être calculée en utilisant un schéma d'intégration numérique et de même $\mathbf{r}$. Ce qui nous permet de connaître la position de chaque particule de fluide au temps t .

Chaque particule de fluide porte des informations, comme la masse ou la vitesse, que nous devons mettre à jour à chaque itération.

La première à mettre à jour, pour chaque particule, est la densité⁴, et on a :

4. Il est important de noter que SPH est une technique de simulation originalement utiliser pour des fluides *compressibles* et l'astrophysique. Ainsi, il est nécessaire de calculer ρ_i pour ensuite utiliser l'équation de Tait 2.18 pour extraire la pression sans avoir d'équation différentielle à résoudre. Cependant, on suppose toujours que le fluide est incompressible, ce qui nous permet d'utiliser l'équation 1.3b. Cette hypothèse est faite dans de nombreux ouvrages, on ne citera que (KELAGER 2006, ADAMS et WICKE 2009). Il existe une variante de SPH, *incompressible-SPH* n'utilisant pas cette hypothèse. Mais elle ne sera pas abordée dans le cadre de cette étude.

$$\rho_i = \sum_j \frac{m_j}{\rho_j} \rho_j W_h^{ji} = \sum_j m_j W_h^{ji} \quad (2.17)$$

Lorsque celle-ci est calculée pour chaque particule de fluide, on peut en déduire grâce à l'équation de Tait, la pression au sein de chaque particule. On a :

$$P_i = K \left(\left(\frac{\rho_i}{\rho_0} \right)^\gamma - 1 \right) \quad (2.18)$$

où K et γ détermine la compressibilité du fluide. On a $K = \frac{c_{s0}^2 \rho_0}{\gamma}$, où c_{s0} est la vitesse du son. Généralement, $\gamma = 1$ est choisi pour l'animation graphique (ADAMS et WICKE 2009), on trouve donc souvent l'équation 2.18 sous la forme :

$$P_i = c_{s0}^2 (\rho_i - \rho_0) \quad (2.19)$$

Le problème de symétrie

Si l'on prend la formule 2.15 pour calculer le gradient de la pression, on remarque que l'on n'aura pas nécessairement de symétrie (principe d'action-réaction). Pour voir cela, il suffit de considérer deux particules. Comme le gradient du noyau est nul en 0, seule la particule 2 contribuera au calcul de la force de pression de la particule 1 et inversement (MÜLLER et al. b). Pour palier ce problème, on utilise ce que l'on appelle l'*approximation symétrique* du gradient.

On cherche à calculer $\frac{\vec{\nabla} P}{\rho}$. On peut remarquer l'égalité suivante

$$\vec{\nabla} \left(\frac{P}{\rho} \right) = \frac{\vec{\nabla} P}{\rho} - \frac{P}{\rho^2} \vec{\nabla} \rho$$

D'où l'on tire⁵ :

$$\frac{\vec{\nabla} P}{\rho} = \vec{\nabla} \left(\frac{P}{\rho} \right) + \frac{P}{\rho^2} \vec{\nabla} \rho$$

On en déduit alors que :

$$\begin{aligned} \frac{\vec{\nabla} P_i}{\rho_i} &= \left(\sum_j \frac{m_j}{\rho_j} \frac{P_j}{\rho_j} \vec{\nabla} W_h^{ji} \right) + \left(\frac{P_i}{\rho_i^2} \sum_j \frac{m_j}{\rho_j} \rho_j \vec{\nabla} W_h^{ji} \right) \\ &= \sum_j m_j \left(\frac{P_j}{\rho_j^2} + \frac{P_i}{\rho_i^2} \right) \vec{\nabla} W_h^{ji} \end{aligned}$$

On a donc :

$$\frac{\mathbf{f}_i^p}{\rho_i} = - \sum_j m_j \left(\frac{P_j}{\rho_j^2} + \frac{P_i}{\rho_i^2} \right) \vec{\nabla} W_h^{ji} \quad (2.20)$$

On cherche maintenant à calculer l'accélération liée à la viscosité $\mathbf{f}_i^v = \mu \nabla^2 \mathbf{u}_i$. Par un raisonnement similaire, on constate que si l'on utilise la formule 2.16, on aboutit à une force asymétrique.

5. C'est l'une des *golden rules* de SPH; mettre la densité dans les opérateurs gradients et Laplacien : http://www.cse-lab.ethz.ch/images/golden_rules_and_derivatives_of_SPH_notes.pdf

On peut remarquer que

$$\nabla^2(\rho \mathbf{u}) = \rho \nabla^2 \mathbf{u} + \mathbf{u} \nabla^2 \rho + 2 \vec{\nabla} \mathbf{u} \cdot \vec{\nabla} \rho$$

d'où l'on tire

$$\nabla^2 \mathbf{u} = \frac{1}{\rho} (\nabla^2(\rho \mathbf{u}) - \mathbf{u} \nabla^2 \rho)$$

puisque que $\vec{\nabla} \mathbf{u} = 0$ d'après l'équation d'incompressibilité 1.4.

On en déduit alors que :

$$\nabla^2 \mathbf{u}_i = \sum_j \frac{m_j}{\rho_i} (\mathbf{u}_j - \mathbf{u}_i) \nabla^2 W_h^{ji}$$

L'accélération due à la viscosité est donc :

$$\frac{\mathbf{f}_i^v}{\rho_i} = \mu \sum_j \frac{m_j}{\rho_i^2} (\mathbf{u}_j - \mathbf{u}_i) \nabla^2 W_h^{ji} \quad (2.21)$$

Si on ne faisait pas ces changements, les calculs seraient instables et mèneraient à des résultats incohérents.

Animation du fluide

On connaît donc l'accélération de chacune des particules, en sommant les différentes forces⁶ que l'on vient de calculer : $\frac{D\mathbf{u}_i}{Dt} = \frac{1}{\rho_i} (\mathbf{f}_i^p + \mathbf{f}_i^v + \mathbf{f}_i^e)$. Une intégration permet de connaître la nouvelle vitesse des particules, puis une seconde la position. On a donc :

$$\mathbf{u}_i \leftarrow \mathbf{u}_i + \frac{D\mathbf{u}_i}{Dt} \Delta t \quad (2.22)$$

$$\mathbf{r}_i \leftarrow \mathbf{r}_i + \mathbf{u}_i \Delta t \quad (2.23)$$

Connaissant les positions de toutes les particules de fluide à la date t , il est possible de les afficher et donc de simuler le fluide.

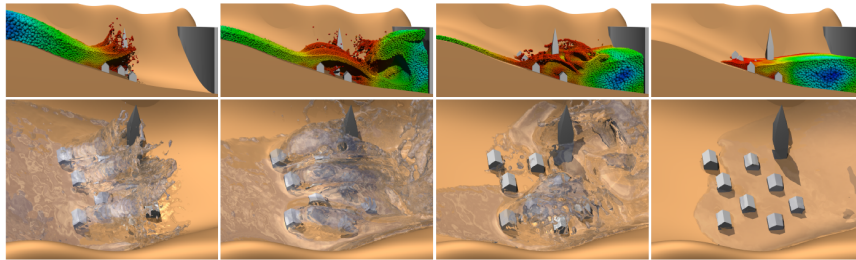


FIGURE 2.6 – Illustration SPH tirée de ADAMS et WICKE (2009)

La gestion des forces extérieures est très simple puisqu'il suffit de modifier la force $\mathbf{f}_i^e$. Dans le cas où une particule de fluide rencontre un obstacle, comme un mur ou un verre, la solution la plus couramment utilisée (KELAGER 2006) consiste à faire rebondir la particule de fluide sur la paroi. Pour cela, on modifie la vitesse ainsi :

$$\mathbf{u}_i = \mathbf{u}_i - (1 + c_r)(\mathbf{u}_i \cdot \vec{n})\vec{n}$$

6. En l'absence de forces extérieures, on posera $\mathbf{f}_i^e = \rho \mathbf{g}$

où $\vec{n}$ est le vecteur normal à la surface au point de l'impact et où c_r est le coefficient de restitution. $c_r \in [0; 1]$, où $c_r = 1$ modélise une collision élastique sans aucune perte d'énergie cinétique, alors que $c_r = 0$ modélise la condition inélastique (ou *réflective*), qui est normalement choisie pour modéliser un fluide.

