

UNIVERSITÉ DE LIMOGES
ÉCOLE DOCTORALE « Science, Technologie, Santé »
FACULTÉ DES SCIENCES ET TECHNIQUES

Thèse N° 532008

THÈSE
pour obtenir le grade de
DOCTEUR DE L'UNIVERSITÉ DE LIMOGES
Discipline / Spécialité : **Informatique**
présentée et soutenue par
Jean-François EL HAJJAR
le 23 octobre 2008

**Simulation d'écoulements liquides sur des surfaces
solides pour l'animation en synthèse d'images**

*Thèse dirigée par Djamchid Ghazanfarpour et Xavier Pueyo,
Co-encadrée par Vincent Jolivet*

JURY

Rapporteurs :

M. Jean-Michel DISCHLER

Professeur à l'Université Louis Pasteur de Strasbourg

M. Robert JOAN-ARINYO

Professeur à Universitat Politècnica de Catalunya

Examineurs :

M. Djamchid GHAZANFARPOUR

Professeur à l'Université de Limoges

M. Xavier PUEYO

Professeur à l'Universitat de Girona

M. Vincent JOLIVET

Maître de conférences à l'Université de Limoges

Remerciements

Je tiens à témoigner de ma reconnaissance envers mes directeurs de thèse, les professeurs Djamchid Ghazanfarpour et Xavier Pueyo, pour m'avoir donné l'opportunité d'effectuer cette thèse en cotutelle avec l'Espagne sous leurs directions ainsi que pour la grande liberté qui m'a été octroyée relativement à mes orientations et travaux de recherche durant ces trois dernières années.

Mes sincères remerciements s'adressent aux professeurs Jean-Michel Dischler et Robert Joan-Arinyo pour avoir accepté la charge de rapporteur, avoir pris de leurs temps afin de lire ce manuscrit ainsi que pour les conseils et encouragements qu'ils m'ont tous deux prodigués lors de notre rencontre.

Je ne saurai jamais assez démontrer ma profonde gratitude ainsi que ma considération envers Frédéric Saubion, sans qui, mon parcours universitaire et ma vision de l'enseignement supérieur ne seraient ce qu'ils sont aujourd'hui.

A mes amis, pour votre présence.

A ma petite sœur, Caroline, pour la place que tu occupes dans ma vie.

A ma mamie et à ma mère, pour votre perpétuel soutien tant moral que matériel.

A mon papy.

Résumé

Dans le domaine de la synthèse d'images où une branche se focalise sur la reproduction du réel par ordinateur, les phénomènes naturels (tels que le vent, l'océan, le feu et la fumée) occupent une place importante dus à leurs omniprésences dans la nature et donc dans la vie de tous les jours. Le phénomène des gouttes n'est pas exempt de cet intérêt, ces faibles quantités de liquide se manifestant au quotidien sous diverses formes (la rosée, la pluie, jet de douche, chute d'un glaçon dans une cocktail etc.). En dépit de la faible échelle à laquelle ce phénomène intervient, le phénomène des gouttes ne peut être négligé en synthèse d'images si perception, photo-réalisme ou immersion est de rigueur. Un flot de gouttes se comporte de façon étonnamment complexe, la forme ainsi que le mouvement adoptés par une goutte étant riches en possibilités et souvent décrit comme difficilement prévisible.

Dans le cadre de cette thèse, deux méthodes pour l'animation de gouttes en synthèse d'images ont été explorées. Afin de cerner au mieux le phénomène des gouttes et recenser ses caractéristiques prédominantes d'un point de vue phénoménologique, une étude physique préalable (succincte pour un physicien mais suffisante pour l'infographiste) a été nécessaire. A partir de la compréhension du phénomène au sens de la physique ainsi que de l'analyse de précédentes méthodes en relation avec la simulation d'un flot de gouttes pour la synthèse d'images, des limitations ont pu être dégagées pour donner lieu au développement de deux méthodes distinctes conceptuellement et dans leur intérêt.

La première méthode est une méthode empirique temps-réel pour l'animation de faibles écoulements liquides évoluant sur une surface plane. La principale motivation derrière cette méthode a été le niveau de performance, qui en conséquence nous a amené à effectuer des choix quant au degré de complexité de de complétude de la prise en compte du phénomène. Cette approche est destinée à intégrer des applications plus générales où les performances priment sur le réalisme comme c'est le cas dans un simulateur de conduite ou dans un jeu vidéo.

Afin de simuler de façon plus convaincante et réaliste le phénomène des gouttes, une seconde méthode basée physique a été explorée. Comme il est généralement le cas en synthèse d'images, ce type d'approche permet d'obtenir des résultats de grandes qualités mais en temps différé du à leurs complexités. En s'inspirant des travaux du domaine de la physique

numérique (fortement actif et d'intérêts variés) tout en partant d'une analyse rigoureuse de l'unique méthode basée physique pour l'animation de gouttes sur des surfaces en synthèse d'images, on propose dans cette seconde méthode de lever certaines limitations en rapport tant avec les complexités calculatoire et mémoires qu'avec la justesse phénoménologique des résultats visuels.

On clôturera ce manuscrit en présentant les perspectives qu'ont engendrées le développement de ces deux méthodes respectives en présentant de futures pistes de recherche.

Mots-clés : synthèse d'images, phénomènes naturels, gouttes, temps-réel, Navier-Stokes, tensions de surface, triple ligne de contact

Abstract

In computer graphics where a part of the community focuses on artificially reproducing nature by the mean of computers, natural phenomena (such as wind, ocean, fire and smoke) are of great importance due to their omnipresence in every day's life. In this scope, droplets are no exception as one can observe them in many occasions and many forms (dew, rain, shower burst, ice cube falling in a cocktail etc.). Despise the small scale at which the phenomenon occurs, droplets can not be neglected in computer graphics if perception, photo-realism or immersion are seeked. A droplet flow will behave in a suprising complex way, its shape as well as its dynamic being rich in possibilities and oftenly described as unpredictable.

In the scope of this thesis, two methods for droplets animation in computer graphics have been explored. In order to understand the essence of the phenomenon and to list the predominant phenomenological characteristics, a preliminary study in physics (brief for the physicist but suffisant for the infographist) has been necessary. From the understanding of the phenomenon in the sens of the physics as well as the analyse of previous works dedicated to on-surface flows in computer graphics, some limitations have been emphasized which in turn have led to the development of two novel methods different in both concept and scope.

The first method is empirical and performs in real-time, it tackles the problem of on-surface flows on planar surfaces. The principal motivations behind this work was the performance issues, which in consequence lead us to pose some hypotheses as well as to consider some simplifications relatively to the phenomenon. This method is aimed at being part of critical time applications where performances are prefered over realism such as in a drive simulator or a video game.

In order to animate in a more convincing and realistic way the flow of the droplets, a second method physically based has been explored. As it is in general the case in computer graphics, such methods allows one to attain a great degree of realism at the price of a high computational cost, thus dedicating this type of methods to offline rendering. By inspiring ourselves from the litterature present in the computational fluid domain as well as thoroughly analysing the only method physically based in computer graphics for the animation of droplets on surfaces, we propose in this second method to overcome some

limitations with respect to the computational and memory complexity as well as the correctness of the shape of the flow.

In conclusion, we present the perspectives that have arisen from the development of these two methods.

Keywords : computer graphics, natural phenomena, droplets, real-time, Navier-Stokes, surface tensions, triple contact line

Table des matières

Introduction	1
1 Gouttes : de l’aspect phénoménologique au rendu	5
1.1 Un aperçu de la physique des gouttes	6
1.1.1 Tension de surface	6
1.1.2 Discontinuité de la pression	7
1.1.3 Contact à trois phases	8
1.2 Modélisation, animation et rendu de gouttes	12
1.2.1 Approches empiriques	13
1.2.2 Méthodes basées physique	21
2 Une approche temps-réel pour les écoulements surfaciques	39
2.1 Introduction	39
2.2 Description de la méthode	41
2.2.1 Modélisation de la surface et des obstacles	41
2.2.2 Impact de gouttes sur la surface	42
2.2.3 Dynamique du flot de gouttes	42
2.2.4 Effets Spéciaux	48
2.3 Implémentation	50
2.3.1 Simulation	50
2.3.2 Rendu	52
2.4 Resultats	57
2.5 Conclusions	61
3 Une approche basée physique pour l’animation de gouttes sur des surfaces arbitraires	63
3.1 Equations de Navier et Stokes	64
3.1.1 Description	64
3.1.2 Discrétisation des équations et des interfaces	66
3.1.3 Résolution numérique	73
3.1.4 Tensions interfaciales	80
3.1.5 Tensions de surface à la triple ligne de contact	84
3.2 Modèles proposés	89

3.2.1	Dynamique de la triple ligne de contact	89
3.2.2	Estimation des sauts de pression à la triple ligne de contact	89
3.3	Etat des travaux et Conclusion	96
	Conclusion	103
	Bibliographie	107

Table des figures

1	Diverses gouttes dans la nature	2
(a)	Une goutte se détachant d'un robinet	2
(b)	Une feuille flottant à la surface de l'eau	2
(c)	De la rosée sur une toile d'araignée	2
(d)	Gouttes sur une plante	2
(e)	Un splash d'eau	2
1.1	Défaut d'énergie à l'interface entre un liquide et un gaz	6
1.2	Théorème de Laplace : illustration de la discontinuité de la pression à une interface	7
1.3	Étalement d'un liquide sur une surface solide	8
1.4	Relation de Young : illustration de l'hystérésis de l'angle de contact	9
1.5	Dynamique de la triple ligne de contact	11
1.6	Quelques animations pionnières en synthèse d'images pour de larges étendues d'eaux.	12
(a)	Résultat extrait de [Per85]	12
(b)	Résultat extrait de [FR86]	12
(c)	Résultat extrait de [Lew87]	12
1.7	Différents résultats obtenus par Kaneda <i>et al.</i>	13
(a)	Résultat extrait de [KKY93]	13
(b)	Résultat extrait de [KZYN96]	13
(c)	Résultat extrait de [KIY99]	13
1.8	Résultats obtenus par Sato <i>et al.</i> [SDY02]	16
(a)	Sans effet de champs de profondeur	16
(b)	Avec effet de champs de profondeur	16
1.9	Résultats obtenus par Yu <i>et al.</i> [YJC98]	17
(a)	Fusion de deux gouttes	17
(b)	Plusieurs gouttes soumises à la gravité	17
1.10	Résultats obtenus par Fournier <i>et al.</i> [FHP98]	18
(a)	Modélisation	18
(b)	Rendu	18
(c)	Animation	18
1.11	Résultats obtenus par Murta et Miller [MM99]	19

1.12	Résultats obtenus par Yang <i>et al.</i> [YZZ04]	20
	(a) Distortion du maillage	20
	(b) Rendu des gouttes	20
1.13	Résultats obtenus par Kass et Miller [KM90]	23
1.14	Résultats obtenus par Chen et Lobo [CdVLHM97]	24
1.15	Résultats obtenus par Foster et Metaxas [FM96]	25
1.16	Résultats obtenus par Foster et Metaxas [FM97]	25
1.17	Résultats obtenus par Foster et Fedkiw [FF01]	27
1.18	Comparaison des méthodes [FF01] et [EMF02]	28
	(a) Résultat extrait de [FF01]	28
	(b) Résultat extrait de [EMF02]	28
1.19	Résultats obtenus par Hong et Kim [HK03]	29
1.20	Résultats obtenus par Losasso <i>et al.</i> [LGF04]	30
	(a) sans tensions de surface	30
	(b) avec tensions de surface	30
1.21	Résultats obtenus par Greenwood et House [GH04]	30
1.22	Résultats obtenus par Hong et Kim [HK05]	32
	(a) Un jet liquide fin	32
	(b) Formation de bulles sous l'eau	32
1.23	Résultats obtenus par Wang <i>et al.</i> [WMT05]	33
1.24	Résultats obtenus par Stam [Sta03]	35
1.25	Résultats obtenus par Shi et Yu [SY04]	35
1.26	Résultats obtenus par Liu <i>et al.</i> [LZLW05]	36
	(a) Une goutte d'eau sur une surface	36
	(b) Un flot liquide sur une surface	36
1.27	Résultats obtenus par Wang <i>et al.</i> [WMT07]	37
2.1	Perturbation du vecteur accélération selon la texture de normales	44
2.2	Tracés de filtres d'atténuation utilisés pour simuler le paramètre de viscosité et les forces de friction	45
2.3	Illustration de l'advection d'un volume liquide vers ses cellules avoisinantes	47
2.4	Schéma du lancer de rayons basé image utilisé pour le calcul d'intersections avec la surface liquide avec prise en compte de la réfraction et de la réflexion	52
2.5	Résultat : une ville vue de nuit derrière une vitre mouillée	54
2.6	Résultat : flot de gouttes non perturbé au dessus de la peinture Nishiki avec mise en valeur de la réfraction	56
2.7	Résultat : animation où une ville est vue de nuit à travers une vitre sur laquelle il pleut de plus en plus	58
2.8	Résultat : animation d'un flot liquide au dessus de la peinture Nishiki dé- nuées de perturbations	58

2.9	Résultat : animation d'un flot liquide au dessus de la peinture Nishiki avec perturbations	59
2.10	Résultat : animation d'un flot liquide sur une peinture de Mona Lisa heureuse avec un effet d'absorption par atténuation	60
2.11	Résultat : animation d'un flot liquide sur une peinture de Mona Lisa de style cubique teintée de vert avec un effet d'absorption par apparition d'une Mona Lisa heureuse	60
2.12	Résultat : animation d'un flot liquide sur un motif de nuit étoilée avec un effet de transport d'encre	61
3.1	Domaine de résolution Ω pour les équations de Navier et Stokes	66
3.2	Discrétisations possibles pour une interface	69
3.3	Grille 3D à arrangement delocalisé	74
3.4	Illustration d'une advection semi-lagrangienne	75
3.5	Illustration du concept de la méthode des fluides fantômes (GFM)	82
3.6	Approximation de la courbure par deux sphères d'axes principaux orthogonaux	92
3.7	Approximation de la courbure par une quadrique	94
3.8	Résultat : Splash d'eau sans effets de tensions de surface	97
3.9	Résultat : Splash d'eau avec effets de tensions de surface	98
3.10	Résultat : Bulle d'air prisonnière dans de l'eau remontant à la surface sans effets de tensions de surface	100
3.11	Résultat : Bulle d'air prisonnière dans de l'eau remontant à la surface avec effets de tensions de surface	101

Introduction

D'un point de vue lexical, une goutte se définit simplement comme étant *une petite quantité de liquide*. Phénoménologiquement parlant, il en est tout autre. Une goutte sera de forme sphérique à l'équilibre, s'aplatira légèrement lorsqu'en chute libre, ondulera inlassablement le long d'une trajectoire courbe, se séparera en deux si son volume devient trop important et fusionnera avec sa voisine lors d'une collision. Lors d'une interaction avec une surface solide, une goutte oscillera avant de former un angle de contact stable à sa frontière commune avec le support, se mettra en mouvement à partir d'une certaine inclinaison seuil de la surface, favorisera les chemins préalablement mouillés par ses semblables et s'immobilisera lorsqu'elle aura laissée derrière elle une trop large traînée.

Riche et complexe, souvent décrit comme magnifique dans la littérature, le phénomène des gouttes aux origines diverses et variées peut être observé au quotidien (voir Figure 1) : la condensation de vapeur d'eau résultant en de la rosée ou de la pluie, un splash provoqué par la chute d'un objet dans une étendue liquide, un robinet qui fuit, de l'huile plongée dans un récipient d'eau remontant à la surface sous forme de gouttes etc. En dépit de l'échelle macroscopique du phénomène, les gouttes interagissent avec notre milieu de façon non négligeable et contribuent ainsi à la compréhension de notre environnement (par exemple, un matériau est perméable si une goutte à sa surface est absorbée) tout en modifiant occasionnellement notre sens de la perception (par exemple, visibilité réduite lors de conduites par temps pluvieux).

En synthèse d'images, et plus particulièrement dans une optique de photo-réalisme, la création d'images assistée par ordinateur mettant en scène des phénomènes naturels est considérée par la communauté comme étant d'un intérêt inhérent au domaine. De par son apport, elle étend l'univers des scénarios possibles tout en les augmentant de réalisme et de beauté, et elle constitue jusqu'à ce jour un réel défi de par la difficulté liée à la compréhension de la nature et donc, à sa formalisation.

Les gouttes appartenant à la vaste famille des phénomènes naturels, des techniques dédiées propres à la synthèse d'images ont ainsi été développées, ces dernières ayant trait tant à la modélisation, à l'animation qu'au rendu. Les intérêts de telles méthodes sont multiples, on citera à titre d'exemples les films d'animations (pour un réalisme accrue), les jeux vidéos (pour une meilleure immersion) ou les simulateurs de conduite (pour une



(a)



(c)



(d)



(b)



(e)

FIG. 1: Des gouttes dans la nature : une goutte s'échappant d'un robinet (a), de la rosée sur une toile d'araignée (c), des gouttes sur une plante (d) et un splash (e).

modification de la perception environnementale).

Comme il est fréquent dans le cadre de méthodes dédiées à l’animation de phénomènes naturels, on peut distinguer trois principales classes d’approches valables pour la présente problématique des écoulements surfaciques. La première est l’empirisme où une méthode est construite intuitivement en fonction du résultat désiré, ce dernier dans le cas présent étant généralement issu d’observations phénoménologiques car présent dans la nature et donc non arbitraire. La nature étant complexe et difficile à cerner comportementalement parlant, une approche empirique posera souvent des hypothèses tout en limitant l’éventail des phénomènes à reproduire afin de simplifier un vaste objectif en un problème partiellement résolvable par l’intuition. La seconde est dite basée images où l’on tente d’extraire des propriétés intrinsèques à un phénomène à partir de données existantes issues de la nature pour finalement les formaliser (par exemple, procéduralement ou spectralement). Ce type d’approche est efficace lorsque les données analysées possèdent de fortes propriétés récurrentes. La troisième et dernière classe d’approches se base sur les équations physiques décrivant un phénomène naturel où des animations sont obtenues par résolution numérique.