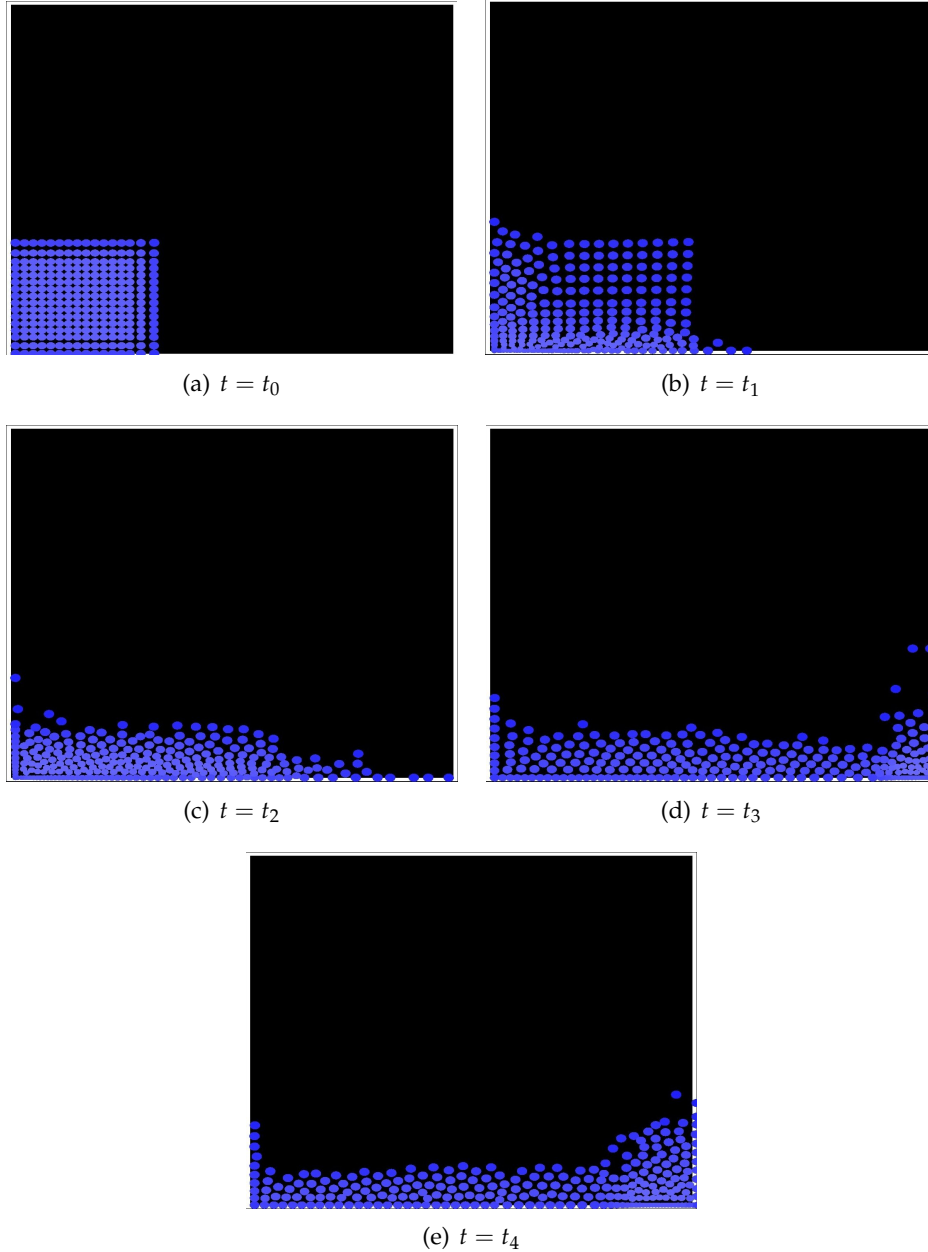


FIGURE 2.7 – Simulation d'un barrage qui rompt

2.4 BILAN

Les trois méthodes que nous venons de voir permettent de simuler le mouvement d'un fluide, chacune avec ses spécificités. Par exemple, la résolution des équations de Saint-Venant, telle que nous l'avons faite, nous permet de connaître l'état de la surface de l'eau. La résolution est relati-

vement simple, mais ne permet pas de simuler tous les comportements possibles du fluide. Comme nous l'avons dit, il est possible de faire un *splash*, ou encore d'essayer de simuler la rupture d'un barrage de cette manière.

Pour décrire tous les mouvements possibles d'un fluide, il faut utiliser des méthodes comme LBM et SPH. Et là encore, il faut faire un choix entre ces deux méthodes qui n'ont pas tout à fait les mêmes applications.

Les méthodes de Boltzmann sur réseau et de *Smoothed Particle Hydrodynamics* ont des coûts de calculs très importants. Une implémentation *naïve* de SPH a un coût de $O(N^2)$, où N est le nombre de particules. Nous allons voir comment il est possible d'améliorer cette complexité et de tendre vers du $O(N \log N)$ ou du $O(Nm)$, selon les optimisations choisies (m étant le nombre moyens de voisins d'une particule de fluide). Commençons par voir rapidement l'idée générale de la parallélisation de LBM, ainsi que quelques autres optimisations possibles.

3.1 QUELQUES OPTIMISATIONS POUR LBM

3.1.1 Vers une parallélisation de LBM

L'avantage certain, du point de vue du développeur, de LBM est la simplicité qui réside dans l'écriture de l'algorithme. En revanche, il faut beaucoup de mémoire pour stocker les valeurs calculées et de temps pour les parcourir. Afin, d'accélérer le calcul il est possible de paralléliser l'algorithme.

Regardons ce que proposent [PENG et al.](#)

Ils suggèrent de diviser le domaine où évolue le fluide, appelé Ω en sous domaines Ω_i , tels que $\Omega = \cup \Omega_i$. Chaque Ω_i est alors affecté à un processeur. Les opérations de *streaming* et de collision se font sur chaque Ω_i en même temps. Ils définissent une fonction de *communication* qui est chargée des migrations des fonctions de distribution f_i entre les différents sous-domaines.

Ainsi, sur la figure 3.1 p. 30, le domaine Ω est divisé en quatre sous-domaines. Si l'on regarde Ω_0 , les mailles blanches représentent les endroits où les f_i seront calculées par le processeur 0. Les mailles noires représentent des obstacles, et les mailles grises sont les mailles gérées grâce à la fonction de communication.

3.1.2 Autres optimisations

Il existe bien d'autres moyens pour optimiser LBM. Il est par exemple possible de combiner l'étape de *streaming* et l'étape de collision. Ainsi, au lieu de parcourir deux fois toutes les mailles, on ne fait qu'un seul tour, ce qui a pour effet de diminuer la constante cachée par la notation $O(N^2)$ (dans le cas 2D). Pour faire cela, on peut utiliser une technique qui se nomme *grid-merging* qui consiste à utiliser deux tableaux : un qui correspond au temps t et l'autre au temps $t + 1$. Ainsi, À chaque itération, on

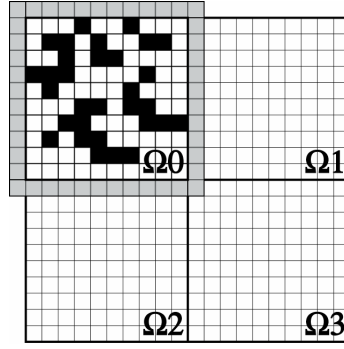


FIGURE 3.1 – Schématisation de la décomposition spatiale d'un domaine Ω . (PENG *et al.*)

calcule les nouveaux f_i et on stocke le résultat dans la case correspondant à $t + 1$. L'image 3.2 illustre cette technique. À l'itération suivante, les tableaux seront intervertis.

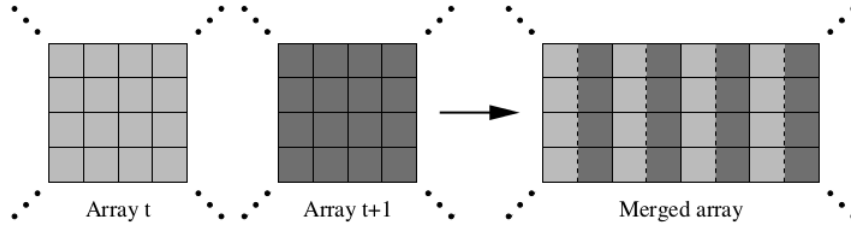


FIGURE 3.2 – Technique de grid-merging (WILKE *et al.*)

D'autres optimisations peuvent être faites, notamment en utilisant le cache du CPU. Ces optimisations sont décrites par WILKE *et al.* et se nomment *1D – Blocking*, *2D – Blocking*.

3.2 UNE MEILLEURE STRUCTURE DE DONNÉES POUR SPH

Pour que la simulation SPH soit de qualité et ait un rendu le plus réaliste possible, il faut utiliser un nombre important de billes ; plusieurs milliers. Le problème qui se pose alors est celui du temps de calcul. Reprenons la formule permettant d'obtenir la force de pression pour illustrer ce problème.

$$\mathbf{f}_i^p = -\rho_i \sum_j m_j \left(\frac{P_j}{\rho_j^2} + \frac{P_i}{\rho_i^2} \right) \vec{\nabla} W_h^{ji} \quad (3.1)$$

Pour calculer la force de pression qui s'exerce sur la bille i , il faut sommer N termes (N correspondant au nombre de billes). Étant donné qu'il faut faire ce calcul pour chaque bille du système, on se retrouve avec un calcul de l'ordre $O(N^2)$, juste pour calculer la force de pression... Puis il faut recommencer au pas de temps suivant. Dès que N devient grand, les calculs sont infaisables sur un ordinateur personnel¹. Et tous ces calculs sont faits, alors que seules les billes j telles que $\|\mathbf{r}_i - \mathbf{r}_j\| \leq h$ ont un impact sur le résultat.

1. À titre d'exemple, sur mon ordinateur, avec un algorithme SPH en $O(N^2)$, dès que le nombre de billes dépasse 170, c'est inutilisable... alors qu'il faudrait environs 5000 billes pour simuler de façon réaliste $0.1m^3$ d'eau (KELAGER 2006).

Il faut donc trouver un moyen d'obtenir rapidement les voisins d'une bille. Deux méthodes sont principalement utilisées :

1. La recherche dans un *arbre-kd*
2. La recherche dans une table de hashage

Voyons rapidement de quoi il s'agit.

3.2.1 Recherche dans un *arbre-kd*

Commençons par définir ce qu'est un arbre-kd.

Définition 3.1 *Un arbre-kd est une structure de données de partition de l'espace permettant de stocker des points, et de faire des recherches plus rapidement qu'en parcourant linéairement le tableau de points. (Wikipédia²)*

Techniquement, un arbre-kd est un arbre binaire où à chaque niveau la clé de tri change. Dans l'exemple ci-dessous, la discrimination se fait selon x , puis selon y puis selon x etc. Ce raisonnement s'étend aisément au cas 3D.

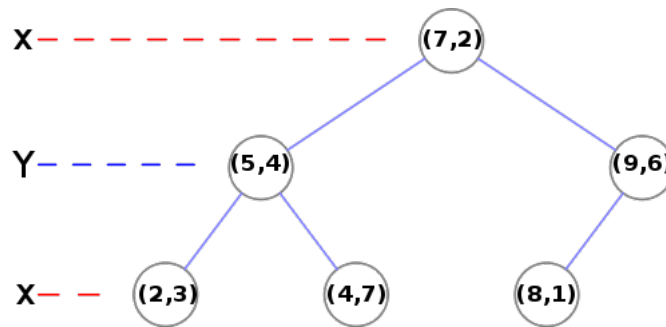


FIGURE 3.3 – Représentation d'un arbre-kd (http://en.wikipedia.org/wiki/K-d_tree)

À l'état initial, on construit³ l'arbre 2D, $\mathcal{T}$. Cette opération a un coût de l'ordre de $O(N \log N)$. Puis, pour chaque particule de fluide i , on demande à $\mathcal{T}$ l'ensemble de particules de fluide qui sont dans les cellules qui intersectent le disque de rayon h , centré en $\mathbf{r}_i$. Soit l'ensemble :

$$\mathcal{N}(p_i) = \{p \in \mathcal{T}, \|\mathbf{r}_i - \mathbf{r}\| \leq h\}$$

L'algorithme SPH est alors en $O(N \log N)$, la recherche dans l'arbre (binaire) se faisant en $O(\log N)$.

Lors d'une nouvelle itération, on préfère reconstruire entièrement l'arbre plutôt que d'essayer de le mettre à jour, étant donné le faible coût de la construction (ADAMS et WICKE 2009).

3.2.2 Recherche dans une table de hashage

Bien que la complexité de la recherche dans une table de hashage soit moins bonne, son implémentation est assez simple. Nous la décrirons donc un peu plus en détail.

2. Janvier 2011 — http://fr.wikipedia.org/wiki/Arbre_kd

3. La construction de l'arbre est admise dans le cadre de cette IR. Un algorithme de construction est donné dans (ADAMS et WICKE 2009).

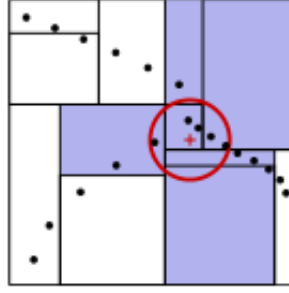


FIGURE 3.4 – Calcul de $\mathcal{N}(p_i)$. Seules les cellules violettes sont lues. (ADAMS et WICKE 2009)

L'utilisation d'une table de hachage permet de connaître en $O(1)$ si à la position (x, y, z) il y a des particules de fluide ou non. L'idée est alors la suivante : en supposant la table de hachage construite, pour déterminer les voisins de la particule i , de coordonnées (x_i, y_i, z_i) il suffit de balayer l'ensemble des positions discrètes entre $(x_i - h, y_i - h, z_i - h)$ et $(x_i + h, y_i + h, z_i + h)$ et stocker les particules trouvées dans l'ensemble $\mathcal{N}(x_i)$ grâce à la table de hachage pour faire ensuite les calculs voulus.

Commençons par construire la table. Il nous faut pour cela une fonction qui à une position dans l'espace⁴ associe un index. Le but est donc de trouver une fonction *le plus injective possible* de $\mathbb{R}^3$ dans $\mathbb{N}$, afin de minimiser les collisions. La fonction utilisée, que l'on retrouve notamment dans (KELAGER 2006, ADAMS et WICKE 2009), est la suivante :

$$\text{hash}(\mathbf{r}) = \left(\left\lfloor \frac{\mathbf{r}_x}{h} \right\rfloor \times p_1 \text{ XOR } \left\lfloor \frac{\mathbf{r}_y}{h} \right\rfloor \times p_2 \text{ XOR } \left\lfloor \frac{\mathbf{r}_z}{h} \right\rfloor \times p_3 \right) \bmod M \quad (3.2)$$

où :

- $p_{1,2,3}$ sont trois nombres premiers. KELAGER (2006) utilisent ceux-ci :

$$p_1 = 73856093 \quad p_2 = 19349663 \quad p_3 = 83492791$$

- M est la taille de la table de hachage. M ne doit être ni trop grand pour ne pas alourdir la taille en mémoire, ni trop petit pour éviter au maximum les collisions. KELAGER (2006) utilise : $M = \text{prime}(2N)$, où $\text{prime}()$ est une fonction qui retourne le nombre premier suivant $2N$.