Dans les présents travaux, on s’intéresse à l’animation de faibles écoulements surfaciques pour laquelle deux pistes ont été explorées. Dans un soucis de performances, une approche empirique a été dégagée qui est décrite dans le chapitre 2. Le modèle présenté se prêtre à des performances temps-réel tout en prenant en compte plusieurs paramètres physiques liés aux liquides, certaines simplifications ayant été en revanche nécessaires faisant de cette approche une approche non générale mais plutôt destinée à un certain type d’applications (jeux vidéos, simulateurs de conduite). Dans un soucis de réalisme, une approche basée physique a été développée faisant l’objet du chapitre 3. Nous verrons comment les équations générales de l’hydrodynamique peuvent être modifiées afin que leurs résolutions numériques permettent d’obtenir des animations réalistes. En contrepartie, les longs temps de calculs requis fait de cette méthode une méthode pour l’animation en temps différé. Etant donné qu’il a fallu effectuer des choix quant aux orientations à prendre au début de ces travaux de recherche, il a été jugé qu’une approche basée image était moins appropriée que ses consœurs dans une optique d’animation d’écoulements surfaciques, le caractère aléatoire et imprévisible d’un flot de gouttes en étant la raison majeure. On n’exclut pas pour autant d’explorer une telle piste à l’avenir.

Chapitre 1

Gouttes : de l'aspect phénoménologique au rendu

Avant de nous focaliser sur les précédents travaux en synthèse d'images relatifs à la thématique des gouttes, il a été jugé utile et nécessaire d'effectuer au préalable une description du phénomène des gouttes au sens des sciences physiques, les intérêts étant multiples :

La physique, étymologiquement du grec « science de la nature », s'applique depuis plusieurs siècles à observer les phénomènes naturels, les mesurer pour finalement en extraire une modélisation par le biais des mathématiques. A travers l'observation sont recensées les principales caractéristiques du phénomène décrivant son comportement, et détermine ainsi un problème bien posé. Cette première étape est commune à l'algorithmique où l'on se doit de déterminer quelles règles devront être ériger dans le but d'obtenir le résultat voulu. Par exemple, une goutte se détachera ou non d'un solide en fonction de l'orientation et de la grandeur de la force qui lui est appliquée. Par les mesures se dégagent des valeurs numériques (souvent des constantes) permettant de classifier un phénomène selon sa nature, et peuvent constituer des paramètres d'entrée à un algorithme. Par exemple, la viscosité d'un fluide est un coefficient décrivant la propriété de résistance à l'écoulement uniforme, et on observe dans la réalité qu'à conditions égales une goutte de miel (forte viscosité) se déplacera plus lentement qu'une goutte d'eau (faible viscosité). Finalement, de la modélisation mathématique découle des équations différentielles dont la résolution permet de tracer l'évolution du phénomène dans le temps. Bien que dans le cas des fluides, ces dernières soient complexes au point de ne pas avoir de solution générale analytique, il existe aujourd'hui pléthore de méthodes d'intégration numérique permettant une résolution discrète du système. Comme on le verra par la suite, la physique numérique est aujourd'hui couramment mise à contribution dans le domaine de la synthèse d'images, et constitue une méthode de solution à part.

Le chapitre présent est organisé de la façon suivante. Dans une première partie, une description physique principalement basée sur les ouvrages [dG85, dGBWQ05] sera établie, cette dernière étant volontairement restreinte quantitativement en vue des besoins du

domaine de la synthèse d'images. Cette partie permettra de répondre notamment à deux questions essentielles :

- Que cherche-t-on à reproduire en synthèse d'images ? On pourra ainsi mieux cerner les contributions de la communauté ainsi que leurs limitations actuelles, pour finalement dégager les problèmes demeurant ouverts.
- Quelles sont les raisons derrière le phénomène ? De par ses éléments de réponse, on sera en mesure de mieux évaluer les approximations effectuées, la validité des hypothèses posées ou la qualité des heuristiques utilisées lors de précédents travaux.

Dans une seconde partie, on passera en revue les contributions significatives apportées par la communauté de la synthèse d'images ces dernières années, chacune étant pourvue d'une analyse succincte mais suffisante à sa compréhension globale. Finalement, on se fixera des objectifs en ciblant certaines limitations afin de les dépasser, objectifs qui sont à la base des motivations des travaux de recherche présentés dans ce manuscrit.

1.1 Un aperçu de la physique des gouttes

Née au début du XIX^{ème} siècle, la capillarité désigne l'étude des surfaces de séparation (dites interfaces) existant entre deux liquides non miscibles ou entre un liquide et un gaz. La capillarité permet entre autres d'expliquer la forme des bulles et des gouttes, la stagnation d'une petite goutte sur une fenêtre, la formation de menisques, et a fait l'objet d'une formalisation succinctement décrite dans ce qui suit.

1.1.1 Tension de surface

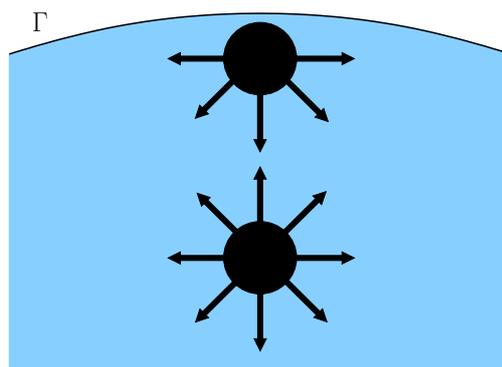


FIG. 1.1: Deux molécules d'eau sont représentées avec leurs interactions respectives sous forme de flèches. La molécule plongée dans le liquide est en interaction avec son voisinage complet alors que la molécule à l'interface Γ perd environ la moitié de ses interactions. Ce phénomène est à la base du défaut d'énergie communément observé à l'interface, et est à l'origine de la tension superficielle.

A l'échelle microscopique, un liquide se compose d'un ensemble de molécules, chacune

étant en interaction avec ses voisines. Cet équilibre est rompu à la surface du liquide où les molécules limitrophes sont moins sollicitées par leur voisinage (voir Figure 1.1), ce qui se traduit par un défaut d'énergie du système. Afin de palier à cette perte énergétique, un liquide aura alors tendance à minimiser sa surface libre, et l'on appelle tension superficielle (respectivement, tension interfaciale) γ la mesure du défaut d'énergie par unité de surface dans le cas d'une interface liquide-gaz (respectivement, une interface liquide-liquide).

Conceptuellement, γ représente l'énergie à fournir pour augmenter la surface d'une unité [$J \cdot m^{-2}$], ou de façon équivalente, la force à appliquer le long de la normale à l'interface pour déformer la surface à l'équilibre [$N \cdot m^{-1}$]. Cette constante mesurée expérimentalement se notera $\gamma_{L_1 L_2}$ dans le cas d'une tension interfaciale entre deux liquides L_1 et L_2 , et γ_{L_1} dans le cas d'une tension superficielle entre un liquide L_1 et un gaz.

Afin de rendre plus aisée la lecture du manuscrit, on utilisera dorénavant le terme plus générique de tension de surface en sus des termes de tension superficielle et de tension interfaciale.

1.1.2 Discontinuité de la pression

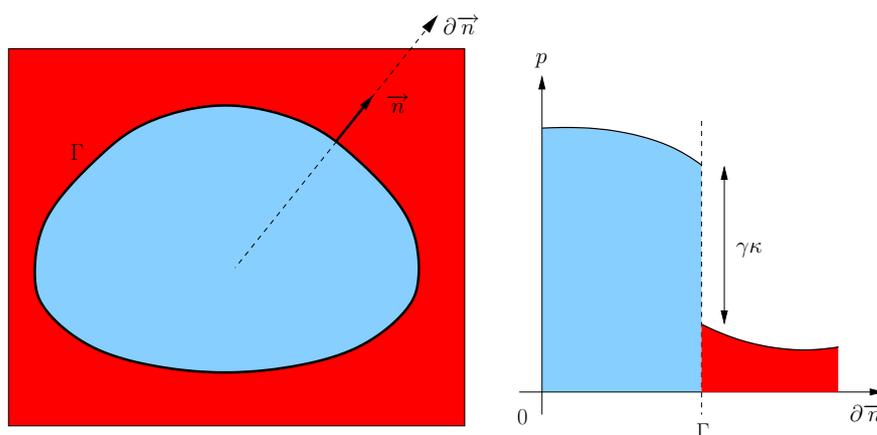


FIG. 1.2: A gauche, une goutte d'eau (en bleu) en contact avec de l'air (en rouge), l'interface étant dénotée Γ . A droite, mesure de la pression p le long de l'axe de la normale $\partial\vec{n}$ pour un point fixe de l'interface. La loi de Laplace est illustrée par le saut de pression d'intensité $\gamma\kappa$ intervenant à la transition interfaciale.

Lorsque l'on traverse une interface séparant deux fluides, il existe une discontinuité dans le champs de pressions due à la présence de la tension de surface. Laplace formalise ce phénomène en 1805 par un théorème qui décrit le saut de pression à l'interface comme étant proportionnel à la courbure de l'interface (voir Figure 1.2) :

$$\Delta p = \gamma\kappa$$

où κ dénote la courbure moyenne au sens de la mécanique des fluides et non de la géométrie différentielle. Soit, si R_1 et R_2 sont respectivement les rayons de courbures dans les

deux directions principales en un point de l'interface, alors les deux courbures principales sont $\kappa_1 = \frac{1}{R_1}$ et $\kappa_2 = \frac{1}{R_2}$ et la courbure moyenne est définie par $\kappa = \kappa_1 + \kappa_2$.

Le théorème de Laplace permet d'expliquer l'origine de la surpression qui règne au sein de gouttes et de bulles, et ainsi comprendre l'adhésion capillaire entre deux plaques ou pourquoi une petite bulle fusionnera avec sa voisine plus volumineuse par exemple.

1.1.3 Contact à trois phases

Lorsqu'une interface fluide-fluide entre en contact avec un solide, il existe une ligne à laquelle les trois phases se recoupent : cette interface particulière se dénomme la triple ligne de contact, le mouillage étant l'étude de l'étalement d'un liquide sur un substrat solide.

Angle de contact stable

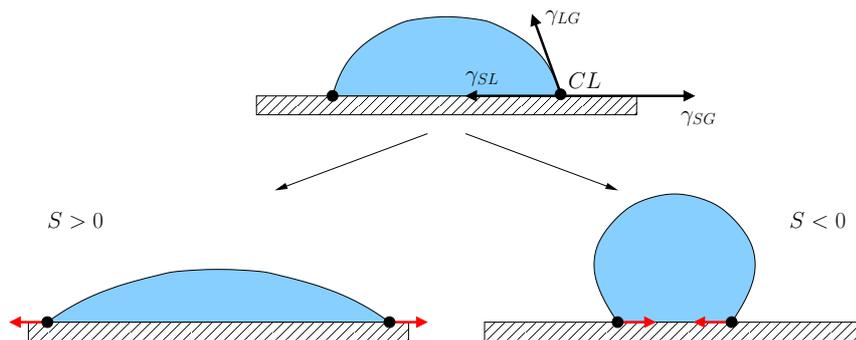


FIG. 1.3: En haut, une goutte liquide posée sur un solide est soumise aux trois tensions de surface à sa triple ligne de contact CL . Si le paramètre d'étalement S est positif, on dit alors que la surface est hydrophile et on assiste à un mouillage total de la surface (en bas, à gauche). Inversement, si le paramètre d'étalement S est négatif, on dit alors que la surface est hydrophobe et on assiste à un mouillage partiel de la surface (en bas, à droite).

Considérons une quantité liquide déposée sur une surface solide. Selon la nature de ces deux derniers, on assistera à un mouillage partiel (on dira alors que la surface est hydrophobe) ou à un mouillage total (on dira alors que la surface est hydrophile) du liquide. La nature du mouillage est décrite par le paramètre d'étalement S qui est fonction des tensions de surface agissant à la triple ligne de contact γ_{SG} , γ_{SL} et γ_{LG} , et on pose $S = \gamma_{SG} - (\gamma_{SL} + \gamma_{LG})$.

Ainsi, comme illustré dans la Figure 1.3, si :

- $S > 0$, γ_{SG} prédomine et on assiste à un étalement du liquide sur le solide.
- $S < 0$, les tensions de surface liquides γ_{SL} et γ_{LG} prédominent et le liquide se rétracte sur lui-même.

A l'équilibre, soit lorsque $S = 0$, l'angle formé par le liquide avec le solide est constant et se dénote θ_E . En projetant l'équilibre des forces capillaires sur le plan solide, on obtient alors la relation de Young (1805) :

$$\gamma_{SG} - (\gamma_{SL} + \gamma_{LG} \cos \theta_E) = 0$$

En reportant l'expression précédente de γ_{SG} dans la formulation initiale de S , on obtient de façon équivalente :

$$\gamma_{LG}(\cos \theta_E - 1) = 0$$

La relation de Young est importante dans le sens où elle permet de caractériser un équilibre capillaire à partir de deux constantes, γ_{LG} et θ_E , en lieu de trois constantes, γ_{LG} , γ_{SG} et γ_{SL} . Pour un couple de fluides et une surface donnés, l'angle de contact à l'équilibre θ_E est obtenu par mesure expérimentale et décrit complètement le balancement des forces à l'équilibre en complémentarité γ_{LG} . Il est ainsi possible de se passer des mesures γ_{SG} et γ_{SL} , ce qui présente des avantages notamment lorsque ces dernières sont sujettes à du bruit et donc plus problématiques à estimer en laboratoire.

Hystérésis de l'angle de contact

Bien que la relation de Young formalise idéalement le phénomène de la triple ligne de contact, les mesures expérimentales obtenues en laboratoire validant le modèle équationnel lorsque des conditions optimales sont atteintes (liquide « pur » posé sur une surface solide « parfaitement » plane et propre), les observations phénoménologiques diffèrent néanmoins quelque peu de l'aspect théorique.

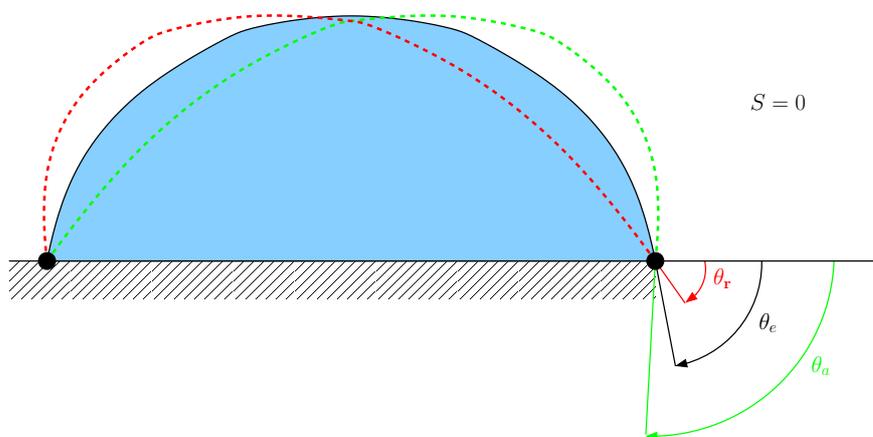


FIG. 1.4: Hystérésis de l'angle de contact : une goutte peut former un angle de contact stable θ_s avec une surface solide tel que $\theta_r \leq \theta_s \leq \theta_a$, sa triple ligne de contact demeurant statique.

En effet, dans la nature, il est quasiment pour ne pas dire impossible de retrouver les conditions optimales précédemment considérées. En conséquence, ces dernières sont des hypothèses trop fortes qui doivent être relaxées en vue d'être en meilleur agrément avec les observations phénoménologiques. Considérons une surface solide horizontale sur laquelle est posée une petite goutte d'eau en équilibre capillaire (soit, la goutte est immobile et l'angle de contact formé avec le solide est stable). Si l'on incline progressivement la surface solide, on observe que la goutte ne décroche du solide sous l'effet de la gravité qu'à partir d'une certaine inclinaison seuil, l'angle de contact décrivant entre-temps un petit éventail de valeurs. En d'autres termes, l'angle de contact stable θ_s ne peut se réduire à une valeur unique θ_E , mais admettra plutôt un intervalle de valeurs borné par deux angles limites caractéristiques : l'angle minimal θ_r dit « angle de reculée » et l'angle maximal θ_a dit « angle d'avancée », avec l'angle idéal à l'équilibre $\theta_e \in [\theta_r, \theta_a]$.

Ce phénomène communément appelé hystérésis de l'angle de contact (voir Figure 1.4) s'explique par le fait que toute surface solide présente des défauts chimiques ou physiques à partir d'une certaine échelle, et peut être observé dans la nature dans de nombreuses circonstances, d'où son importance.

Dynamique de la ligne triple de contact

Le déplacement de la triple ligne de contact sur une surface solide est certainement à ce jour le phénomène le moins bien compris en physique des gouttes, et son étude a même permis de corriger une importante singularité qui était communément admise en mécanique des fluides ¹.

Bien qu'il n'existe pas de formalisation générale universellement admise mettant en rapport les différentes propriétés des fluides et du solide (densité, viscosité, friction etc.) avec le déplacement de la triple ligne de contact pour une configuration donnée, la communauté physique a néanmoins proposé plusieurs modélisations empiriques du phénomène basées sur les observations suivantes :

- l'angle de contact dynamique θ_d est toujours compris entre l'angle de reculée θ_r et l'angle d'avancée θ_a .
- si l'interface se déforme (par exemple, sous l'effet de la gravité) tel que l'angle de contact dynamique dépasse l'angle d'avancée θ_a , alors la triple ligne de contact sera soumise à une accélération normale à l'interface, cela afin de recouvrir l'équilibre capillaire.
- inversement, si l'angle de contact dynamique devient inférieur à l'angle de reculée θ_r , alors la triple ligne de contact sera soumise à une accélération opposée à la normale de l'interface afin de recouvrir l'équilibre capillaire.

¹Cette singularité relative aux conditions limites des vitesses du couple de fluides au bord de la surface solide sera décrite lorsque les simulations numériques seront abordées.

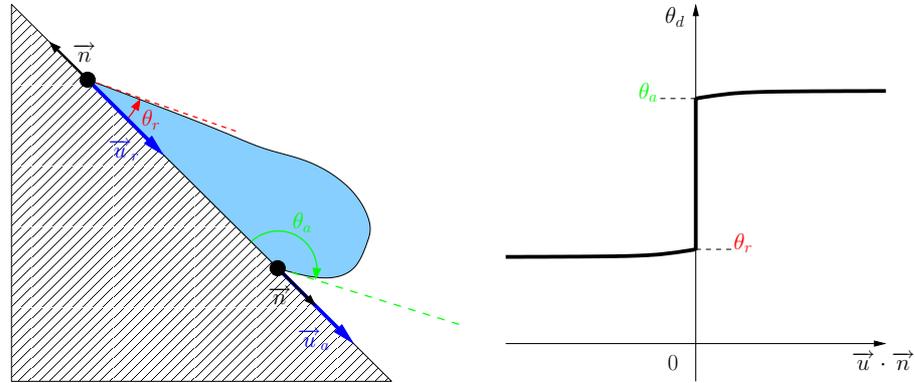


FIG. 1.5: A gauche, une goutte d'eau glisse le long d'une surface solide inclinée. La valeur de l'angle de contact θ_d dépend de la vitesse normale à l'interface, et $\theta_r \leq \theta_d \leq \theta_a$ comme schématisé sur la droite.

La Figure 1.5 illustre une triple ligne de contact en mouvement ainsi que la corrélation entre l'angle de contact et la vitesse à l'interface. On notera que le schéma de droite décrit non seulement la dynamique de la triple ligne de contact, mais aussi l'hystérésis de l'angle de contact où à l'équilibre (soit, pour une vitesse nulle) il est permis à θ_d d'osciller entre θ_r et θ_a .

Si l'on considère une surface solide où deux gouttes sont à proximité, on remarquera le phénomène tant banal qu'étonnant de l'attraction entre ces deux dernières, la plus petite goutte se déversant dans sa voisine avant fusion totale. Bien qu'en apparence non corrélé avec le phénomène de l'angle de contact, ce comportement s'explique dans un premier temps par l'existence d'un film précurseur microscopique se développant au delà de la triple ligne de contact. Une surface devenant plus hydrophile lorsque humide, la valeur de l'angle de contact stable devient dans ce cas inférieure à sa valeur initiale sur surface sèche. Le film précurseur (non visible à l'oeil nu) en périphérie des gouttes a pour conséquence de provoquer une diminution nette de l'angle de contact lorsque qu'une goutte est relativement proche d'une de ses consœurs, infligeant ainsi un déplacement net de la triple ligne de contact dans le sens de la normale. Les deux liquides fusionnent alors en une seule goutte initialement en déséquilibre énergétique, les tensions de surface corrigeant ce dernier afin de finalement dicter la forme de la nouvelle goutte.

Les différents modèles élaborés dans le cadre de la présente problématique étant destinés à un usage au sein de simulations numériques expérimentales, on les omettra volontairement dans la présente partie qui se veut théorique pour y revenir en détail ultérieurement lorsque des méthodes numériques basées physiques seront étudiées.

1.2 Modélisation, animation et rendu de gouttes

Au début des années 1980, des méthodes pionnières en matière de rendu de liquides ont été proposées en abordant le problème initial des grandes étendues d'eau pour l'intérêt trivial qu'il représente en synthèse d'images. En constatant qu'un liquide se distingue en partie d'un solide de par les fortes variations d'intensité lumineuse qui surviennent à sa surface dues à une géométrie irrégulière (petites vagues, ondes etc.), la technique encore récente du « bump-mapping » [Bli78] fut mise à contribution dans ce cadre spécifique. En considérant la surface de l'eau comme un plan, une texture bidimensionnelle est utilisée afin de modifier l'intensité des normales à la surface selon une fonction de perturbation judicieusement déterminée et généralement basée sur des sinusoïdes ou du bruit [Sch80, Per85]. La géométrie planaire sous-jacente limitant sérieusement les domaines d'application des approches précédentes (déferlement, réfraction et atténuation de la lumière, point de vue fortement contraint etc.), des modèles paramétriques issus de l'hydrodynamique ou de l'océanographie furent introduit afin d'animer des scènes littorales [Pea86, FR86, TB87], une approche spectrale [MWM87] ainsi qu'une méthode basée sur une subdivision stochastique de la surface [Lew87] permettant quant à elles de générer des images statiques de scènes océanographiques simples.



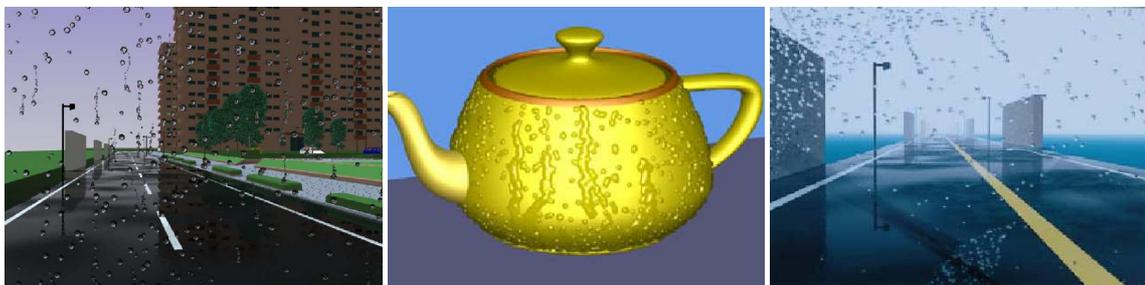
FIG. 1.6: Quelques animations pionnières en synthèse d'images pour de larges étendues d'eaux.

Ces méthodes précurseuses étant intrinsèquement liées à la nature des larges étendues d'eau (voir Figure 1.6), elles ne pouvaient être ni revisitées ni étendues à la problématique des faibles écoulements liquides. Alors que la majorité de la communauté s'orientait vers des méthodes de solution physiques de plus en plus inspirées par la mécanique des fluides tout en demeurant dans l'optique initiale, des méthodes empiriques spécifiques aux gouttes émergèrent en parallèle dès le début des années 1990. Ces premiers travaux donneront lieu par la suite à plusieurs autres contributions, et formeront en raison de leur nature empirique partagée une première famille d'approches qui fera l'objet de la partie 1.2.1.

Depuis leur introduction en synthèse d'images par Foster et Metaxas [FM96] en 1996,

les équations de Navier et Stokes ont permis d'initier progressivement une partie de la communauté graphique à la mécanique des fluides ainsi qu'à la physique numérique, la première traitant des aspects théoriques fondamentaux tandis que la seconde concentre ses efforts sur l'implémentation de méthodes de résolution. En effet, convaincue face au réalisme des animations obtenues (alors jamais atteint) et réalisant graduellement le caractère malléable des équations, la communauté persévéra dans cette voix et apporte depuis de nombreuses méthodes permettant de simuler toujours plus de phénomènes naturels. On pourra citer par exemple la simulation de liquides de diverses natures en interaction, l'ébullition ainsi que l'évaporation résultante, les bulles ou encore les gouttes. Cette famille d'approches issue de la mécanique des fluides sera présentée et analysée dans la partie 1.2.2, cela pour deux raisons. Bien qu'il ait été insinué précédemment l'existence de méthodes dédiées aux gouttes, il n'en existe en réalité qu'une seule prenant en compte la complexité du phénomène dans son intégralité [WMT05]. Cependant, cette méthode pouvant être considérée comme une extension de ses prédécesseurs, la présentation des contributions majeures lui étant antérieures est primordiale à sa compréhension. De plus, une approche physique novatrice ayant fait l'objet de recherche dans le cadre de cette thèse, cette dernière se basant sur la famille de méthodes en question, la nécessité quant à situer et analyser les travaux antérieurs n'en est qu'accrue.

1.2.1 Approches empiriques



(a) Résultat extrait de [KKY93] (b) Résultat extrait de [KZYN96] (c) Résultat extrait de [KIY99]

FIG. 1.7: Différents résultats obtenus par Kaneda *et al.*

La première méthode dédiée au rendu de gouttes fut proposée en 1993 par Kaneda *et al* [KKY93] où il est question d'animer des gouttes d'eau sur une surface plane translucide orientable selon l'axe des abscisses dans l'espace. La simulation empirique emploie une discrétisation 2D régulière de la surface où chaque élément de la grille contient deux variables ainsi qu'une constante, respectivement la masse d'eau et la vélocité à un instant donné ainsi qu'un coefficient dit d'affinité de la surface. Ce coefficient d'affinité local à chaque élément de la grille peut être utilisé afin de représenter la nature hydrophile ou hydrophobe de la surface, mais aussi dénoter la présence d'insalubrités. Un paramètre global nécessaire au déroulement de la simulation est celui de l'angle d'inclinaison de la surface selon une unique dimension et se substitue -non réciproquement- à la force gravi-

tationnelle. En considérant indépendamment chaque élément de la grille, l'algorithme se résume de la façon suivante : si la masse locale dépasse une certaine masse seuil (paramètre fixé par l'utilisateur), alors la goutte est autorisée à se mouvoir vers l'une de ses trois voisines. Vient alors le choix de la direction de mouvement, qui favorise l'élément connexe le plus humide si présent, ou à défaut l'élément ayant le coefficient d'affinité le plus élevé. Avant d'effectuer le déplacement de la goutte, une quantité d'eau sera retenue en l'élément courant proportionnellement à son affinité locale, sa vitesse étant mise à jour relativement à une accélération calculée en fonction de l'angle d'inclinaison de la surface et de l'humidité de l'élément cible. Ainsi, une goutte se déplaçant en zone humide sera accélérée proportionnellement à l'angle d'inclinaison de la surface tandis qu'une goutte se déplaçant en zone sèche sera décélérée inversement proportionnellement à l'angle d'inclinaison de la surface. Connaissant la direction du mouvement et la vitesse de la goutte, le temps nécessaire à la translation de la masse d'eau vers sa nouvelle destination peut être dérivé. A noter que cette translation s'effectue de façon « binaire » et non progressive. Le dernier point à considérer est celui de la fusion de deux gouttes après avoir mis à jour la grille. Dans ce cas précis, la vélocité de la goutte résultante est recalculée selon la loi de conservation des moments de forces, soit le rapport de la somme des deux vélocités pondérées par leurs masses respectives sur la somme des masses avant collision. Les gouttes sont modélisées individuellement comme intersection d'une sphère avec un plan soumise à deux contraintes. Le volume de la sphère non contenu dans le plan doit refléter la masse de la goutte, l'angle de contact formé à l'intersection étant imposé. Inspiré par la relation de Young et par le phénomène d'hystérésis de l'angle de contact, trois paramètres fixés par l'utilisateur sont requis, à savoir l'angle de contact stable, l'angle de reculée et l'angle d'avancée. Selon que la goutte soit immobile, se déplace vers une zone humide ou vers une zone sèche, l'angle de contact imposé sera alors respectivement l'angle de contact stable, l'angle de reculée ou l'angle d'avancée. Afin de rendre la surface mouillée, une technique de lancé de rayons est combinée à de l'environnement mapping [Gre86]. Les rayons sont dans un premier temps intersectés avec l'arrière de la surface (zone non mouillable) pour ensuite être réfractés en utilisant les valeurs de la grille, la couleur des pixels finaux étant alors obtenue en indexant une cubemap représentant la scène par ces derniers.

Peu après, la méthode précédente fut étendue aux surfaces courbes de matériau quelconque et librement orientables dans l'espace [KZYN96]. Une surface courbe définie par des carreaux de Bézier est discrétisée en un maillage non régulier composé de quadrilatères dont les centres adressent les variables, la vélocité étant cette fois exprimée sous forme vectorielle et une information de normale ayant été rajoutée. La simulation se distingue de sa consœur par l'usage de la seconde loi de Newton ainsi que par une dynamique dictée par un modèle probabilistique : la somme des forces extérieures (telles que la gravité et le vent) est projetée sur le plan de chaque élément, la norme du vecteur résultant autorisant le déplacement de la goutte si il dépasse un seuil proportionnel à l'affinité surfacique locale. Dans l'affirmative, chacun des voisins 8-connexe se voit affecté une « note » déterminée

empiriquement selon une combinaison linéaire de trois coefficients, le premier étant relatif à la position de son centre et à la loi du mouvement de Newton, le second à son affinité de surface et le troisième à sa masse liquide. Un tirage aléatoire usant d'une roulette est alors effectué afin de déterminer la direction de mouvement de la goutte, permettant ainsi de perturber occasionnellement le flot et de rompre une certaine uniformité. Les vitesses sont quant à elles mises à jour en intégrant l'accélération dans le temps. Une dernière différence se situe au niveau du rendu où sont mentionnées une approche rapide (adaptation de [KKY93] à un maillage irrégulier courbe) ainsi qu'une approche photo-réaliste où la surface mouillée est exprimée en une surface de Bézier et les gouttes sous forme de metaballs.

Ces travaux furent encore étendus afin de gérer notamment des obstacles dynamiques [KIY99]. Le maillage étant complété avec une information booléenne quant à la présence d'un obstacle, la simulation se déroule alors en deux phases successives où les obstacles sont respectivement considérés comme étant statiques puis libres de mouvement. La première phase se focalise sur la dynamique du flot et est identique en tout point à la simulation proposée dans [KZYN96] si ce n'est que la combinaison linéaire de chaque élément voisin se voit pondérée d'une dirac relative à la présence d'obstacles, forçant ainsi un déplacement des gouttes tangentiel à un obstacle si nécessaire. Lors de la seconde phase dite de « non-gravité », les obstacles mobiles poussent dans un premier temps les gouttes sur leurs passages vers leurs éléments voisins libres les plus proches. Pour chacune de ces gouttes, les autres étant spatio-temporellement gelées, est ensuite calculée une force centrifuge dépendant du moment angulaire de l'obstacle qui déterminera un déplacement supplémentaire en se substituant à l'action des forces extérieures. La méthode de rendu est identique à celle proposée dans [KKY93], deux effets visuels ayant été rajoutés : le « defocus » des gouttes lorsque les objets de la scène sont éloignés de la surface mouillée en atténuant empiriquement les contributions des gouttes pré-filtrées au reste de la scène dans l'espace des couleurs ; l'effet de floue provoqué par les obstacles en mouvement étant obtenu par une rasterization de la surface balayée d'intensité proportionnelle au ratio de la surface balayée sur le laps de temps écoulé entre deux images successives de l'animation.

Synthèse de [KKY93, KZYN96, KIY99] (voir Figure 1.7) : les trois simulations présentent une caractéristique commune, le déplacement des gouttes : une goutte est totalement transportée dans sa nouvelle cellule après un laps de temps fixé. En découle une première limitation qui est celle de la fluidité de l'animation, cette dernière étant directement liée à la résolution du maillage utilisé pour la simulation. Ainsi, afin d'obtenir une animation relativement plaisante, la taille des maillages se doit d'être relativement grande avec les inconvénients que cela apporte (complexité calculatoire et mémoire). On est aussi en droit de remettre en cause la nécessité de modéliser des gouttes avec prise en compte de l'angle de contact, étant donné qu'un flot sera phénoménologiquement continu alors que sa modélisation sera une succession de gouttes individuelles. Le phénomène de réfraction

sera donc faussé dans tous les cas. Concernant les obstacles, ces derniers se résument à des objets soumis à un moment angulaire (tel un essuie-glaces) et non à une force quelconque, ce qui limite les possibilités de représentations d'obstacles.



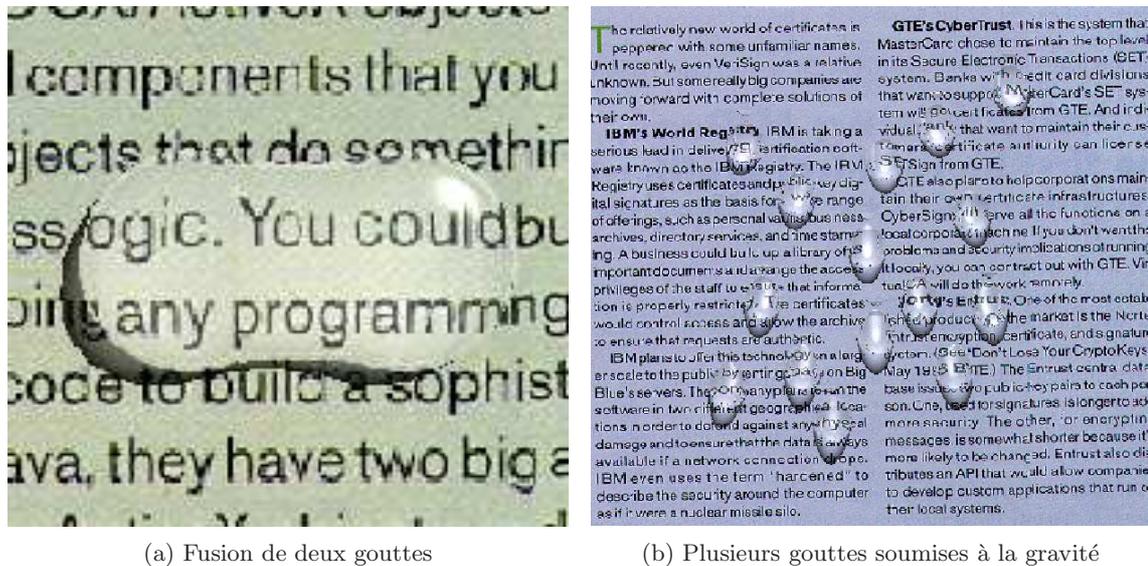
FIG. 1.8: Résultats obtenus par Sato *et al.* [SDY02]

Sato *et al.* [SDY02] réutilisent la simulation empirique de [KKY93] et proposent une technique de rendu tirant parti de la puissance des processeurs graphiques (communément appelés GPU²). Une goutte est modélisée en tant qu'un hémisphère discrétisé sous la forme d'un maillage traditionnel, l'environnement se limitant à une image de fond. Pour chaque élément de la grille 2D de la simulation, si une masse d'eau est présente, alors une texture partiellement transparente (communément appelée billboard) est générée dynamiquement sur GPU à partir de la projection du maillage de la goutte à laquelle est appliquée une technique d'environnement mapping permettant de simuler l'effet de réfraction en modifiant les coordonnées de texture relative à l'image de fond. Lorsque le calcul de toutes les billboards s'achève, l'image de fond est dans un premier temps projetée puis chaque goutte est rendue en projetant orthogonalement sa billboard translaturée en son centre et mise à l'échelle proportionnellement à sa masse d'eau. Un effet de champs de profondeur a aussi été implémenté en convoluant les billboards avec un filtre d'interpolation avant projection (voir Figure 1.8).