Construction de la table de hachage À chaque pas de temps, il faut construire la table de hachage, ce qui à un coût de $O(N)$. L'algorithme 1, permet de construire la table de hachage. Pour chaque particule du système, on calcule son *hash* à l'aide de la formule 3.2, puis on l'insère dans la case du tableau correspondante. Pour gérer les collisions, chaque case du tableau contient non pas une particule, mais un ensemble de particules (initialement vide).

Construction de l'ensemble $\mathcal{N}(p_i)$ Pour calculer les informations nécessaires à la résolution de l'équation du bilan de quantité de mouvement de

4. On suppose que l'on travaille en 3D. La transposition en 2D étant immédiate.

Algorithme 1: Création de la table de hashage**Entrées :** P , ensemble des particules du système**Données :** $table_{hash}$, un tableau de type *ensemble de particules*, de taille M . Tableau dont chaque case est initialisée à $\emptyset$.**début** **pour chaque** $p \in P$ **faire** $table_{hash}[hash(p.r)] \leftarrow table_{hash}[hash(p.r)] \cup p$ **retourner** $table_{hash}$

Navier - Stokes, il faut pour chaque particule p_i connaître les informations que contiennent les particules voisines. Ainsi, dans l'algorithme de SPH, lorsque l'on itère sur toutes les particules p_i , on va construire à l'aide de la table de hashage l'ensemble $\mathcal{N}(p_i)$. Pour cela, on va parcourir l'ensemble des positions discrètes entre $(x_i - h, y_i - h, z_i - h)$ et $(x_i + h, y_i + h, z_i + h)$ et stocker les particules trouvées dans l'ensemble $\mathcal{N}(x_i)$ grâce à la table de hashage. L'algorithme est le suivant :

Algorithme 2: Calcul de $\mathcal{N}(p_i)$ **Entrées :** p_i , particule dont on cherche les voisins. $table_{hash}$, la table de hashage construite grâce à l'algorithme 1.**Données :** $neighborsSet = \emptyset$ **début** **pour** $x = p_i.r_x - h$ **à** $p_i.r_x + h$ **par pas de** h **faire** **pour** $y = p_i.r_y - h$ **à** $p_i.r_y + h$ **par pas de** h **faire** **pour** $z = p_i.r_z - h$ **à** $p_i.r_z + h$ **par pas de** h **faire** $neighborsSet = neighborsSet \cup$ $\{p \in table_{hash}[hash(x, y, z)], \|p.r - p_i.r\| \leq h\}$ **retourner** $neighborsSet$

On remarquera que l'on ne prend qu'un sous-ensemble de $table_{hash}[hash(x, y, z)]$. En effet, à cause des collisions, il n'est pas improbable que deux points aussi différents qu'ils soient, aient le même hash. Cet algorithme est de l'ordre de $O(m)$, où m est le nombre moyen de voisins d'une particule.

Algorithme général Ainsi, l'algorithme général de SPH peut être décrit par :

1. Construire la table de hashage.
2. Pour chaque particule p_i du système :
 - (a) Construire $\mathcal{N}(p_i)$.
 - (b) Pour chaque particule $p_j \in \mathcal{N}(p_i)$, calculer la densité de p_i et sa pression, à l'aide des p_j .
3. Pour chaque particule p_i du système :
 - (a) Construire $\mathcal{N}(p_i)$.
 - (b) Pour chaque particule $p_j \in \mathcal{N}(p_i)$,

- i. Calculer les accélérations auxquelles est soumise p_i , grâce aux p_j
 - (c) Calculer la nouvelle vitesse de p_i et sa nouvelle position, par intégration.
4. Recommencez à l'étape 1

Si on appelle m , le nombre moyen de voisins d'une bille, alors une itération a un coût de $O(Nm)$. On peut remarquer que pour chaque particule de fluide, on calcule deux fois l'ensemble $\mathcal{N}(p_i)$. Afin de diminuer la constante cachée par la notation *grand O*, [KELAGER \(2006\)](#) suggère de stocker dans un ensemble les paires $(p_i, \mathcal{N}(p_i))$, pour ne les calculer qu'une seule fois. Ce qui a pour effet de diminuer le nombre de calculs réellement effectués (tout en gardant une complexité identique).

3.3 BILAN

Simuler un fluide est, comme nous venons de le voir, une opération coûteuse. Même après optimisations, pour SPH on a au mieux du $O(N \log N)$, avec un N de l'ordre du millier. Comment, en pratique, simuler un fluide alors ? La solution est simple. Il faut utiliser des techniques pour tromper le spectateur : simuler 10cm^3 de fluide en faisant croire, grâce à un jeu d'échelle qu'il s'agit d'un barrage de 10m de hauteur par exemple.

Si on s'intéresse aux optimisations présentées pour SPH, la figure suivante illustre ce que nous avons montré. L'utilisation d'une table de hashage est simple et l'implémentation rapide, par rapport à l'arbre-kd. Mais, elle est beaucoup moins adaptative que l'arbre, comme l'illustre la figure.

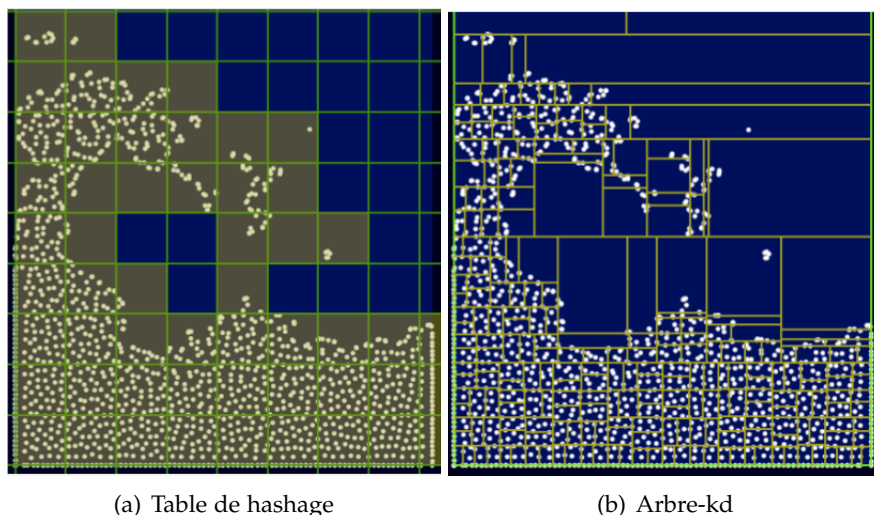


FIGURE 3.5 – Structures de recherche ([ADAMS et WICKE 2009](#))

CONCLUSION GÉNÉRALE

Ainsi s'achève ce mémoire. Cette étude a été l'occasion de faire le point sur différentes méthodes de simulations de fluides. Nous avons donc vu la démarche qu'il y a à réaliser pour faire une simulation. Tout d'abord, cerner mathématiquement le problème, essayer de trouver les équations qui définissent le mouvement. Ensuite le problème, qui sur un plan personnel me plait le plus, chercher les moyens qui vont permettre de passer des équations à la simulation. C'est là qu'interviennent les méthodes de Boltzmann sur réseau ou encore SPH par exemple. Et pour finir, cette étude ne pouvant être complète sans une étude de faisabilité, pour assurer une complexité raisonnable, je me suis intéressé aux moyens existants pour optimiser ces méthodes.

C'est un sentiment assez *drôle* que de rédiger ce qu'on appelle une conclusion, alors qu'il s'agit plutôt pour moi (du moins, je l'espère) du point de départ d'autres aventures. Le point de départ me permettant de faire ce que je n'ai pas pu faire ici. Par exemple, j'aurai aimé pouvoir m'intéresser d'avantage au temps réel avec SPH, ou encore étudier les différentes techniques d'extractions de surfaces permettant de passer des particules de fluides de SPH (ou des fonctions de distribution de LBM) à de vraies surfaces, comme sur la figure 2.6 (je pense aux iso-surfaces et à α -shape). J'aurai aussi aimé m'intéresser à l'implémentation de SPH en 3D avec OpenGL, ainsi qu'à utilisation de particules fantômes pour la gestion des frontières, mais ce sera dorénavant pour une prochaine fois.

LA MÉTHODE DE BOLTZMANN SUR RÉSEAU

A

A.1 OBTENTION DE LA DISTRIBUTION D'ÉQUILIBRE DE LA FONCTION DE BOLTZMANN

Le but de cette section est de justifier l'équation 2.8, donnant l'expression de la fonction de distribution à l'équilibre utilisée dans la méthode de Boltzmann sur réseau. Pour rappel, on a :

$$f_i^{eq} = w_i \rho \left[1 + 3(\mathbf{e}_i \cdot \mathbf{u}) - \frac{3}{2} \mathbf{u}^2 + \frac{9}{2} (\mathbf{e}_i \cdot \mathbf{u})^2 \right]$$

où les w_i sont des coefficients donnés.

La distribution à l'équilibre normalisée de Maxwell peut s'écrire comme (MOHAMAD 2011) :

$$f^{eq} = \frac{\rho}{\frac{2}{3}\pi} e^{-\frac{3}{2}(\mathbf{e}-\mathbf{u})^2}$$

Ce que l'on peut aussi écrire comme :

$$f^{eq} = \frac{\rho}{\frac{2}{3}\pi} e^{-\frac{3}{2}(\mathbf{e}^2 - 2\mathbf{e} \cdot \mathbf{u} + \mathbf{u}^2)} = \frac{\rho}{\frac{2}{3}\pi} e^{-\frac{3}{2}\mathbf{e}^2} e^{-\frac{3}{2}(\mathbf{u}^2 - \mathbf{e} \cdot \mathbf{u})}$$

On a donc :

$$f_i^{eq} = w_i \rho e^{-\frac{3}{2}(\mathbf{u}^2 - \mathbf{e}_i \cdot \mathbf{u})} \quad \text{où } w_i = \frac{1}{\frac{2}{3}\pi} e^{-\frac{3}{2}\mathbf{e}_i^2}$$

Pour finir, il ne reste plus qu'à faire un développement limité de $e^{-\frac{3}{2}(\mathbf{u}^2 - \mathbf{e}_i \cdot \mathbf{u})}$ à l'ordre 2 :

$$\begin{aligned} f_i^{eq} &= w_i \rho \left(1 - \frac{3(\mathbf{u}^2 - 2\mathbf{e}_i \cdot \mathbf{u})}{2} + \frac{1}{2} \left(\frac{3}{2} (\mathbf{u}^2 - 2\mathbf{e}_i \cdot \mathbf{u}) \right)^2 + O(\mathbf{u}^2) \right) \\ &\approx w_i \rho \left(1 + 3(\mathbf{e}_i \cdot \mathbf{u}) - \frac{3}{2} \mathbf{u}^2 + \frac{9}{2} (\mathbf{e}_i \cdot \mathbf{u})^2 \right) \end{aligned}$$

D'où le résultat.

A.2 IMPLÉMENTATION DE LBM

Cette annexe explicite l'algorithme utilisé par le programme vu sur la figure 2.5, page 21. Il s'agit donc d'un maillage D_2Q_9 .

Algorithme 3: Initialisation des variables globales**début** $nx \leftarrow 60$ $ny \leftarrow 60$ $w[9] \leftarrow (4/9, 1/9, 1/36, 1/9, 1/36, 1/9, 1/36, 1/9, 1/36)$ $ex[9] \leftarrow (0, 1, 1, 0, -1, -1, -1, 0, 1)$ $ey[9] \leftarrow (0, 0, 1, 1, 1, 0, -1, -1, -1)$ *Ce tableau contient les indices des vitesses de direction opposée* $invert[9] \leftarrow (0, 5, 6, 7, 8, 1, 2, 3, 4)$ $\omega \leftarrow 1.9$ **Algorithme 4:** Initialisation du domaine et du fluide**Entrées :** $domaine[nx][ny]$, $f[nx][ny][9]$ **Sorties :** $domaine[nx][ny]$, $f[nx][ny][9]$ **début****pour** $i \in [0; nx - 1]$ **faire****pour** $j \in [0; ny - 1]$ **faire****si** $i = 0$ **OU** $i = nx - 1$ **OU** $j = 0$ **OU** $j = ny - 1$ **alors** $\quad \quad \quad domaine[i][j] \leftarrow OBS$ **sinon** $\quad \quad \quad domaine[i][j] \leftarrow FLUID$ **pour** $k \in [0; 8]$ **faire** $\quad \quad \quad f[i][j][k] \leftarrow w[k]$ **si** $i = nx/2$ **ET** $j = ny/2$ **alors** $\quad \quad \quad \quad \quad f[i][j][k] \leftarrow f[i][j][k] + 0.5$

Algorithme 5: Calcul d'une itération complète**Entrées :** $\text{domaine}[nx][ny]$, $f[nx][ny][9]$ **Sorties :** $\text{domaine}[nx][ny]$, $f[nx][ny][9]$ **début**

/* Streaming step */

pour $i \in [0; nx - 1]$ **faire** **pour** $j \in [0; ny - 1]$ **faire** **pour** $k \in [0; 8]$ **faire** $\text{kinv} \leftarrow \text{invert}[k]$ **si** $\text{domaine}[i + \text{ex}[\text{kinv}]] [j + \text{ey}[\text{kinv}]] = \text{OBS}$ **alors** $f_p[i][j][k] = f[i][j][\text{kinv}]$ **sinon** $f_p[i][j][k] = f[i + \text{ex}[\text{kinv}]] [j + \text{ey}[\text{kinv}]] [k]$

/* Collide step */

pour $i \in [0; nx - 1]$ **faire** **pour** $j \in [0; ny - 1]$ **faire** **si** $\text{domaine}[i][j] = \text{OBS}$ **alors**

continue

 $(\rho, ux, uy) = \text{getDensityAndVelocities}(f, i, j)$ **pour** $k \in [0; 8]$ **faire** $eu \leftarrow \text{ex}[k] * ux + \text{ey}[k] * uy$ $f_{eq} \leftarrow$ $\rho * w[k] * (1 + 3 * eu + 9/2 * eu^2 - 3/2 * (ux^2 + uy^2))$ $f[i][j][k] = (1 - \omega) * f_p[i][j][k] + \omega f_{eq}$ **Algorithme 6:** $\text{getDensityAndVelocities}()$ **Entrées :** $i, j, f[nx][ny][9]$ **Sorties :** ρ, ux, uy **début** $\rho \leftarrow 0$ $ux \leftarrow 0$ $uy \leftarrow 0$ **pour** $k \in [0; 8]$ **faire** $\rho \leftarrow \rho + f[i][j][k]$ $ux \leftarrow ux + \text{ex}[k] * f[i][j][k]$ $uy \leftarrow uy + \text{ey}[k] * f[i][j][k]$ $ux \leftarrow \frac{ux}{\rho}$ $uy \leftarrow \frac{uy}{\rho}$