Cette approche uniquement dédiée au rendu des gouttes utilise une modélisation simplifiée de celle présentée dans [KKY93, KZYN96, KIY99]. Chaque goutte est considérée individuellement et fait abstraction de son voisinage, l'angle de contact formé étant toujours de $\frac{\pi}{2}$. De plus, la réfraction est approchée par une technique d'environnement mapping 2D et non 3D, ce qui limite le champs de réfraction des gouttes et fausse d'autant plus les informations de couleurs obtenues.

En constatant certaines incohérences entre la forme des gouttes obtenues par une pré-

²Graphics Processor Unit



(a) Fusion de deux gouttes

(b) Plusieurs gouttes soumises à la gravité

FIG. 1.9: Résultats obtenus par Yu *et al.* [YJC98]

écédente approche [KZYN96] et les observations phénoménologiques, Yu *et al.* [YJC98] proposent une modélisation des gouttes basée sur des surfaces implicites. Leur contribution est la modification de la formulation standard des metaballs où l'isosurface de la goutte est déterminée dans le cas présent par un champs scalaire en concordance avec un champs vectoriel. Partant d'un hémisphère (plus précisément, les auteurs ne considèrent que la partie supérieure de l'intersection entre une metaball et un plan passant par son centre), le champs vectoriel à l'isosurface permet d'altérer empiriquement le champs scalaire afin de refléter l'effet des forces de gravitation et de friction agissant sur la goutte, une transformation analytique inverse permettant de corréler l'isosurface initiale avec sa déformation. La forme de la goutte ainsi obtenue peut de plus être soumise à des contraintes spatiales en ajoutant des points de contrôles. L'isosurface des metaballs est alors extraite par une technique de lancé de rayon prenant en compte la réflexion ainsi que la réfraction de la lumière. Dans des travaux faisant suite [YJC99], la méthode précédente est étendue afin de prendre en compte l'impact des insalubrités de surface sur la forme des gouttes en utilisant une discrétisation régulière 2D de la surface où les coefficients en question sont définis.

Ces deux approches [YJC98, YJC99] se focalisent sur la modélisation des gouttes (voir Figure 1.9) et non sur leur dynamique bien que des paramètres physiques soient pris en compte afin d'influencer la forme des gouttes. La principale limitation de ces méthodes est qu'elles ne peuvent être généralisées à un flot de goutte arbitraire étant donné une algorithmique agissant au cas par cas. De plus, afin d'obtenir des résultats réalistes, une modification des paramètres par l'utilisateur peut être nécessaire. Finalement, la modélisation des gouttes en tant que surface implicite rends ces méthodes difficilement adaptables en vue d'atteindre des performances interactives.

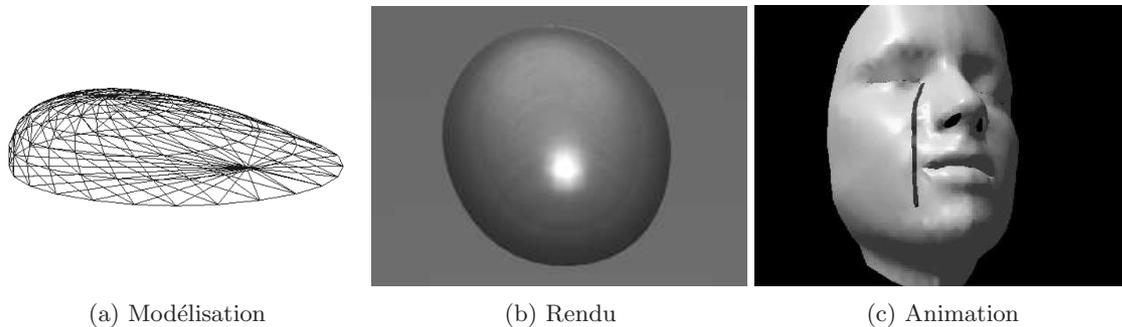


FIG. 1.10: Résultats obtenus par Fournier *et al.* [FHP98]

Fournier *et al.* [FHP98] proposent de dissocier le mouvement des gouttes de leur modélisation afin d'optimiser les temps de calculs. La simulation opère sur un système de particules où chaque goutte est considérée comme une unique particule, un maillage triangulaire représentant la surface. La vitesse de chaque goutte évolue selon une approche inspirée de la mécanique classique modifiée empiriquement afin de prendre en compte des paramètres relatifs à l'adhésion, la friction et la rugosité de la surface. Lorsque les gouttes entrent en mouvement, une phase de détection de collision permet de déterminer les gouttes entrant en fusion, auquel cas la masse de la nouvelle goutte se voit affectée la moyenne des deux anciennes masses tandis que sa position et sa vitesse sont calculées en tant que moyenne pondérée. Partant d'une discrétisation initiale d'un hémisphère, la forme de la goutte est obtenue par la résolution d'un système de masses ressorts soumis à quatre contraintes, ces dernières étant relatives à la conservation du volume, la sensibilité aux forces extérieures, la minimisation de la surface en contact avec l'air et la maximisation de celle en contact avec la surface solide. Les gouttes sont finalement visualisées en temps différé en omettant le phénomène de réfraction tandis que la réflexion de la lumière est approché par une technique d'environnement mapping. La figure 1.10 illustre certains aspects des résultats.

Une sérieuse limitation de cette méthode est la gestion des collisions où plusieurs pas de temps intermédiaires sont nécessaires afin d'obtenir des résultats cohérents. Ces collisions sont rendues d'autant plus complexes qu'elles doivent satisfaire à la conservation de la masse d'un liquide, et donc aucune inter-pénétration de particules devraient être tolérées. Aussi, la forme des gouttes calculée ne dépend pas de sa forme à l'instant précédent, ce qui est une simplification nuisant au réalisme des animations.

Murta et Miller [MM99] proposent une approche basée sur un système de particules adaptatif afin de modéliser et d'animer des gouttes en contact avec l'air ou une surface solide. Les particules sont soumises à l'influence de forces extérieures telle que la gravité, leur distribution étant régie par un modèle d'attraction-répulsion permettant de garder une cohérence en vue du phénomène simulé (une goutte correspond à un regroupement de

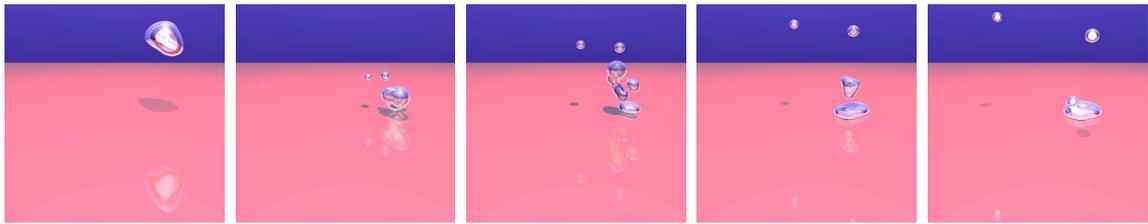


FIG. 1.11: Résultats obtenus par Murta et Miller [MM99] : de gauche à droite, une goutte entre en collision avec une surface solide.

particules scindable uniquement dans de certains cas). Afin d'alléger les temps de calcul sans nuire à la simulation ou au rendu, les particules sont autorisées à fusionner lorsqu'un profil de vitesse est partagé localement, une scission étant de rigueur lors d'un contact solide ou de l'isolement d'une goutte. Lors d'une violente collision avec une surface solide, les particules se verront attribuer une force de rebond et inversement, lors d'une faible collision les forces attribuées seront tangentielles à la surface. Les particules étant considérées comme des surfaces implicites, la conservation de la masse ainsi que la forme des gouttes en contact solide sont adressées par une modification des paramètres relatifs aux surfaces implicites, le rendu étant obtenu par une visualisation de l'isosurface résultante. Afin de conserver la masse, la variation de volume pour chaque goutte est traquée entre deux pas de temps successifs de la simulation, l'approximation des volumes passant par une intégration numérique sur une grille 3D des isosurfaces préalablement discrétisées. Lors d'un contact solide, les paramètres des surfaces implicites sont modifiés dans un premier temps en vue d'éliminer le volume sous la surface pour le réinjecter dans la partie supérieure. L'aplatissement du à la gravité est ensuite pris en compte, le phénomène de l'angle de contact étant quant à lui obtenu par une déformation de l'isosurface relative aux vitesses tangentielles.

D'algorithmique relativement complexe, l'implémentation de cette méthode est rendue difficile de par une présentation trop furtive de certains points clefs. De plus, bien que les résultats visuels semblent convaincant (voir Figure 1.11), le manque d'informations les accompagnant rendent difficile d'estimer la complexité mémoire et calculatoire nécessaire à l'obtention d'animations convaincantes.

Iglesias *et al.* [IGPP01] se sont intéressés au mouvement d'une goutte sur une surface solide courbe. La surface est modélisée en tant qu'une surface paramétrique différentiable ou bien une union de carreaux continue de ces dernières (les surfaces de type Bézier, B-Splines et NURBS ont été implémentées par les auteurs). A noter que la dynamique propre aux gouttes n'est pas prise en compte dans la présente méthode, les propriétés d'une goutte se résumant à une position dans l'espace dotée d'une force et les interactions n'étant pas prises en compte. Une équation différentielle ordinaire (EDO) est formée afin de paramétrer la localisation de la goutte par le « temps » ou la longueur d'arc. L'EDO est ensuite intégrée par un schéma de Runge-Kutta explicite à l'ordre 4, la position de la

goutte à tout instant étant ainsi obtenue. Le passage d'une goutte sur un extremum local de la surface ou la transition d'une goutte d'un carreau de surface à un autre posant des problèmes de stabilité numérique, un ajustement manuel est nécessaire afin de converger vers une trajectoire cohérente.

Bien que dédiée aux gouttes dans son intitulé, cette méthode a plus attiré à la résolution numérique d'un système général où un point quelconque se mouvoit sur une surface paramétrique. Les propriétés relatives aux gouttes sont omises dans leurs dynamiques, les éventuelles collisions inter-gouttes étant ignorées.

Tong *et al.* [TKY02] adressent le problème spécifique de la conservation volumique lors d'un flot généré par des metaballs. Bien que le flot se doit d'être animé dans le temps afin que la problématique ait un sens, ces derniers considèrent deux configurations consécutives du flot en faisant abstraction de la méthode utilisée pour la simulation. Dans un premier temps, le flot est décomposé en ses sous-ensembles disjoints de metaballs afin de ne s'intéresser qu'à la conservation volumique locale des flaques. Une intégration numérique récursive accélérée par une analyse d'intervalle permet ensuite d'estimer les volumes de chaque sous-ensemble pour la configuration du flot précédente et actuelle. En posant un système contraint par l'immobilité des centres des metaballs, les paramètres des metaballs sont recalculés pour le flot courant afin de recouvrir les volumes de chaque flaque à l'itération précédente.

Les auteurs présentent ici un outil numérique rigoureux pouvant servir au sein de méthodes s'y prêtant telles que [YJC98, YJC99]. L'utilité d'une telle méthode se justifie par l'importance de conserver le volume d'un liquide, cela même dans le cadre de la synthèse d'images.

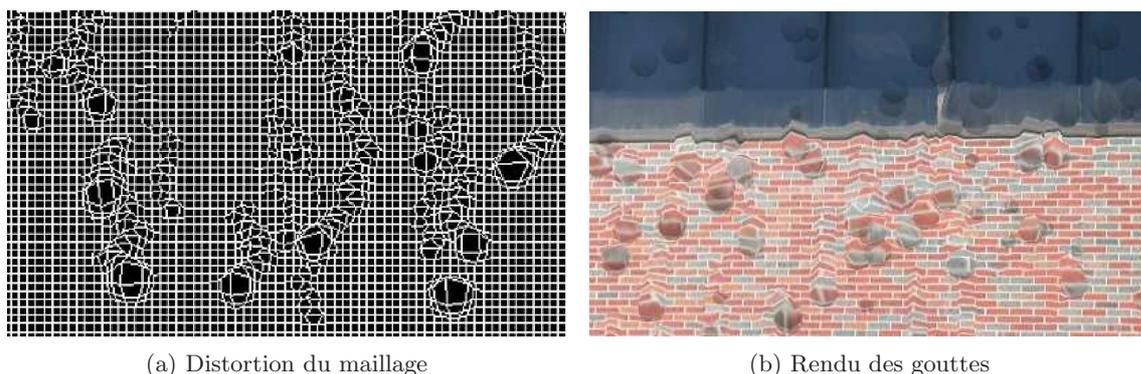


FIG. 1.12: Résultats obtenus par Yang *et al.* [YZZ04]

Yang *et al.* [YZZ04] proposent une approche temps-réel au rendu de gouttes dans le cas d'une surface plane parallèle faisant face à l'observateur. La scène est dans un premier temps rendue pour être stockée dans une texture 2D indexée par une discrétisation régulière 2D de la surface mouillée, cette dernière recouvrant la fenêtre d'affichage. Le maillage de la surface est ensuite déformé dans le plan en fonction de la présence de gouttes selon

une formule trigonométrique empirique agissant conceptuellement comme une loupe grossissante. Cette formule trigonométrique dépend du rayon et du centre de la goutte ainsi que d'un paramètre scalaire déterminant l'importance de la déformation locale au maillage. La surface mouillée est finalement visualisée en plaquant sur le maillage déformée la texture de fond, simulant ainsi simplement la réfraction de l'environnement.

En simulant le phénomène de réfraction du aux gouttes par une distortion de maillage à faible coût calculatoire, cette approche est capable de produire des animations en temps-réel. Cependant, il n'est pas clair comment la dynamique des gouttes est gérée au sein de la simulation, l'absence de réflexion de la lumière nuisant au réalisme des animations (voir la figure 1.12).

1.2.2 Méthodes basées physique

La dynamique des fluides est une branche des sciences physiques consacrée à l'étude générale du comportement des fluides, comportement considéré comme étant entièrement décrit à l'échelle macroscopique par les équations de Navier et Stokes depuis le début du XIX^{ème} siècle. Les équations de Navier et Stokes formant un système d'équations différentielles partielles non linéaires, il n'existe toujours pas à ce jour de solutions analytiques en raison de leur complexité, ni de preuves quant à l'existence et l'unicité d'une solution en trois dimensions. Avec l'avènement de l'informatique et des supers calculateurs, les physiciens ont pu mettre à l'oeuvre les équations de Navier et Stokes lors de simulations numériques dès le milieu du XX^{ème} siècle, et vérifièrent ainsi leur aptitude à reproduire avec exactitude de multiples phénomènes naturels. Les résultats issus de ces simulations présentant des intérêts majeurs tant d'un point de vue recherche qu'industriel, une branche à part entière de la recherche dite dynamique des fluides numérique naquit, cette dernière étant consacrée aux méthodes de résolution numérique pour les équations de Navier et Stokes.

En dynamique des fluides numériques, une méthode de résolution se définit principalement par trois axes majeurs qui sont :

- la discrétisation du domaine
- les opérateurs différentiels
- les méthodes d'intégrations

La discrétisation du domaine est certainement le point le plus important à considérer, on distingue notamment les approches dites lagrangiennes (où le mouvement du fluide est suivi) des approches eulériennes (où le mouvement du fluide est observé statiquement). Techniquement parlant, la première discrétisera un fluide en un champs de particules alors que la seconde usera d'un maillage du domaine. Historiquement les premières, les méthodes eulériennes ont été intensivement étudiées et présentent de nombreux avantages

justifiant leur popularité en dynamique des fluides numériques. On pourra citer entre autres : une structure adaptée aux architectures machines dans le cas de discrétisation régulière, des opérateurs différentiels rigoureusement dérivés de séries de Taylor avec en sus une estimation d'erreur théorique, une localisation aisée des interfaces. A contrario, les approches lagrangiennes usent de filtres d'interpolation afin d'estimer les dérivées spatiales, des structures hiérarchiques sont nécessaires à l'accélération de la recherche de voisinage, la reconstruction des interfaces étant quant à elle non triviale. En revanche, les approches lagrangiennes respectent de façon intrinsèque la conservation de la masse. Il existe aussi une troisième classe de méthodes relativement récentes et de plus en plus étudiées, les méthodes dites de Lattice-Boltzmann. Issues de la généralisation de l'équation de Boltzmann pour les gazes aux cas des fluides, les méthodes de Lattice-Boltzmann permettent de simuler à l'échelle mésoscopique un fluide discrétisé sur une grille en considérant successivement la propagation puis la collision de densités de particules. D'algorithmique relativement simple, on notera cependant la nécessité de considérer un voisinage dense (pouvant aller jusqu'à la 26-connexité en trois dimensions) lors des calculs, et surtout une étendue plus restreinte des phénomènes simulés (telle que la dynamique de la triple ligne de contact) en comparaison avec les équations de Navier et Stokes.

Dans le cadre de cette thèse, il a été décidé d'explorer une approche basée physique pour l'animation de gouttes, et le choix s'est orienté vers une approche eulérienne pour les avantages qu'elle présentait potentiellement. Un état de l'art complet sur l'animation basée physique de liquides n'étant pas envisageable dans le présent manuscrit, la littérature à son sujet étant trop consistante, seules des méthodes jugées en rapport avec notre problématique seront présentées en détail dans ce qui suit. On s'intéressera dans un premier temps aux approches dédiées à l'animation de fluides -dans la majorité des cas, il s'agira d'un liquide- dans un contexte conventionnel (c'est à dire, où les trois dimensions de l'espace sont conceptuellement considérées et d'intérêts), piochant ainsi dans une vaste littérature. Des méthodes conçues pour des écoulements surfaciques seront ensuite présentées où une simplification de domaine est généralement de rigueur. Ce dernier point justifie ainsi l'intérêt porté à ces approches étant donné notre problématique, ces méthodes tant spécialisées qu'originales jouissant d'une littérature relativement restreinte.

On notera que de méthodes lagrangiennes basées sur les SPH ³ [Luc77, GM77] ainsi que sur les MPS ⁴ [KO96] ont été introduites avec succès en synthèses d'images dans respectivement [MCG03, MST⁺04, MSKG05] et [PTB⁺03]), de même pour des méthodes de Lattice-Boltzmann [TR04, FZKH05, TIR06].

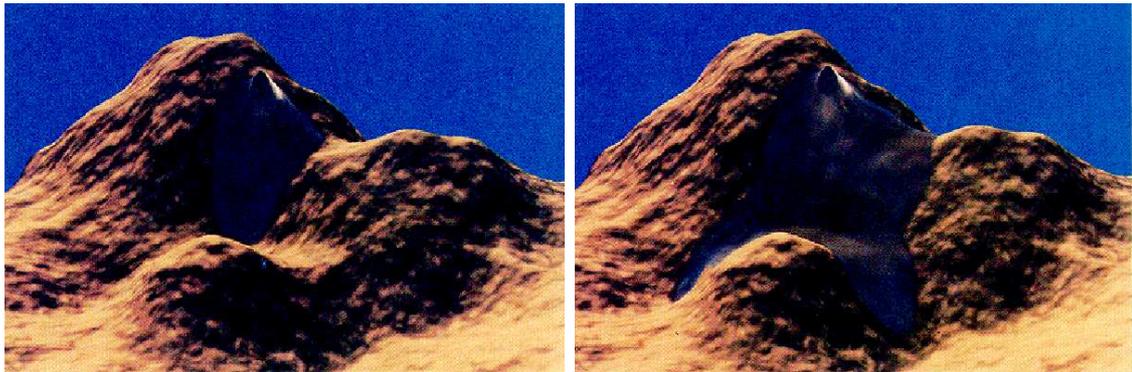


FIG. 1.13: Résultats obtenus par Kass et Miller [KM90] : une coulée d'eau sur une colline rocheuse

Animation de fluides dans l'espace

Kass et Miller [KM90] proposèrent la première méthode basée sur un système d'équations différentielles partielles (EDP) permettant d'animer un liquide. Dans cette approche, le liquide est discrétisé en un champs de hauteur (soit, une grille régulière 2D) dont le mouvement est régit par une simplification des équations d'eau peu profonde où seule l'accélération gravitationnelle est considérée. Un second champs de hauteur est de plus mis à contribution afin de modéliser la géométrie de la surface solide sur laquelle le liquide évolue. En utilisant un schéma d'intégration implicite (et donc stable), les EDP forment un système d'équations linéaires tridiagonal (et donc rapide à résoudre) dont le terme de droite dépend des solutions aux deux itérations précédentes. Un effet de viscosité peut de plus être achevé en pondérant les variables des deux itérations précédentes. Une technique simplifiée de génération de texture est aussi présentée afin de rendre les caustiques, une autre texture dite d'humidité servant à obscurcir les zones ayant été mouillées.

En discrétisant le fluide en un champs de hauteur, plusieurs phénomènes liquides ne peuvent être simulés tel que les déferlements ou les splashes. De plus, le terme gravitationnel étant le seul pris en compte dans les équations d'eaux peu profondes, les vagues générées évoluent indépendamment de la configuration globale du liquide. Cependant, ce sont ces mêmes limitations qui sont à la base d'une implémentation aisée et de performances attractives tant numériquement que visuellement (voir Figure 1.13).

Le modèle précédent se limitant conceptuellement à une superposition de vagues d'accélération constante, Chen et Lobo [CdVL95] se sont intéressés aux équations de Navier et Stokes dans leur formulation 2D afin d'enrichir les animations de liquides de turbulences ainsi que d'interactions solides. Leur méthode de solution est basée sur une discrétisation « staggered Marker And Cell (MAC) » [HW65], soit une grille régulière où les champs de vitesses sont délocalisés et le champs de pressions est centré, des particules inertielles

³Smoothed Particle Hydrodynamics

⁴Moving Particle Semi-Implicit

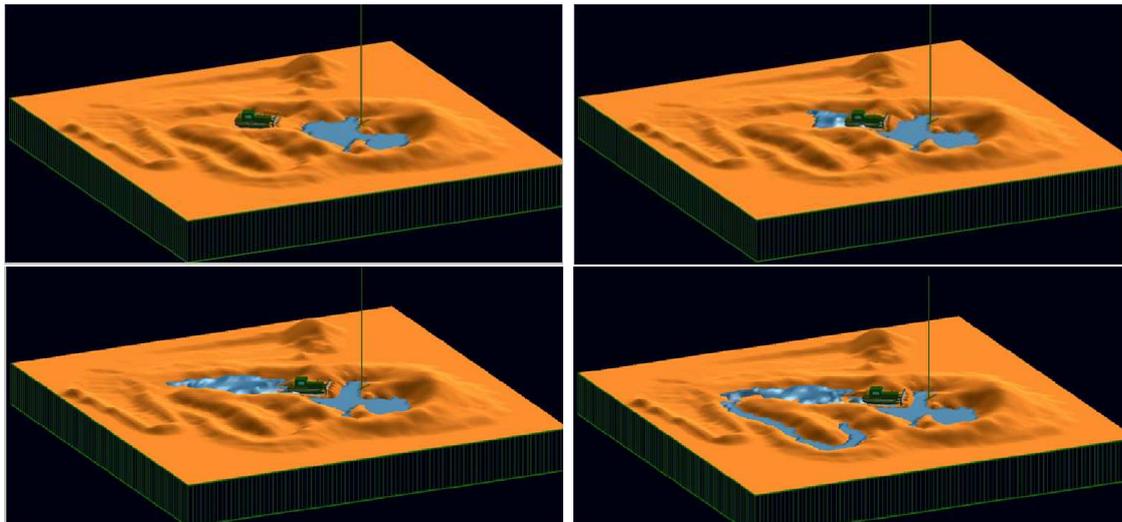


FIG. 1.14: Résultats obtenus par Chen et Lobo [CdVLHM97]. De haut en bas, de gauche à droite, une animation d'une marre s'étendant sur le terrain du à un déplacement d'obstacles.

transportées par le champs de vitesse déterminant les éléments de la grille occupés par le liquide. A chaque itération, les champs de vitesses sont intégrés dans le temps explicitement et le champs de pressions est recouvert par une technique de pénalisation afin de satisfaire le théorème de la divergence. La grille étant en deux dimensions, les auteurs proposent de reconstruire la surface libre du liquide en élevant chaque élément de la grille proportionnellement à sa pression. Afin de considérer les interactions solide-liquide, des conditions aux bords sur les champs de variables sont introduites simulant ainsi des objets transportés à la surface du flot. La méthode est étendue [CdVLHM97] afin de conserver le volume liquide au cours de la simulation, des conditions aux bords dynamiques permettant de modifier l'environnement dans le temps. Cependant, le champs de pression qui se développe phénoménologiquement dans les profondeurs et à l'origine de perturbations à la surface est toujours ignoré, la représentation des obstacles en 2D limitant grandement les possibilités d'interactions liquide-solide. Un exemple d'animation avec des obstacles mobiles est montré dans la figure 1.14.

Foster et Metaxas [FM96] furent les premiers à utiliser les équations complètes de Navier et Stokes en 3D afin de simuler un flot liquide de façon réaliste. Une grille staggered MAC est employée où l'ensemble des termes composant l'EDP sont intégrés selon un schéma explicite. Afin de satisfaire à la contrainte de la conservation du volume, une technique itérative de relaxation dépendant d'un paramètre fixé empiriquement est employée afin de mettre à jour le champs de pression, les vitesses satisfaisant le théorème de la divergence par la soustraction des gradients de ce dernier. Différentes conditions aux bords sont présentés afin de simuler des objets statiques, transportés à la surface ou plongés dans le liquide. A partir des particules inertielles, un champs de hauteur est re-



FIG. 1.15: Résultats obtenus par Foster et Metaxas [FM96]

construit qui sera advecté avec le flot permettant ainsi de représenter la surface du liquide. La différence fondamentale avec l'approche de Kass et Miller [KM90] réside dans le fait que la surface du liquide évolue relativement à son champs de vélocité et non plus qu'à la force gravitationnelle, et peut être ainsi influencée par la présence de fortes pressions dans les profondeurs du volume liquide. Finalement, afin de prendre en compte plus de phénomènes naturels tels que les splashes et les vagues déferlantes, les particules soumises à de fortes forces sont prises en compte dans la simulation en leur appliquant les lois de la mécanique Newtonienne, complétant ainsi la surface libre dictée par le champs de hauteur. La limitation majeure de cette méthode provient d'une résolution fortement encline à des instabilités numériques, de petit pas de temps (et donc de longs temps de calcul) permettant l'obtention de résultats de bonne qualité certes (voir la figure 1.15).

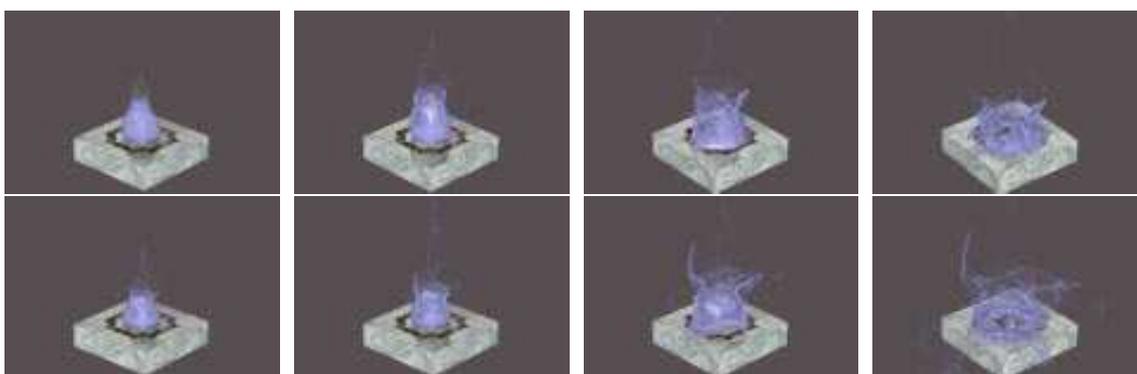


FIG. 1.16: Résultats obtenus par Foster et Metaxas [FM97] : en haut, de gauche à droite, une explosion de peinture sans la prise en compte des effets de tensions de surface. En bas, la même animation avec les effets de tensions de surface intégrés à la simulation. On remarque dans ce dernier cas la tendance du liquide à former des feuillets et non des gouttes.

Les auteurs étendent leurs travaux dans [FM97] en vue d'une meilleure stabilité numérique, d'une modélisation du liquide plus générale et de contrôler l'animation. Lors de la présence de fortes vitesses dans les champs de vélocités, le pas d'intégration temporel se doit

d'être fractionné en plusieurs petit pas intermédiaires afin que la simulation n'explose pas numériquement respectant ainsi la condition de CFL. Afin de relaxer cette contrainte, les vitesses dépassant un certain seuil sont atténuées localement de façon empirique. Bien que cette modification directe des variables affecte théoriquement la forme du flot découlant de la simulation, les auteurs prétendent que la différence en découlant est négligeable pour la synthèse d'images, les principales caractéristiques du flot étant tout de même capturées (tourbillons, splashes etc.). La surface liquide est extraite à partir des particules inertielles servant de marqueurs, une projection d'ellipsoïdes centrées en leur positions ainsi qu'éti-rées selon leurs vitesses ayant été adoptée dans le cas présent. Les auteurs mentionnent cependant la possibilité d'utiliser d'autres techniques de rendu adaptées basées sur des surfaces implicites, une triangulation du maillage ou une intégration volumique. Parmi les méthodes de contrôle d'animation est mentionnée la prise de compte simplifiée des effets de tension de surface. Chaque voxel intersecté par la surface libre du liquide a son voisinage 6 connexes exploré, et pour chacun de ses voisins eux aussi intersectés par la surface libre, la pression est décrétementée selon un coefficient dit de tension de surface. Il en résulte ainsi un champs de pressions négatif à la surface du liquide approchant les effets de tension de surface physiquement parlant qui se traduit par une rétractation de la surface libre sur elle-même (voir la figure 1.16).

Afin de remédier aux problèmes de stabilité numériques des approches précédentes [CdVL95, CdVLHM97, FM96, FM97], Stam [Sta99] proposa un solveur stable pour les équations de Navier et Stokes permettant l'usage de pas d'intégration significativement plus grands. Chaque terme de l'EDP est résolu successivement et indépendamment selon la méthode des pas fractionnaires, la stabilité du solveur découlant de chacun des schémas d'intégration utilisés. Le terme d'advection est résolu par la méthode des caractéristiques (aussi dite semi-lagrangienne) qui trace la solution en arrière le long d'une ligne caractéristique de courant. Ce schéma présente deux qualités non négligeables qui sont sa simplicité d'implémentation et sa stabilité inconditionnelle due au fait que les variables sont mises à jour avec les valeurs du champs précédent (l'explosion numérique n'est plus possible car la solution est bornée par les optimums globaux de la solution précédente). Le terme de viscosité est quant à lui intégré selon un schéma implicite, les vitesses devenant ainsi solution d'un système d'équations linéaires, système dont la matrice clairsemée et bien conditionnée permet une inversion rapide par des méthodes itératives. Stam proposera d'utiliser la méthode itérative de Jacobi -facile d'implémentation- pour cette classe de systèmes. La contribution majeure se situe au niveau du traitement du champs de pressions qui, délaissé lors des précédents pas fractionnaires, sera finalement pris en charge en usant du théorème de Helmholtz-Hodge. Ce dernier statue que tout champs vectoriel peut se décomposer en un champs vectoriel de divergence nulle et en un champs de gradients (issu d'un champs scalaire donc). En faisant l'analogie entre le champs vectoriel quelconque et le champs de vitesses en cours d'intégration ainsi qu'entre le champs scalaires et le champs de pressions, un système d'équations linéaires pour la pression est formé en tirant partie du fait

que le champs de vitesse solution aura sa divergence nulle, ce système étant résolu de la même manière que lors du traitement du terme de viscosité. Le champs de vitesses final conservant la masse est alors obtenu par simple soustraction avec les gradients de pression. On notera que cette méthode est dédiée à l'animation d'un gaz et ne traite pas des fluides telle que l'eau où une interface est présente.

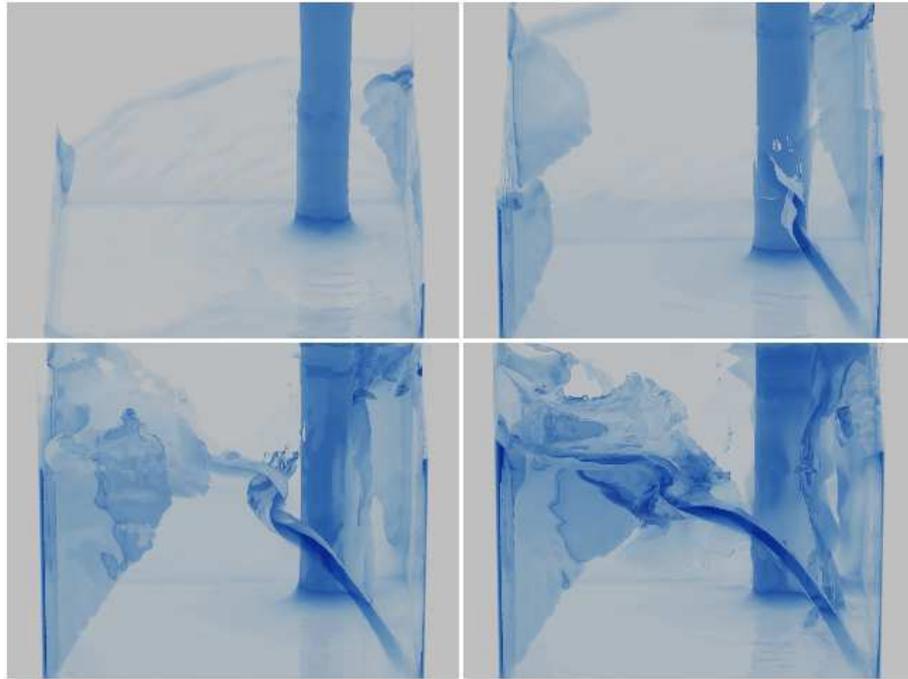
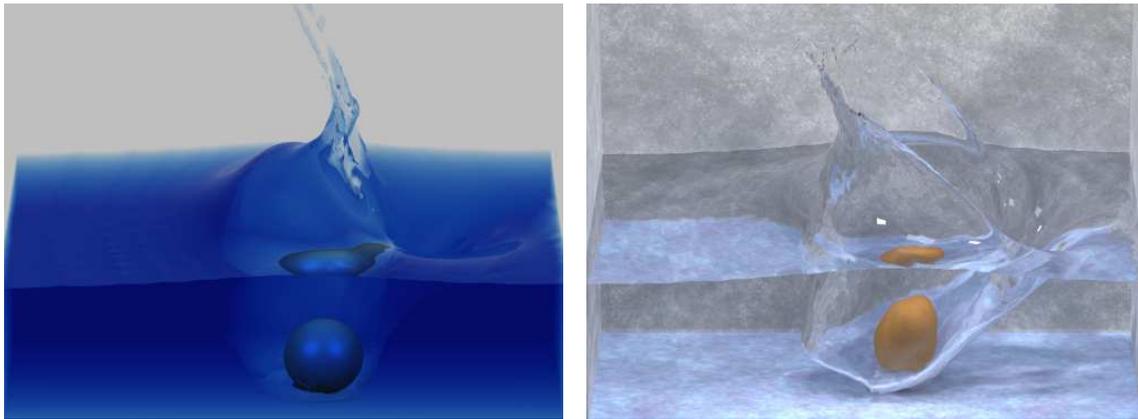


FIG. 1.17: Résultats obtenus par Foster et Fedkiw [FF01]

Foster et Fedkiw [FF01] adaptent la méthode de Stam [Sta99] à l'animation de liquides afin de bénéficier de ses avantages en terme de performance et de stabilité. Le volume liquide est modélisé en tant qu'un level set [OS88], soit un champs de distances signées où l'interface se situe à l'isosurface nulle, le signe désignant l'intérieur ou l'extérieur du liquide. Cette représentation continue d'un domaine permet de pister efficacement la localisation de l'interface et est parfaitement adaptée aux cas de fusion et de séparation, de précieuses propriétés telles que le vecteur normal ou la courbure pouvant être de plus estimées fiablement. Cependant, l'advection répétée d'un level set telle que dans une simulation de fluide pose des problèmes de conservation volumique qui ne peuvent être tolérés en physique numérique comme en synthèse d'images. Les auteurs proposent alors de combiner un level set avec des particules inertielles afin de bénéficier de leurs avantages complémentaires. A chaque nouveau champs de vitesses obtenu, le level set ainsi que les particules sont advectées par le flot. Lorsque le level set proche de l'interface présente une forte courbure, les particules sont utilisées en tant que surfaces implicites afin de reconstruire localement le level set, et donc redéfinir son isosurface nulle. De plus, la résolution du level set limitant sa capacité à représenter de faibles quantités liquides en suspension, les particules échappent

pées du volume liquide sont autorisées à contribuer activement à la simulation en étant soumises aux lois de la mécanique classique ainsi qu'au rendu où gouttes et splashes peuvent être observés. Ainsi, la résolution des équations de Navier et Stokes peut bénéficier d'une advection semi-lagrangienne, le terme visqueux étant quant à lui intégré explicitement. En reformulant astucieusement les conditions aux bords pour le champs de pressions et reprenant l'idée de projection de Stam [Sta99], un système d'équations linéaires symétrique défini positif est posé pour lequel des méthodes itératives d'inversion basées sur les espaces de Krylov existent. Ces derniers, dont la convergence peut se prouver formellement, nécessitent beaucoup moins d'itérations afin de converger vers une solution acceptable et ne peuvent en aucun cas diverger, contrairement à une méthode itérative de Jacobi utilisée dans [Sta99]. Le champs de pression est obtenu par la méthode du gradient conjugué préconditionné [Saa03] et le rendu final par une triangulation du level set avec un algorithme de marching cubes [LC87] (voir la figure 1.17).