SMOOTHED PARTICLE HYDRODYNAMICS

B

B.1 CONVERGENCE VERS $\delta(x)$

Il s'agit de prouver ici que toute fonction $W(x, h)$ ayant les propriétés suivantes converge vers la distribution de Dirac $\delta(x)$, quand h tend vers 0 :

$$W(x, h) \geq 0 \quad \int W(x, h) dx = 1, \quad \text{supp}(W(x, h)) \subset]-h; h[$$

Preuve. On dit qu'une suite de distributions (T_n) converge vers T si et seulement :

$$\lim_{n \rightarrow \infty} \langle T_n, \varphi \rangle = \langle T, \varphi \rangle \quad \forall \varphi \in D$$

Puisque le support de W est bornée et que $\int W(x, h) dx = 1$, on en déduit que W appartient à L^1_{loc} , l'espace des fonctions localement intégrables, et définit donc une distribution. On pose $W_n(x) = W(x, \frac{1}{n})$.

$$\begin{aligned} \langle W_n, \varphi \rangle &= \int W_n \varphi dx \\ &= \int_{-\frac{1}{n}}^{\frac{1}{n}} W_n \varphi dx \quad \text{car } \text{supp}(W_n(x)) \subset \left] -\frac{1}{n}; \frac{1}{n} \right[\\ &= \varphi(c) \underbrace{\int_{-\frac{1}{n}}^{\frac{1}{n}} W_n dx}_{=1} \quad c \in \left] -\frac{1}{n}; \frac{1}{n} \right[, \text{ (théo. de la moyenne)} \end{aligned}$$

Ainsi, $\lim_{n \rightarrow \infty} \langle W_n, \varphi \rangle = \varphi(0) = \langle \delta, \varphi \rangle, \forall \varphi \in D$.

D'où :

$$\lim_{h \rightarrow 0} W(x, h) = \delta(x)$$

□

B.2 EXEMPLE DE NOYAUX UTILISÉS POUR SPH

Ces noyaux sont donnés par [ADAMS et WICKE \(2009\)](#).

B.2.1 Simulation normale

Pour une simulation 3D

$$W_h(d) = \begin{cases} \frac{315}{64\pi h^3} \left(1 - \frac{d^2}{h^2}\right)^3 & \text{Si } d < h \\ 0 & \text{Sinon} \end{cases} \quad (\text{B.1})$$

$$W'_h(d) = \begin{cases} -\frac{945}{32\pi h^3} \frac{d}{h^2} \left(1 - \frac{d^2}{h^2}\right)^2 & \text{Si } d < h \\ 0 & \text{Sinon} \end{cases} \quad (\text{B.2})$$

$$W''_h(d) = \begin{cases} \frac{945}{32\pi h^3} \frac{1}{h^2} \left(1 - \frac{d^2}{h^2}\right) \left(5\frac{d^2}{h^2} - 1\right) & \text{Si } d < h \\ 0 & \text{Sinon} \end{cases} \quad (\text{B.3})$$

Pour une simulation 2D

$$W_h(d) = \begin{cases} \frac{4}{\pi h^2} \left(1 - \frac{d^2}{h^2}\right)^3 & \text{Si } d < h \\ 0 & \text{Sinon} \end{cases} \quad (\text{B.4})$$

$$W'_h(d) = \begin{cases} -\frac{24}{\pi h^2} \frac{d}{h^2} \left(1 - \frac{d^2}{h^2}\right)^2 & \text{Si } d < h \\ 0 & \text{Sinon} \end{cases} \quad (\text{B.5})$$

$$W''_h(d) = \begin{cases} \frac{24}{\pi h^2} \frac{1}{h^2} \left(1 - \frac{d^2}{h^2}\right) \left(5\frac{d^2}{h^2} - 1\right) & \text{Si } d < h \\ 0 & \text{Sinon} \end{cases} \quad (\text{B.6})$$

B.2.2 Simulation à faible résolution

ADAMS et WICKE (2009) suggèrent d'utiliser pour les simulations à faible résolution les noyaux suivants, dits *noyaux hérissés*, afin d'éviter un phénomène d'agglutination.

Pour une simulation 3D

$$\hat{W}_h(d) = \begin{cases} \frac{15}{\pi h^3} \left(1 - \frac{d}{h}\right)^3 & \text{Si } d < h \\ 0 & \text{Sinon} \end{cases} \quad (\text{B.7})$$

$$\hat{W}'_h(d) = \begin{cases} -\frac{45}{\pi h^3} \frac{1}{h} \left(1 - \frac{d}{h}\right)^2 & \text{Si } d < h \\ 0 & \text{Sinon} \end{cases} \quad (\text{B.8})$$

$$\hat{W}''_h(d) = \begin{cases} \frac{90}{\pi h^3} \frac{1}{h^2} \left(1 - \frac{d}{h}\right) & \text{Si } d < h \\ 0 & \text{Sinon} \end{cases} \quad (\text{B.9})$$

Pour une simulation 2D

$$\hat{W}_h(d) = \begin{cases} \frac{10}{\pi h^2} \left(1 - \frac{d}{h}\right)^3 & \text{Si } d < h \\ 0 & \text{Sinon} \end{cases} \quad (\text{B.10})$$

$$\hat{W}'_h(d) = \begin{cases} -\frac{30}{\pi h^2} \frac{1}{h} \left(1 - \frac{d}{h}\right)^2 & \text{Si } d < h \\ 0 & \text{Sinon} \end{cases} \quad (\text{B.11})$$

$$\hat{W}''_h(d) = \begin{cases} \frac{60}{\pi h^2} \frac{1}{h^2} \left(1 - \frac{d}{h}\right) & \text{Si } d < h \\ 0 & \text{Sinon} \end{cases} \quad (\text{B.12})$$

DOCUMENTS NÉCESSAIRES À L'UV

C

Dans les documents suivants, j'ai supprimé les références bibliographiques qui étaient originalement présentes pour ne pas alourdir inutilement la bibliographie automatique.

C.1 NOTE DE CLARIFICATION

Voici la note de clarification, telle que rédigée en octobre 2011.

C.1.1 Contexte, motivation et problématique

« La modélisation mathématique des mouvements d'un fluide a été de tous temps un sujet de recherche actif. L'étude des interactions du fluide avec son environnement (côtes, corps flottants, immergés, fixes ou en mouvement) a été longtemps un sujet intéressant essentiellement les domaines de la construction navale et du génie civil. Depuis une vingtaine d'années, ce sujet intéresse aussi le domaine de l'animation et les mouvements de la mer (houles, vagues déferlantes, etc.) sont maintenant rendus de manière très convaincante pour ne pas dire réaliste. [...] »

Cette étude devra permettre de dresser un inventaire des différents modèles mathématiques utilisés pour simuler un fluide dans son environnement. Cet inventaire sera comparatif, dans le sens où les modèles présentés seront confrontés entre eux afin de comprendre dans quel(s) contexte(s) ils peuvent être utilisés — génie civil, animation, eaux profondes, eaux peu profondes. . .

Cette étude devrait m'apporter des connaissances théoriques et un regard critique sur les techniques de la modélisation d'un fluide dans son environnement. J'aimerais ainsi à travers cette étude acquérir des connaissances que je pourrai utiliser plus tard ; dans le cadre d'une étude plus *pratique* par exemple ¹.

C.1.2 Démarche employée

L'information sera cherchée dans les publications de travaux scientifiques traitant du sujet, dans des mémoires de thèses. J'utiliserai égale-

1. C'est à dire, modéliser un fluide en utilisant un des modèles présentés.

ment certains ouvrages, ainsi que des présentations réalisées lors de colloques. Je me servirai aussi des supports de cours universitaires. Cette liste est non-exhaustive, et certains supports ne seront peut-être pas pris en compte dans le mémoire final.