(a) Résultat extrait de [FF01]

(b) Résultat extrait de [EMF02]

FIG. 1.18: Comparaison visuelle pour une scène identique de la méthode [FF01] (figure (a)) et de la méthode [EMF02] où le comportement du liquide est plus réaliste (figure (b)).

Bien que de grande qualité, les animations produites par la méthode précédente [FF01] sont parfois sujettes à certaines incohérences allant à l'encontre des observations phénoménologiques. Enright *et al.* [EMF02] réalisent que la source de ces dernières est liée en partie à une trop pauvre représentation du volume complémentaire au liquide et proposent une approche se focalisant conceptuellement sur l'interface et non plus uniquement sur le volume liquide. Les auteurs étendent directement les travaux de Foster et Fedkiw en proposant de border l'interface liquide-air par des particules inertielles des deux cotés. Ainsi, avec des heuristiques appropriées pour la distribution des particules ainsi que la reconstruction du level set, la conservation de la masse est respectée tout en représentant avec précision l'interface cette fois-ci. Les conditions aux bords du champs de vitesses sont posées non plus « manuellement » comme dans de précédentes approches [CdVL95, FM96, FF01] mais « automatiquement » en utilisant une technique d'extrapolation rapide [Set96] des

vélocités dans l'air. A noter que cette extrapolation est effectuée jusqu'à plusieurs voxels relativement à l'interface, et contribue à une advection plus réaliste de la surface libre du liquide telle qu'illustrée dans la figure 1.18.

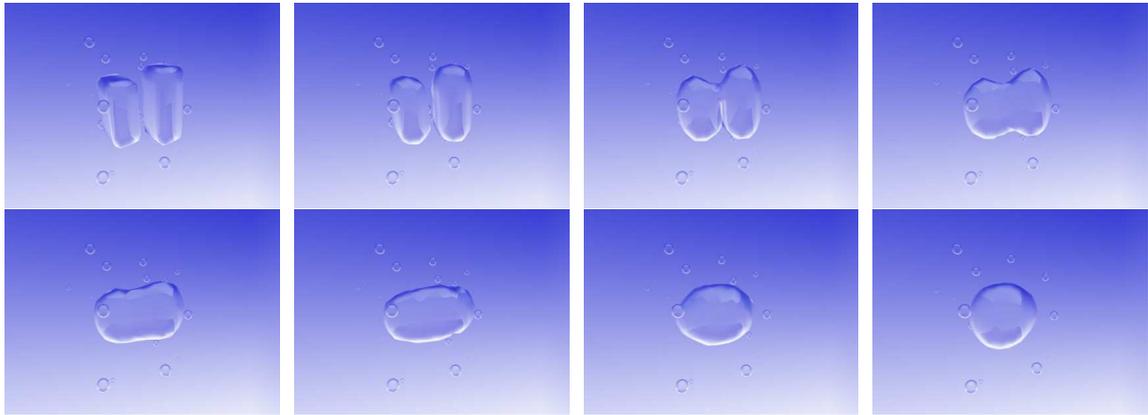


FIG. 1.19: Résultats obtenus par Hong et Kim [HK03] : de haut en bas et de gauche à droite, deux bulles fusionnent sous les effets de tensions de surface.

Hong et Kim [HK03] se sont penchés sur l'animation de bulles d'air dans un liquide, phénomène dont l'origine physique est due à la différence de densités (qui se traduit en flottabilité) ainsi qu'aux tensions de surface (influençant la forme des bulles). Ainsi, une simulation diphasée (soit deux fluides de phase différente en interaction) est proposée pour la première fois avec la prise en compte des effets de tensions de surface. Le fluide est discrétisé selon un schéma Volume Of Fluid (VOF) [HN81] popularisé en physique numérique pour ses propriétés de conservation massique et introduit en synthèse d'images par Kunitatsu *et al.* [KWF⁺01], mais aussi pour la difficulté à reconstruire l'interface notamment en 3D où des méthodes itératives coûteuses sont nécessaires. Comme dicté par le théorème de Laplace, les tensions de surface dépendent linéairement de la courbure à l'interface, d'où l'importance d'une bonne estimation des courbures dans un schéma numérique. La discrétisation VOF des fluides ne permettant pas d'obtenir assez fiablement une telle mesure, les auteurs propose d'estimer empiriquement les effets de tensions de surface à partir de la grille de simulation en les incorporant par la suite en tant que force extérieure. La première étape est l'estimation d'une ligne virtuelle dite de stress nulle dont l'intensité est la moyenne des volumes du voisinage 6 connexe. En observant la variation de volume relativement à la ligne virtuelle selon les 3 dimensions, une force est construite de direction normale à l'interface, orientée vers la ligne virtuelle et dont la norme est proportionnelle à la déformation interfaciale. Cette force une fois ajoutée en tant que telle dans les équations de Navier et Stokes pour les voxels intersectés par l'interface permet de simuler les effets de tension de surface sans passer par une reconstruction de l'interface. Le reste de la simulation opère comme décrit dans [FF01] sauf pour l'advection des volumes où le schéma employé est enclin à de la diffusion numérique et donc à une perte volumique. Afin d'y remédier, le rendu étant basé sur un algorithme de marching cubes en dépit de la nature

volumique des variables, les auteurs détectent la différence globale de volume entre deux itérations successives afin de modifier empiriquement la valeur de l'isosurface à visualiser. Des résultats sont montrés sur la figure 1.19.

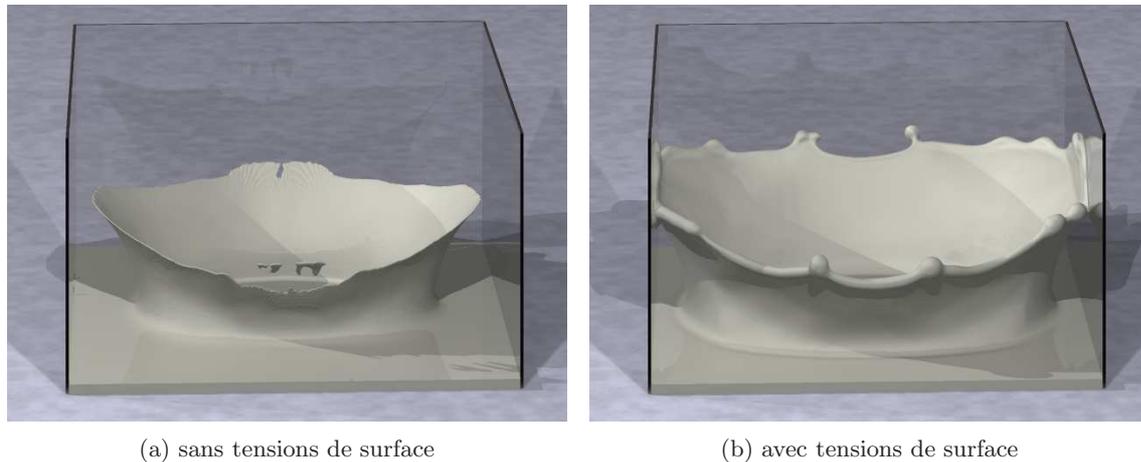


FIG. 1.20: Résultats obtenus par Losasso *et al.* [LGF04]

Dans un article focalisant sur l'adaptation de la méthode des particle level set à des grilles adaptatives, Losasso *et al.* [LGF04] considèrent aussi les effets de tensions de surface. En raffinant localement le maillage vers la surface libre du liquide, les conditions aux bords pour la pression sont modifiées afin de prendre en compte le saut de pression découlant des tensions de surface. Les résultats obtenus sont convaincants et sont très certainement dus à une bonne estimation des courbures à l'interface grâce à la formulation level set employée ainsi qu'au raffinage local du maillage (voir la figure 1.20). Cependant, le degré de raffinement du maillage peut être élevé afin de reproduire visuellement le phénomène de tensions de surface, la résolution des EDP étant quant à elle plus complexe et donc plus lente en raison du caractère adaptatif de la méthode.

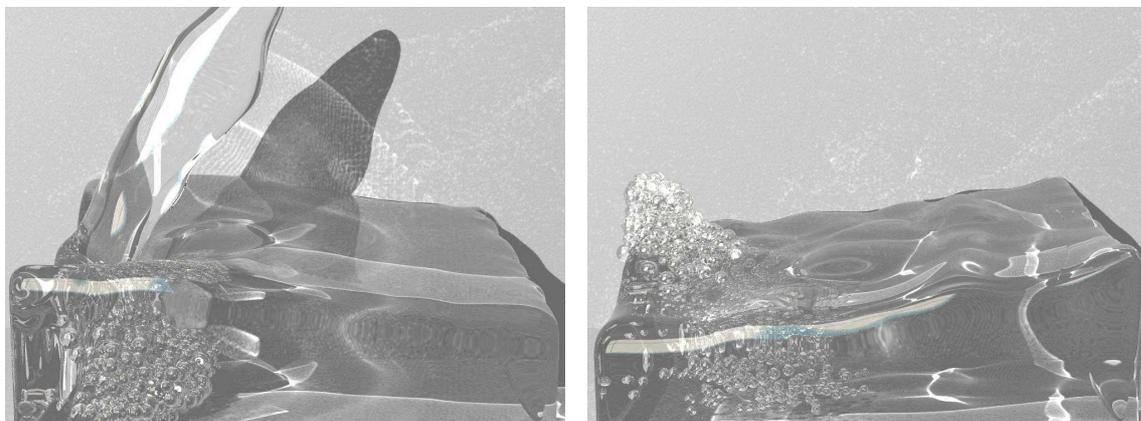


FIG. 1.21: Résultats obtenus par Greenwood et House [GH04]

Greenwood et House [GH04] réutilisent directement la méthode des particles level set [EMF02] afin d’y incorporer des bulles. Les auteurs posent l’hypothèse que les bulles n’ont pas besoin d’être simulées physiquement parlant, leurs contributions au flot étant négligeables. Ainsi, la méthode intègre des bulles sous forme de particules sphériques qui seront passivement advectées par le flot, ces dernières étant créées à chaque fois que des particules inertielles issues de l’air se retrouvent sous l’interface ou lorsqu’une poche d’air est détectée dans le liquide par un algorithme de « rasterization ». Cependant, on remarquera que les bulles générées par la présente méthode ne sont pas aussi réaliste que dans d’autres approches notamment basées physiques (voir la figure 1.21).

Cohen et Molemaker [CM04] intègrent à la méthode des particle level set [EMF02] les effets de tensions de surface afin d’animer des vagues capillaires. Les auteurs négligent le terme de viscosité dans les équations de Navier et Stokes et y rajoutent les effets de tensions de surface en tant que force extérieure selon un modèle physique [BKZ92]. Simple mais ingénieuse, leur idée se base sur l’observation de la sévérité de la restriction CFL pour le terme de tensions de surface en comparaison avec celle usuellement employée pour les termes standard des équations de Navier et Stokes. Ainsi, selon une méthode des pas fractionnaires, les effets de tensions de surface sont intégrés indépendamment au champs de vitesses en plusieurs petit pas intermédiaires, ces derniers étant issus d’une subdivision du pas de temps employé pour le reste des termes des équations de Navier et Stokes. Le champs de vitesse est ensuite mis à jour de façon classique avec le pas de temps global, les animations résultantes permettant d’observer les effets de tensions de surface.

Song *et al.* [SSK05] proposent une formulation diphasée des équations de Navier et Stokes avec un level set où le problème de diffusion numérique inhérent aux méthodes semi-lagrangiennes est contrée en utilisant une advection basée CIP (Constrained Interpolation Profile) [KWF⁺01]. Brièvement, ce schéma d’avection permet une plus robuste estimation des dérivées spatiales capturant ainsi plus précisément les chocs qui peuvent survenir à l’interface de deux fluides aux propriétés significativement différentes (comme c’est le cas pour l’eau et l’air). Bien que les effets de tensions de surface soient intégrés dans le solveur sous forme de forces extérieures [BKZ92], l’impact visuel est difficilement remarquable, cela étant très probablement du à une condition de CFL trop laxiste comme mentionné par les auteurs. Lorsque de trop faibles volumes apparaissent au cours de la simulation, ces derniers sont estimés numériquement pour être ensuite modélisés sous forme de petites bulles ou gouttes ensuite soumises à une physique lagrangienne. Sans cette manoeuvre et sous l’effet de la dissipation numérique, ces phénomènes ne seraient plus observables après plusieurs itérations sur le level set.

Hong et Kim [HK05] présentent la première méthode basée physique focalisée sur les effets de tensions de surface et permettant d’animer des gouttes et des bulles en suspension. En utilisant une méthode issue de la physique numérique dite des fluides fantômes (Ghost

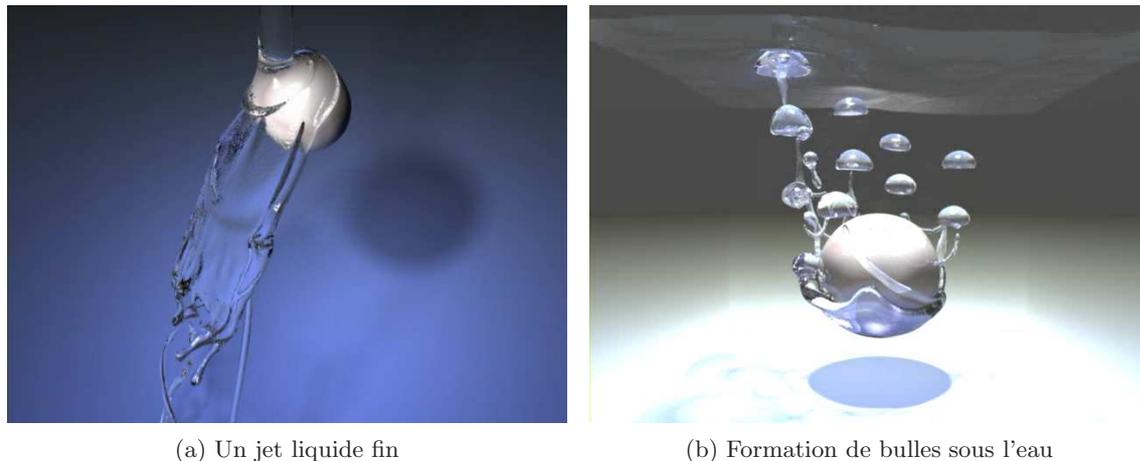


FIG. 1.22: Résultats obtenus par Hong et Kim [HK05]

Fluid Method, alias GFM) [FAMO99] préalablement utilisée en synthèse d'images dans le cadre de la modélisation du feu [NFJ02], le saut de pression théorique due aux tensions de surface agissant à l'interface est intégré dans le terme de droite de l'équation de Poisson pour la pression qui survient lors de la projection du champ de vitesse dans leur espace de divergence nulle. Cette formulation des effets de tensions de surface présente trois avantages majeurs. Tout d'abord, malgré une discrétisation des opérateurs différentiels sur une grille régulière, la courbure à l'interface est estimée exactement par interpolation linéaire des courbures aux voxels en détectant la position de l'isosurface nulle grâce au level set. La GFM permettant de traiter chaque dimension indépendamment, l'implémentation est rendue aisée même en trois dimensions. Finalement, et non des moindres, le terme de droite étant uniquement modifié dans l'équation de Poisson, le système d'équations linéaires demeure symétrique défini positif et donc propice à une inversion par des méthodes itératives rapides aux bonnes propriétés de convergence. En sus de la discontinuité de la pression, les auteurs reformulent l'intégration implicite du terme de viscosité dans les équations de Navier et Stokes afin de prendre en compte la discontinuité qui lui est relative, le système d'équations linéaires en découlant étant aussi symétrique défini positif. Ainsi, bulles et gouttes en suspension sont animées avec réalisme (voir la figure 1.22) sans la nécessité d'un contrôle extérieur, la méthode ne requérant que peu de modifications comparée aux précédents solveurs et cela à un faible surcoût calculatoire.

Wang *et al.* [WMT05] utilisent la méthode des particules level set [EMF02] dans le but d'animer des gouttes liquides sur des surfaces solides. Leur contribution se situe au niveau de l'estimation des tensions de surface agissant à la triple ligne de contact, les méthodes précédentes s'étant focalisées uniquement sur les interfaces plus simples conceptuellement liquide-fluide. Pour ce faire, les auteurs partent de la loi de Young qui met en relation l'équilibre des forces interfaciales avec un angle de contact stable, et reformulent le problème selon le théorème de Laplace en terme de différence de pression le long de la triple ligne

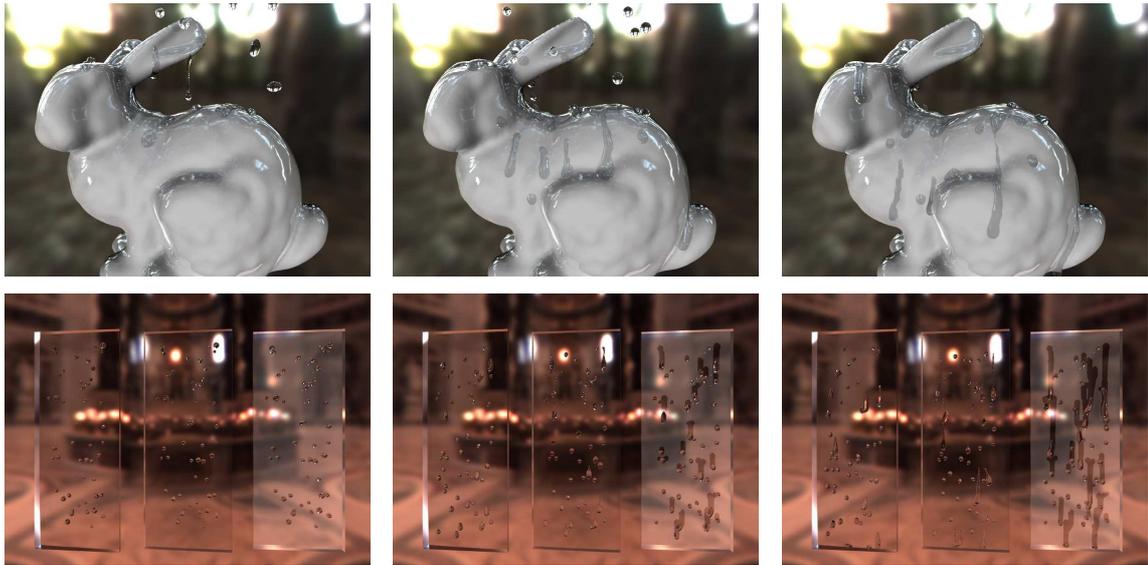


FIG. 1.23: Résultats obtenus par Wang *et al.* [WMT05] : la prise en compte des tensions de surface à la triple ligne de contact permet d'obtenir des gouttes de formes réalistes comme illustré dans ces deux animations.

de contact. Leur méthode utilise deux grilles 3D où un level set discrétise respectivement l'interface liquide-air et l'interface solide-air. Afin d'estimer les courbures de l'interface liquide-air-solide et donc les effets de tensions de surface à la triple ligne de contact, une troisième grille 3D est nécessaire dont les valeurs seront calculées dynamiquement selon l'évolution du liquide. Cette grille est appelée surface virtuelle, elle consiste à garder la partie de l'interface liquide-air non contenue dans le solide pour y modifier les valeurs intérieures au solide afin d'y refléter l'angle de contact stable désiré. Selon une équation trigonométrique proposée par les auteurs, la nouvelle distance des éléments inclus dans le solide et connexes à l'interface liquide-solide est calculée en prenant en compte l'angle de contact. La surface virtuelle est alors complétée par une technique d'extrapolation rapide [Set96] dans le solide pour un voisinage limité à quelques voxels. Lors de la résolution de l'équation de poisson pour la pression, les effets de tensions de surface sont obtenus en posant les conditions aux bords adéquates, les courbures étant estimées à partir de la surface virtuelle dans le cas de la présence de la triple ligne de contact et à défaut à partir de la grille représentant l'interface liquide-air [LGF04]. Les auteurs simulent de plus les phénomènes d'hysteresis de l'angle et du mouillage. Pour cela, une texture d'humidité est utilisée afin de déterminer les zones où les gouttes sont déjà passées, les paramètres d'angles de reculées, d'avancées et d'avancées en zone humide étant nécessaire à la simulation. A la triple ligne, les pressions relatives aux angles d'avancée (en zone humide si le voisinage de la texture d'humidité le dicte) et de reculée sont dans un premier temps calculées. Selon leurs signes respectifs est alors choisie la pression adéquate à poser en tant que condition au bord. Le phénomène de mouillage -soit la goutte est attirée par les zones humides en priorité- est reproduit avec succès étant donné que la pression de l'angle d'avancée en zone

humide est inférieure à celle de l'angle d'avancée. Des résultats impressionnant de réalisme sont illustrés dans la figure 1.23.

Zheng *et al.* [ZYP06] approfondissent les problématiques liées aux bulles en tenant compte de l'interface liquide très fine qui les entourent, cela afin de reproduire les phénomènes d'agglomération, de fusion et de rupture liés aux tensions de surface. Afin de passer outre la limitation des level set à représenter des interfaces très fines, les auteurs proposent d'employer des level set régionaux, soit un level set pour le liquide ainsi qu'un level set pour chacune des bulles en concordance avec des opérateurs à caractère géométrique. Ainsi, il n'est pas nécessaire d'utiliser une unique grille sur-échantillonnée. Des heuristiques servent ensuite à dicter le comportement de la membrane liquide des bulles, une première étant liée à sa durée de vie et la seconde à la distorsion qu'elle subit. Ces deux heuristiques sont formulées en tant que deux équations différentielles ordinaires qui une fois combinée détermineront l'instant à laquelle la membrane se rompra. La bulle considérée fusionnera alors avec sa voisine si présente, ou à défaut explosera à la surface liquide. Une contribution intéressante est le schéma d'intégration implicite utilisé pour les effets de tensions de surface, qui sont intégrés aux équations de Navier et Stokes en tant que forces extérieures [BKZ92]. En se basant sur un théorème de géométrie différentielle, une formulation implicite et donc stable donnant lieu à un système d'équations linéaires symétrique défini positif est obtenue, ce dernier se prêtant ainsi à une résolution rapide pour l'obtention de l'intensité des tensions de surface.

Il existe de nombreux travaux n'ayant pas trait de près ou de loin aux effets de tensions de surface, et l'on citera brièvement une sélection de ces contributions jugées comme importantes au domaine : Fedkiw *et al.* [FSJ01] limitent les effets de dissipations numériques en réinjectant quand nécessaire de façon artificielle des tourbillons dans le champs de vitesses. Carlson *et al.* [CMRBVHT02] s'intéressent à la fonte d'objets en variant la viscosité dynamiquement. Carlson *et al.* [CMT04] propose de gérer les interactions fluides-solides en considérant le solide tel un fluide ainsi que des multiplicateurs de Lagrange distribués. Guendelman *et al.* [GSLF05] simulent des interactions fluides avec des obstacles infiniment fins sur des grilles régulières de taille pratique. Batty *et al.* [BBB07] proposent une formulation variationnelle de l'équation de poisson pour la pression dont la résolution se veut rapide et efficace dans le cas de géométries solides non alignées sur la grille. Irving *et al.* [IGLF06] optimisent les temps de calcul de simulations liquides en combinant astucieusement grilles 2D et 3D. Losasso *et al.* [LSSF06] propose une méthode de simulation pour gérer plusieurs liquides en interaction avec changement de phase. Mihalef *et al.* [MUM⁺06] s'intéressent à l'ébullition en combinant un level set avec une grille VOF pour la conservation de volume, un changement de phase simulé physiquement étant responsable de la formation des bulles. Losasso *et al.* [LTKF08] combinent la méthode des particules level set avec des SPH afin de gérer respectivement les régions liquides dense et diffuse (par exemple, une vague déferlant avec ses éclaboussures et son écume). Les SPH ont aussi été

combinée à des grilles afin de simuler des bulles d'air dans l'eau [HLYK08], les SPH permettant d'obtenir de petites bulles au comportement réaliste lorsque la grille ne le permet plus.

Écoulements de fluides sur des surfaces

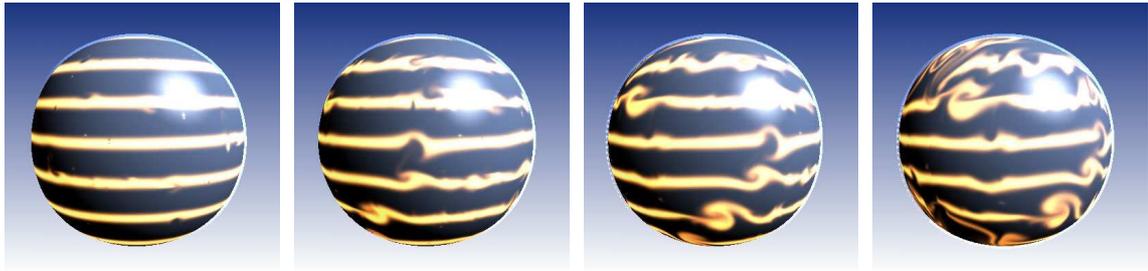


FIG. 1.24: Résultats obtenus par Stam [Sta03]

Stam [Sta03] propose pour la première fois une méthode dédiée à l'animation de fluide sur des surfaces arbitraires. Une surface est discrétisée en une union de quadrilatères auxquels des carreaux de Catmull-Clark sont appliqués afin d'obtenir une représentation continue de la surface. Les surfaces paramétriques courbes de Catmull-Clark ont non seulement été choisies dans un contexte esthétique, mais surtout pour leurs métriques intrinsèques permettant ainsi de redéfinir localement les opérateurs différentiels classiques. L'auteur adapte alors un de ses précédents travaux [Sta99] en redéfinissant les équations de Navier et Stokes en coordonnées curvilignes avec les conséquences qui en résultent, soit des opérateurs différentiels adaptés et des conditions aux bords prenant en compte la transition de carreaux. Chaque quadrilatère est discrétisé en une grille 2D régulière MAC pour lesquels seront stockées les variables relatives à la simulation. Une technique d'advection semi-lagrangienne est employée, les termes visqueux et de pression étant intégrés selon un schéma implicite par une méthode de gradient conjugué pré-conditionné. Les animations obtenus permettent ainsi de visualiser un champ de vitesses réaliste sur une surface courbe en observant le transport de densités (voir la figure 1.24).

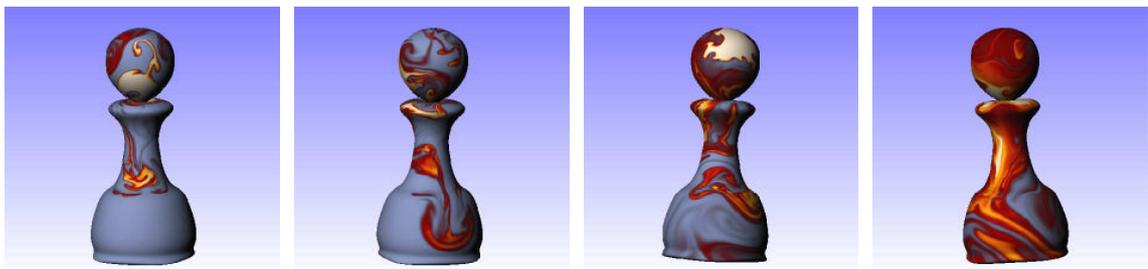


FIG. 1.25: Résultats obtenus par Shi et Yu [SY04]

Shi et Yu [SY04] s'intéressent aussi à la problématique des flots surfaciques, et pro-

posent de conserver les maillages usuels (triangulation d'objets) afin d'éviter les conversions de surfaces ainsi que les problèmes de distorsion inhérents dans de tels cas. Les vitesses sont discrétisées aux centres des triangles tandis que les champs de pressions et de densités le sont aux sommets du maillage. En redéfinissant l'opérateur de divergence pour la conservation de la masse et en usant de filtres d'interpolations adaptés ainsi que de transformations affines pour l'advection, les équations de Navier et Stokes sont en mesure d'être intégrés. Des animations interactives et convaincantes pour le transport de densités par un flot sont alors obtenues, des exemples de ces dernières étant montrées dans la figure 1.25.

Toujours dans la problématique des flots sur surfaces, un schéma général d'advection numériquement plus stable a été proposé dans [KLLR07], ce dernier pouvant s'intégrer à la méthode de Shi et Yu [SY04] comme mentionné par les auteurs. Toujours afin de réduire les effets de dissipation numérique, une simulation basée sur les équations de Navier et Stokes dans leur formulation rotationnelle a été proposée dans [ETK⁺07] où le cas des maillages triangulaires est également traité.

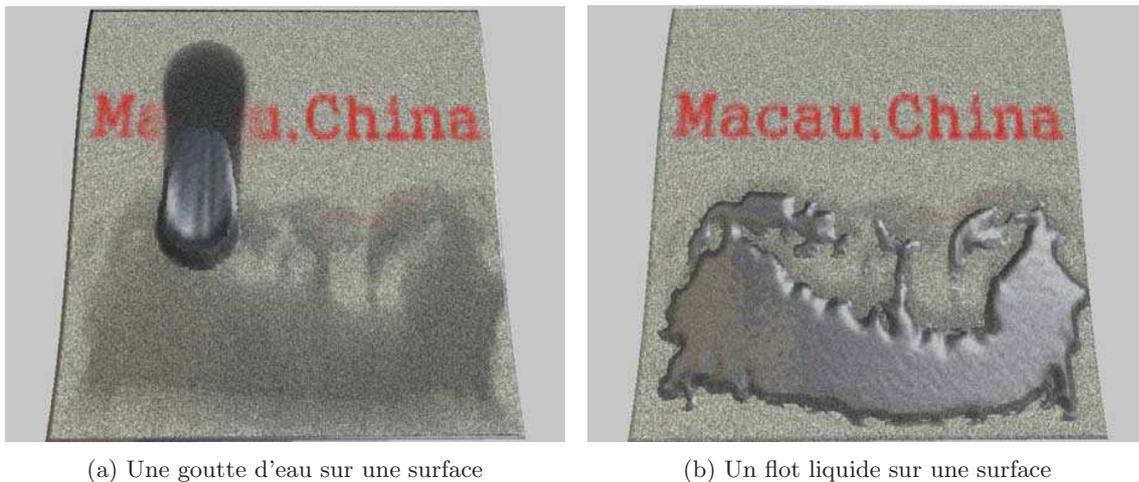


FIG. 1.26: Résultats obtenus par Liu *et al.* [LZLW05]

Liu *et al.* [LZLW05] proposent une méthode pour les flots de nature liquide sur une surface plane. Dans une même optique que Dorsey *et al.* [DPH96], les auteurs s'intéressent en partie aux phénomènes d'absorption, d'érosion et de déposition, qui sortent du cadre de cette thèse. D'autre part, ils adressent les spécificités des faibles écoulements liquides où les effets de tensions de surface dominent, et propose de prendre en compte ces derniers au sein des équations de Navier et Stokes. Leur solveur utilise une advection semi-lagrangienne avec une projection du champs de pression, la grille de simulation étant 2D régulière et de nature VOF. La surface solide sous-jacente pouvant présenter des irrégularités, une carte de hauteur servant à modéliser ces dernières permettra lors de la simulation de calculer une force de friction allant à l'encontre du flot. Les effets de tensions de surface sont

pris en compte par l'ajout d'une force extérieure dépendant de la courbure de l'interface. En dépit de sa nature physique, cette méthode ne permet pas d'obtenir un flot de forme convaincante et réaliste (voir la figure 1.26), le domaine bidimensionnel ainsi que la non prise en compte du phénomène de l'angle de contact en étant très certainement les causes principales.

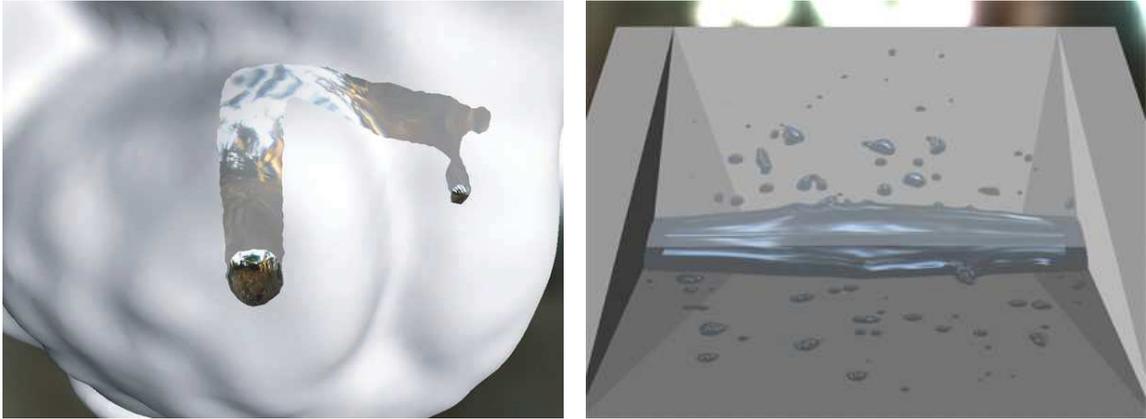


FIG. 1.27: Résultats obtenus par Wang *et al.* [WMT07]

Wang *et al.* [WMT07] proposent une approche basée sur les équations générales d'eaux peu profondes où le terme visqueux est négligé, mais les effets de tensions de surface pris en compte. Une surface courbe est discrétisée en une union de quadrilatères auxquels sera associée pour chacun une carte de hauteur, les hauteurs étant relatives à la normale locale de la surface. Les tensions de surfaces sont intégrées au solveur en tant que différence de pression, le cas spécifique de la triple ligne de contact étant quant à lui géré en adaptant la méthode de la surface virtuelle [WMT05] aux cartes de hauteurs. Une advection semi-lagrangienne ainsi qu'une intégration selon un schéma implicite pour les forces extérieures et le champs de pressions permettent d'obtenir des animations interactives. On notera cependant que du à la modélisation en un champs de hauteur du flot, les angles de contact possibles sont limités. Aussi, la qualité des résultats dépendent grandement de la complexité de la surface solide et de la finesse de sa discrétisation. Des résultats sont montrés dans la figure 1.27.

Chapitre 2

Une approche temps-réel pour les écoulements surfaciques

2.1 Introduction

Le premier objectif de cette thèse a été de concevoir une méthode pour l'animation de faibles écoulements liquides sur une surface solide, avec comme principale contrainte d'atteindre des performances temps-réel. Comme il a été précédemment vu dans l'état de l'art, les méthodes relatives à notre problématique peuvent se diviser dans un premier temps en deux catégories distinctes : les approches basées physiques et les approches empiriques.