Concernant la recherche sur la modélisation en animation, je m'intéresserai principalement au fonctionnement et aux modèles utilisés par le simulateur de fluide du logiciel libre d'animation 3D *Blender*, à savoir *El'Beem*. Ce simulateur est basé sur la méthode de Boltzman sur réseau, méthode à laquelle je m'intéresserai particulièrement.

Je pense également prendre un rendez-vous avec M. Ouahsine Abdelatif qui fait des recherches sur le sujet de la modélisation de la houle en zone côtière² au laboratoire Roberval.

C.1.3 Calendrier prévisionnel

Note de clarification	Mi-octobre
Prise de rendez - vous avec M. Ouahsine	Fin octobre
Première synthèse	Mi-novembre
Plan final avec bibliographie	Fin novembre
Début de la rédaction du mémoire	Début décembre
Rendu du mémoire	Mi-janvier

C.2 PLAN PROVISOIRE

Voici le plan provisoire tel que rédigé en décembre 2011.

C.2.1 Introduction à la modélisation d'un fluide

Eulérien VS. Lagrangien

On présentera ici les deux modèles conceptuels que sont le modèle Eulérien et Lagrangien. Cette présentation permettra d'introduire les différents modèles mathématiques présentés dans la partie 2. Les deux conceptions seront également comparées d'un point de vue pratique ; lequel est le plus adapté à des frontières variables etc.

Description Lagrangienne On décrira ici ce qu'est la description Lagrangienne, ce qu'elle permet etc.

- On s'intéresse au mouvement des particules de fluides prise individuellement
- advection (contamination, suivi d'éléments radioactifs)

Description Eulérienne On décrira ici ce qu'est la description Eulérienne, ce qu'elle permet etc.

- Mouvement global du fluide.
- Lignes de courants

2. <http://www.umr6253roberval.fr/spip.php?article64>

Équation de Navier - Stokes

Un fluide newtonien peut être décrit par les équations de Navier - Stokes :

$$0 = \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) \quad (\text{C.1})$$

$$\rho \frac{D \mathbf{u}}{Dt} = -\nabla P + \mu \nabla^2 \mathbf{u} + \mathbf{F} \quad (\text{C.2})$$

Le but de cette section est de présenter les équations de Navier - Stokes, de définir les variables utilisées dans ces équations — et que seront utilisées dans la suite — , décrire comment elles fonctionnent et surtout d'où elles viennent.

C.2.2 Modélisation d'une vague

Cette section, sans doute la plus conséquente du mémoire présentera 3 méthodes différentes utilisées pour modéliser des vagues. Seront présentées, les équations de Shallow Water, la méthode de Boltzmann sur réseau et pour finir la méthode Smoothed Particles Hydrodynamics.

Shallow Water Equation

Dans cette sous-section, consacrée aux équations de Shallow Water, on montrera comment on peut les dériver depuis les équations de Navier Stokes et comment on peut approcher une solution de ces équations grâce aux différences finies. (2D seulement, je pense...)

Étant donné la forme de la solution, $h = f(x, y)$, il est impossible d'avoir un rouleau, la surface que nous obtenons ne peut pas se replier sur elle même. Pour un rendu plus réaliste d'une vague, nous devons utiliser un modèle différent.

Lattice Boltzmann Method

Dans cette sous-section sera introduite la méthode de Boltzmann sur réseaux. Après un bref historique, on explicitera en quoi ce modèle est clairement une approche différente de la précédente.

Équation de Boltzmann continue On présentera ici l'équation continue de Boltzmann. On montrera comment parvenir à cette équation.

$$\frac{\partial f}{\partial t} + \mathbf{u} \frac{\partial f}{\partial \mathbf{r}} + \frac{\mathbf{F}}{m} \frac{\partial f}{\partial \mathbf{u}} = \Omega(f) \quad (\text{C.3})$$

Discrétisation de l'équation de Boltzmann On expliquera ici pourquoi et comment se fait la discrétisation de l'équation de Boltzmann. On présentera naturellement les modèles de discrétisation usuels D_2Q_9 et D_3Q_{19} . Ensuite, on s'intéressera à la résolution de l'équation.

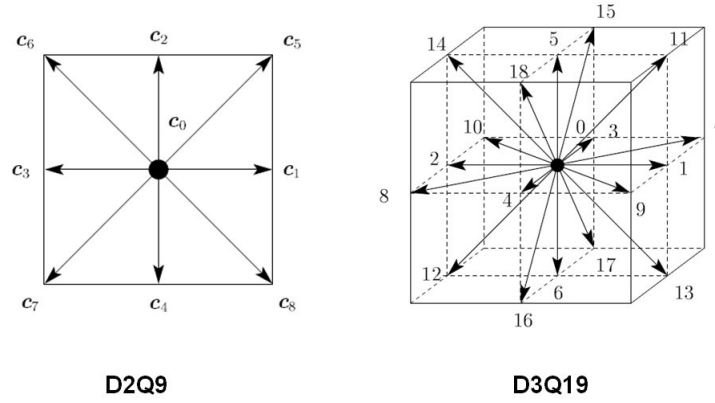


FIGURE C.1 – Maillages couramment utilisés

Cas du modèle D_3Q_{19} On expliquera ici que pour chaque vecteur vitesse, une fonction de distribution des particules f_i est définie. f_i représente la quantité de particules de fluides de déplaçant à la vitesse $\mathbf{e}_i$.

On définira les vecteurs vitesses de ce modèle.

Description de la méthode Il y a deux étapes,

1. la phase de mouvement
2. la phase de collision

L'étape 1 consiste à définir les fonctions de distribution f'_i en fonction des f_i en supposant qu'il a pas eu de collisions. On a donc :

$$f'_i(\mathbf{r}, t + \Delta t) = f_i(\mathbf{r} - \Delta t \mathbf{e}_i, t)$$

À l'étape 2, les collisions entre en jeu. On définit alors la densité ρ et la vitesse du fluide $\mathbf{u}$ à l'aide des f_i . Puis on calcul f_i^{eq} .

À cette étape, après avoir effectué les calculs précédents on mets à jour chaque f_i de la façon suivante :

$$f_i(\mathbf{r}, t + \Delta t) = f'_i + \frac{f_i^{eq} - f'_i}{\lambda}$$

Quid des conditions aux limites ? On n'omettra pas de parler des conditions aux limites.

1. Que faire quand des particules de fluide rencontrent une frontière ?
2. Et si la frontière n'épouse pas le maillage ?
3. Et si la frontière bouge ?

No-slip boundary condition Rotation de π radians

Slip boundary condition Similaire à la précédente, mais fonctionne comme un miroir (cf Snell-Descartes).

Velocity and Pressure boundary condition von Neumann BC and Dirichlet BC

Smoothed Particle Hydrodynamics

On présentera ici la méthode “Smoothed Particles Hydrodynamics”, qui est une méthode Lagrangienne où le fluide est assimilé à un nombre fini de billes se déplaçant dans un domaine défini.

Approximation d’une fonction Le but de cette sous-section est d’introduire les concepts mathématiques qui seront utilisés dans la suite de cette méthode. On montrera donc comment on peut approximer une fonction, son gradient, son Laplacien etc.

Expression d’une fonction, de son gradient et de son Laplacien

Fonction On montrera que l’on peut approximer $f(x)$ par :

$$\langle f(x) \rangle = \int f(y) W(\|y - x\|, h) dy \quad (C.4)$$

où $W(\|x\|, h)$ une fonction qui tends vers $\delta(x)$ quand h tends vers 0. ($h = \text{supp } W$).

Gradient

$$\langle \nabla f(x) \rangle = \int f(y) \nabla W(\|y - x\|, h) dy \quad (C.5)$$

Laplacien

$$\langle \Delta f(x) \rangle = \int f(y) \Delta W(\|y - x\|, h) dy \quad (C.6)$$

Discretisation dans le cas de particules Étant donné que l’on assimile le fluide à un nombre fini N de petites billes, nous devons discrétiser les expressions que nous avons définies ci-dessus qui sont continues. On montrera donc ici comment sont définis f_i , ∇f_i et Δf_i . On montrera également que certaines propriétés du continu sont perdues (par exemple, que le Laplacien d’une fonction constante n’est plus nécessairement nul, que l’on perd la symétrie, etc) et comment palier ce problème.

Calcul de la position des particules En utilisant ce qui aura été défini avant, on montrera dans cette partie comment calculer la position des billes à l’instant $t + 1$.

Préciser que :

1. Il faut utiliser l’approximation du gradient symétrique pour $\mathbf{f}_p$
2. Il faut utiliser l’approximation du Laplacien qui s’annule pour une fonction constante pour $\mathbf{f}_v$

C.2.3 Applications de ces modélisations

Dans cette troisième et dernière partie, nous présenterons les domaines intéressés par la modélisation des vagues (et/ou fluide plus généralement) :

- Génie civil
- Animation

Les besoins de ces deux domaines diffèrent naturellement :

- l'un a besoin d'un réalisme visuel, pour saisir le spectateur. La moindre gouttelette pouvant rendre l'animation plus vraie est nécessaire. Au détriment d'une physique exacte.
- l'autre au contraire, se fiche du rendu visuel. Ce qui importe est le mouvement du fluide. Est-ce que la structure résiste à la vague, est-ce que le bateau ne se retourne pas si la vague est ainsi. Etc.

Domaines

Pour le génie civil Quelles sont les applications pour le génie civil ? :

1. Étude numérique des efforts sur les structures marines + simulation de la résistance :
 - Plateformes pétrolières
 - Navires
 - Tsunamis
2. Aménagement côtier, érosion

Pour l'animation Depuis que les ordinateurs existent, simuler le réel n'est plus du domaine de l'imaginaire. Aujourd'hui, le rendu est très convaincant, réaliste. Les infographistes/mathématiciens le savent bien et tentent donc de trouver des modèles qui permettent de rendre un univers aussi réel, aussi saisissant que possible. Notamment aujourd'hui, où les films d'animations sont très prisés, tout est fait pour que le spectateur y croit. La modélisation d'un fluide et plus particulièrement des vagues intéresse donc l'animation. On distingue dans ce secteur deux types de simulations : la *simulation en temps réel* et la *simulation enregistrée* *Trouver un autre terme.*

1. La simulation en temps réel → jeux vidéos (moins réaliste) (SPH + utilisation du GPU).
2. La simulation enregistrée → films d'animation (plus réaliste, plus de calculs) (LBM cf : Blender)

Coût de la modélisation

Il s'agira de montrer dans cette sous-section qu'en pratique, simuler un fluide avec réalisme est extrêmement difficile. On montrera ici où sont les difficultés et quels sont les moyens employés pour y palier (GPU + réduction d'échelle). "*Fake it*", est il écrit sur le wiki de Blender.

« In general, it is really hard for current computers to correctly simulate even a 1-meter tank of water. For simulating a wave crashing through a city, you would probably need one of the

most expensive supercomputers you could get, and it might still not work properly, no matter which of the three algorithms above you're using namespace. Therefore, to achieve "the effect that you really want", you'll need to resort to strategies very similar to what filmmakers have been doing (quite successfully...) in "analogue" movies for many years : "fake it!"
» — Wiki de Blender³

ou encore

« For Blender's LBM solver, the following things will make the simulation harder to compute : Large domains. Long duration. Low viscosities. High velocities. »

3. <http://wiki.blender.org/index.php/Doc:2.4/Manual/Physics/Fluid>

BIBLIOGRAPHIE

- Bart ADAMS et Martin WICKE. Meshless approximation methods and applications in physics bases modeling and animation. 2009.
- John BURKARDT. Numerical solution of the shallow water equation. ICAM/Informatique Technology Departement, 2010.
- Kenny ERLEBEN. Smoothed particle hydrodynamics, a short introduction to principles and ideas. University of Copenhagen, 2010.
- Hidemitsu HAYASHI. Lattice boltzmann method and its application to flow analysis in porous media. *R&D Review of Toyota CRDL*.
- Micky KELAGER. Lagrangian fluid dynamics, using smoothed particle hydrodynamics. Rapport technique, Departement of Computer Science. University of Copenhagen, 2006.
- Simon MARIE. *Étude de la méthode Boltzmann sur Réseau pour les simulations en aéroacoustique*. PhD thesis, Université Pierre et Marie Curie, 2008.
- A. A. MOHAMAD. *Lattice Boltzmann Method. Fundamentals and Engineering Application with Computer Codes*. Springer, 2011.
- M. MÜLLER, J. STAM, D. JAMES, et N. THÜREY. Real time physics. class notes. Rapport technique, NVIDIA, Autodesk, Cornell University, ETH Zurich, a.
- Matthias MÜLLER, David CHARYPAR, et Markus Gross. Particle-based fluid simulation for interactive applications. Rapport technique, Departement of Computer Science, Federal Institute of Technology Zürich (ETHZ), Switzerlan, b.
- Liu PENG, Ken ichi NOMURA, Takehiro OYAKAWA, Rajiv K. KALIA, Aiichiro NAKANO, et Priya Vishasta. Parallel lattice boltzmann flow simulation on emerging multi-core platforms. Rapport technique, Collaboratory for Advanced Computing and Simulations, Department of Computer Science, Department of Physics & Astronomy, Department of Chemical Engineering & Materials Science, University of Southern California, Los Angeles, CA 90089-0242, USA.
- Nils THÜREY. *Physically based Animation of Free Surface Flows with the Lattice Boltzmann Method*. PhD thesis, Universtität Erlangen-Nürnberg, 2007.
- Jens WILKE, Thomas POHL, Markus KOWARSCHIK, et Ulrich RUDE. Cache performance optimizations for parallel lattice boltzmann code in 2d. Rapport technique, Friedrich-Alexander-Universität Erlangen-Nürnberg.

LISTE DES FIGURES

1.1	Visualisation d'une trajectoire (Guy PATRY — LAL 1.3 — http://www.freemages.fr/browse/photo-1457-voie-berge-nuit.html)	6
1.2	Description Eulérienne du vent en France	7
2.1	Shallow water 1D (BURKARDT 2010)	16
2.2	Maillages couramment utilisés	17
2.3	Visualisation de la méthode appliquée sur une maille (MÜLLER et al. a)	18
2.4	Illustration des différentes étapes du processus (THÜREY 2007)	20
2.5	LBM D_2Q_9	21
2.6	Illustration SPH tirée de ADAMS et WICKE (2009)	26
2.7	Simulation d'un barrage qui rompt	27
3.1	Schématisation de la décomposition spatial d'un domaine Ω . (PENG et al.)	30
3.2	Technique de <i>grid-merging</i> (WILKE et al.)	30
3.3	Représentation d'un arbre-kd (http://en.wikipedia.org/wiki/K-d_tree)	31
3.4	Calcul de $\mathcal{N}(p_i)$. Seules les cellules violettes sont lues. (ADAMS et WICKE 2009)	32
3.5	Structures de recherche (ADAMS et WICKE 2009)	34
C.1	Maillages couramment utilisés	48

Titre La modélisation des vagues : pourquoi, comment ?

Résumé La modélisation mathématique des mouvements d'un fluide a été de tous temps un sujet de recherche actif. Quels sont aujourd'hui les moyens utilisés pour rendre numériquement compte de ces mouvements ? Quels sont les coûts de calculs et comment peut-on les améliorer ? Tant de questions auxquelles ce document tente de répondre en présentant quelques méthodes mathématiques utilisées par l'animation et le génie civil.

Mots-clés simulation numérique, fluide, Smoothed Particle Hydrodynamics, Lattice Boltzmann Method, Shallow Water, méthode particulière