D'un point de vue général, les approches basées physiques présentent l'avantage majeur de reproduire fidèlement un phénomène naturel dans toute sa complexité, à condition que les équations de bases soient agrémentées des termes adéquats (soit, dans le cas présent, des termes de tensions de surface à l'interface liquide-air ainsi qu'à la triple ligne de contact). Nous avons vu que la résolution numérique de de telles équations même dépourvues de termes supplémentaires [FM96, FF01, EMF02] nécessitent des temps de calculs très lourds, la synthèse de deux images successives pouvant d'autant plus impliquer plusieurs passes intermédiaires afin d'éviter des instabilités numériques. Dans le cas d'une simulation physique dédiée aux gouttes [WMT05], la complexité tant calculatoire que mémoire est de surcroît grandement augmentée, le temps de calcul moyen pour une image avoisinant facilement l'heure sur une machine récente. En vue de performances temps-réel, cette première constatation a eu pour conséquence d'orienter notre choix vers une algorithmique empirique. Dans ce contexte, il a fallu déterminer quelles simplifications effectuer quant au phénomène considéré, ainsi que quelles limitations dépasser relativement aux précédentes approches de même filiation.

Les méthodes empiriques présentées dans l'état de l'art peuvent se catégoriser en trois groupes selon qu'elles se focalisent sur :

1. la forme des gouttes [FHP98, YJC98, MM99, YJC99, TKY02]

2. la dynamique des gouttes [KKY93, KZYN96, KIY99, IGPP01, SDY02, YZZ04]
3. les effets du flot sur la surface [DPH96, LZLW05]

La méthode développée et décrite dans le présent chapitre se situe dans la seconde catégorie. Les travaux précédents posent en général de fortes hypothèses sur le matériau de la surface solide (du verre), sur la localisation du point de vue (centrée et derrière la surface) ainsi que sur la forme des gouttes (sphérique).

On propose de visualiser une surface plane librement orientable pouvant être translucide ou opaque, aucune contrainte n'étant imposée quant au point de vue. Les gouttes sont considérées après leurs impacts sur la surface, rejoignant ainsi un flot dont le comportement est dicté par l'advection des gouttes, la viscosité du liquide ainsi que la présence d'obstacles sur la surface.

La surface mouillée est modélisée par un quadrilatère (doté d'une texture 2D si de nature non translucide) auquel on associe une grille 2D régulière relative au flot liquide. Chaque cellule de cette grille contient la fraction de volume liquide locale à la surface ainsi qu'un vecteur vitesse 2D exprimé dans le plan tangent à la surface. La grille est dans un premier temps mise à jour en agrémentant d'un certain volume les cellules impliquées dans un impact de gouttes, une goutte étant modélisée par un disque. Un nouveau schéma d'advection sert alors à transporter les quantités liquides ainsi que le champs de vitesses, les vitesses étant ensuite intégrées dans le temps selon la seconde loi fondamentale de la mécanique. Ce schéma d'advection empirique permet contrairement à de précédentes approches [KKY93, KZYN96, KIY99, SDY02, YZZ04] une évolution souple et continue du flot indépendamment de la résolution employée pour la grille. Un effet de viscosité est obtenu en atténuant les vitesses relativement à une gaussienne dont les paramètres sont définis par l'utilisateur en fonction de l'effet souhaité. Les obstacles sont modélisés en tant qu'une carte de normales traditionnelle, le champs de vitesses du flot étant influencé lors de son intégration dans le temps par cette dernière. Cette carte de normales peut de plus permettre d'obtenir un comportement du flot non monotone comme observé phénoménologiquement, cela en induisant de légères perturbations (à l'aide de bruit par exemple) sur les directions ainsi que sur les normes des vecteurs. En considérant la grille des volumes liquides telle un champs de hauteur, la surface mouillée est visualisée par une approche de lancer de rayons basée image sur GPU, permettant ainsi un rendu tridimensionnel où réfraction et réflexion de la lumière sont prises en compte. On montre de plus comment intégrer simplement à la simulation deux effets visuels : l'absorption du liquide par une surface poreuse ainsi que le transport d'encre d'une surface décorée.

Finalement, l'algorithme de la simulation étant enclin à une parallélisation, une implémentation sur GPU est présentée, cette dernière achevant des performances 120 fois plus rapide en moyenne comparée à une approche CPU, permettant en conséquence d'obtenir des performances temps-réel.

2.2 Description de la méthode

La méthode proposée est de nature empirique, elle se concentre sur la génération d'animations de flots liquides surfaciques se voulant convaincantes visuellement, et cela à un faible coût calculatoire. Les effets de tensions de surface (entre autre responsables de la forme des gouttes ainsi que de l'attraction entre deux gouttes) sont en conséquence négligés dus à leurs complexités pour se focaliser principalement sur la dynamique du flot, certains effets physiques étant néanmoins considérés (viscosité, force de friction, mouillage de la surface etc.). Etant donné que la méthode sera intégralement décrite dans le cadre d'une implémentation sur GPU, on entendra de façon équivoque par texture 2D une grille 2D régulière.

On présentera dans un premier temps les données nécessaires au déroulement de la simulation, suivra ensuite une description détaillée de l'algorithme d'écoulement pour finalement expliciter la technique de rendu.

2.2.1 Modélisation de la surface et des obstacles

La simulation requiert en entrée les données et les paramètres suivants :

- un quadrilatère représentant la surface solide, dynamiquement ainsi que librement orientable dans l'espace durant la simulation. Ce dernier se situe dans le plan (O, x, y) en espace objet auquel lui est associé une matrice de modélisation M_M .
- une hauteur maximale relative au quadrilatère précédent le long de l'axe des profondeurs z en espace objet, permettant ainsi de borner le volume maximal que peut supporter la surface. Une matrice de mise à échelle M_S est associée à la boîte englobante de la surface afin de définir le changement de repère entre l'espace objet et l'espace tangent.
- une texture 2D dite liquide où chaque texel de coordonnées (i, j) a pour propriétés q_l la fraction volumique de liquide (normalisée entre 0 et 1), $\vec{v}_t = (v_x, v_y)$ le vecteur vitesse exprimé dans l'espace tangent (soit, le vecteur vitesse après projection sur le plan tangent (O, x, y)).
- une carte de normales modélisant les irrégularités géométriques sous-jacentes à la surface (par exemple, présence d'insalubrités ou d'un obstacle solide) possiblement perturbées (par exemple, à l'aide de bruit) en directions et normes si un comportement du flot non uniforme est désiré. Cette carte de normales entrera en considération lors de l'advection du flot de gouttes, et sera stockée sous la forme d'une texture 2D.
- une texture 2D classique dite décorative dans le cas d'une surface opaque. Cette dernière sera nécessaire afin de simuler la réfraction de la lumière.

Les différentes textures utilisées par la simulation (textures liquide, de normales et décorative) peuvent être de dimensions différentes. Cependant, afin d'éviter des effets de distorsion sur le flot lors de la visualisation, les textures utilisées en pratique sont de même rapport d'aspect que le quadrilatère de surface. Etant donné que le flot est modélisé par

la texture liquide, il est important de définir de façon appropriée la résolution de cette dernière. Dans le cas d'une grille trop grossière, les gouttes occuperont un volume réel trop important, nuisant ainsi grandement au réalisme des animations. Inversement, dans le cas d'une grille trop fine, bien que le flot puisse être décrit de façon détaillée, les complexités calculatoire et mémoire résultantes limiteraient grandement les performances. Comme tout simulateur de fluides basé sur des grilles, un choix raisonnable pour la discrétisation du domaine se doit de prendre en compte l'aire de la surface solide en concordance avec le diamètre minimal souhaité pour les gouttes.

2.2.2 Impact de gouttes sur la surface

D'un point de vue de la simulation, on ne considère les gouttes qu'une fois déjà entrées en collision avec la surface solide (les gouttes en suspension dans l'air ne sont pas modélisées car hors contexte), ainsi qu'après stabilisation (les splashes tout comme les oscillations de l'interface liquide-air ne sont pas pris en compte).

La première étape de la simulation consiste à ajouter à la texture liquide des informations volumiques relatives aux gouttes. Une goutte est modélisée par un disque plein avec comme propriétés son rayon et une valeur scalaire q_d déterminant la distribution volumique de la goutte dans l'espace tangent. Ainsi, la texture liquide est mise à jour en projetant sur la surface solide des disques de rayons donnés à des positions données. Les texels complètement recouverts voient leurs précédents volumes liquides q_l incrémentés de q_d , les texels limitrophes aux disques (et donc partiellement recouverts) ayant leurs valeurs précédentes q_l incrémentés de q_d pondérés par le rapport entre l'aire de la zone recouverte du texel sur l'aire totale du texel. Cette dernière considération permet de réduire les effets d'aliasage près de la triple ligne de contact. Lorsque la fraction volumique d'un texel excède la borne maximale, une troncature est effectuée afin d'éviter des instabilités numériques et garder une cohérence globale du flot. Des vitesses initiales peuvent aussi être définies, ces dernières étant prise en compte de la même façon que les volumes liquides.

Bien que simpliste, ce formalisme a été choisi pour sa complexité calculatoire moindre mais surtout parce que totalement implémentable sur GPU. En conséquence, des formes réalistes de gouttes prenant en compte les angles de contact ne peuvent ni être obtenues après impact, ni être préservées dans le temps par la simulation. Des méthodes plus sophistiquées et beaucoup plus lourdes d'un point de vue calculatoire sont nécessaires à cette fin [WMT05], ce qui va à l'encontre de nos motivations initiales. On notera cependant que la forme du flot obtenue par la présente méthode est suffisamment réaliste dans un contexte de performances temps-réel, comme illustré à travers les résultats.

2.2.3 Dynamique du flot de gouttes

Une fois que les gouttes ont été ajoutées à la texture liquide, les propriétés de chaque texel sont mises à jour, à commencer par les vecteurs vitesses. Ces derniers sont influencés

par :

- la force gravitationnelle, et possiblement d'autres forces extérieures
- le vecteur normal à la surface issu de la texture de normales
- la force de friction relative à la surface solide, dépendant du volume liquide local
- le vecteur vitesse à l'itération précédente

La force gravitationnelle $\vec{G} = (0, G_y, 0)$ est exprimée initialement en coordonnées monde, avec donc $G_y < 0$. A un pas de temps donné, la force gravitationnelle ainsi que toutes forces extérieures facultatives (par exemple, du vent, des obstacles mobiles etc.) doivent être exprimées dans l'espace tangent de la surface. Ainsi, pour toute force quelconque \vec{F} définie en coordonnées monde, ses coordonnées dans l'espace tangent \vec{F}_t sont obtenues après multiplication par la matrice de modélisation inverse M_M^{-1} suivie d'une mise à l'échelle par M_S :

$$\vec{F}_t = M_S * M_M^{-1} * \vec{F}$$

Selon la seconde loi fondamentale de la mécanique $\vec{a} = \frac{d^2x}{dt^2} = \frac{1}{m} \sum \vec{F}$, l'accélération \vec{a} correspond à la somme des forces extérieures divisée par la masse. Pour chaque cellule de la texture liquide, un vecteur d'accélération initial est calculé en trois dimensions relativement aux forces extérieures qui lui sont appliquées en son centre ainsi que relativement à sa masse (obtenue par multiplication du volume local q_l avec la densité volumique ρ). Ce vecteur d'accélération en trois dimensions sera ensuite modifié afin de successivement prendre en compte les obstacles, la force de friction exercée par le solide et la viscosité du liquide, pour finalement être projeté sur le plan tangent à la surface.

Pour chaque cellule (i, j) de la texture liquide, le vecteur d'accélération en espace tangent \vec{a}_t est transformé en fonction du vecteur normal à la surface issu de l'interpolation bilinéaire en (i, j) dans la texture de normales. Soit \vec{N} ce vecteur normal, le vecteur d'accélération devient alors :

$$\vec{a}_t = \vec{a}_{ext, t} - \left((1 - N_z) * (\vec{N} \cdot \vec{a}_{ext, t}) \right) * \vec{N} \quad (2.1)$$

où $\vec{a}_{ext, t} = \vec{G}_t + \frac{\sum \vec{F}_t}{q_l * \rho}$, \vec{F}_t représentant des forces extérieures et ρ la densité du liquide.

Selon l'intensité de la perturbation de la surface solide telle que dictée par la texture de normales ($N_z = 1$ symbolisant une surface lisse et inversement, $N_z = 0$ un obstacle insurmontable pour le flot), le vecteur d'accélération est projeté proportionnellement à N_z sur le plan tangent de la surface. L'équation 2.1 est en fait une orthogonalisation de Graham Schmidt où une pondération par l'intensité de la perturbation surfacique a été ajoutée, plus grande sera la pente relative à l'axe des profondeurs, plus le flot sera influencé par la

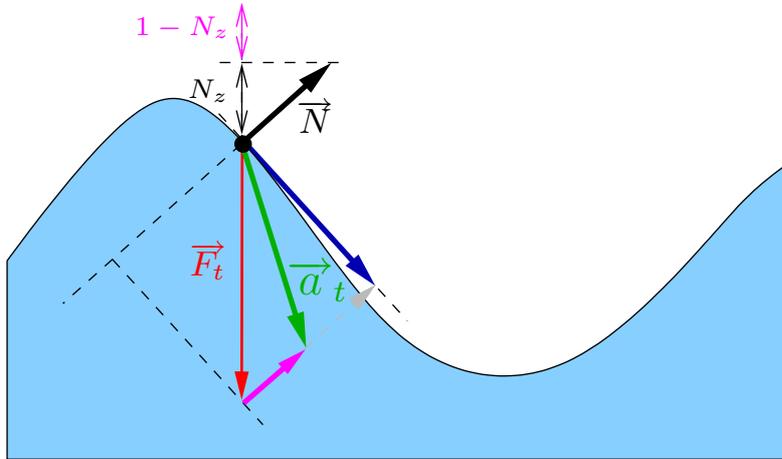


FIG. 2.1: Perturbation du vecteur accélération le long du vecteur normal de la surface. L'intensité de la perturbation est linéairement proportionnelle à $(1 - N_z)$

présence de l'obstacle. La figure 2.1 illustre cette transformation.

Une force de friction est définie comme étant une force s'opposant à la direction de mouvement, sa norme dépendant de paramètres physiques. Dans le contexte d'écoulements liquides, plus petite sera la quantité liquide, plus grande devra être la force nécessaire à son déplacement en raison des tensions de surface [dG85]. On simule cet effet de façon empirique en atténuant le vecteur accélération précédent par une fonction R dépendant du volume liquide q_l . En vue d'obtenir divers comportements de flots avec de moindres paramètres à définir, une fonction inspirée d'une gaussienne classique a été utilisée afin de définir un profil d'atténuation non linéaire :

$$R(q_l) = \lambda \exp\left(-\frac{(1 - q_l)^n}{\sigma}\right) \quad (2.2)$$

En utilisant différentes valeurs d'amplitude pour le paramètre λ (fixe lors de la simulation), un effet de viscosité peut être simulé (plus faible sera la valeur de λ , plus forte sera la viscosité du liquide).

Dans le cadre de notre implémentation, les paramètres de la fonction R ont été fixés de façon expérimentale à $n = 4$ et $\sigma = 0.1$. La figure 2.2 illustre le filtre utilisé dans les résultats avec $\lambda = 1$ (courbe en noir), ainsi que trois autres filtres (en rouge, vert et bleu) pour les animations satisfaisantes qu'ils permirent d'obtenir.

Après mise à l'échelle du vecteur d'accélération \vec{a}_t relativement à $R(q_l)$, le vecteur d'accélération final $\vec{a}_{t(x,y)}$ est obtenu par projection sur le plan (O, x, y) en espace tangent.

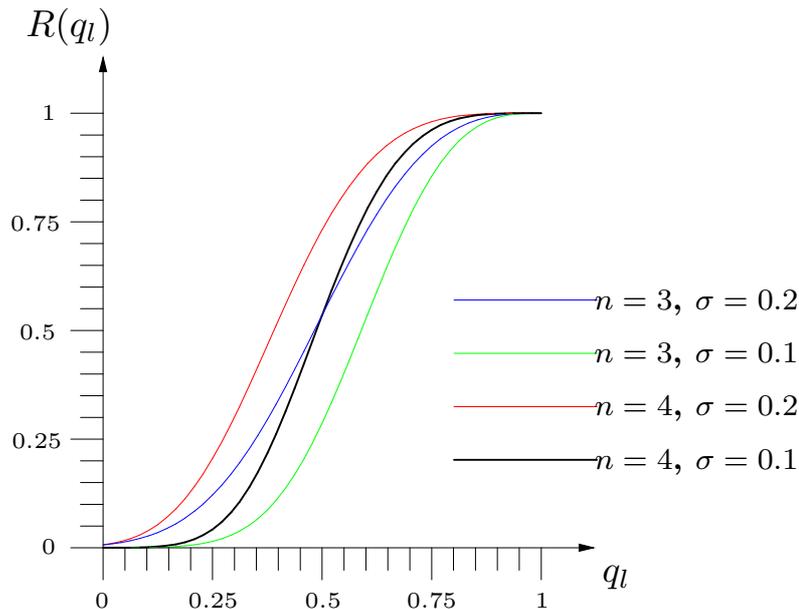


FIG. 2.2: Tracé de la fonction de mise à échelle utilisée afin de simuler les forces de friction et le paramètre de viscosité λ (ici, $\lambda = 1$ a été utilisé). De faibles volumes liquides voient leur accélération réduite tandis que de plus larges volumes auront une accélération accrue.

Soit δT le temps écoulé entre deux pas de temps de la simulation, T étant le temps à l'itération précédente. Les vitesses sont mises à jour par une intégration explicite :

$$\vec{v}_t(T + \delta T) = \vec{v}_t(T) + \delta T * \vec{a}_t(x,y) \quad (2.3)$$

Le nouveau champs de vitesses une fois déterminé, chaque cellule a son volume liquide modifié en raison du phénomène d'advection. Un schéma d'intégration explicite empirique est utilisé à cette fin, les volumes étant transportés par le champs de vitesses à l'itération précédente. La présente méthode étant destinée à une implémentation sur processeurs graphiques, une intégration explicite dans l'espace ne serait pas un choix approprié en raison de l'architecture actuelle des GPU. Afin de remédier à cette problématique tout en gardant une approche basée grille (et non particules), une exploration du voisinage 8 connexes est effectuée afin de transporter les variables. Une conséquence directe de cette reformulation est la pénalisation des advections distantes rendues impossibles, les cellules ne faisant pas partie du voisinage 8 connexes ne pouvant contribuer à le mise à jour du volume local. En dynamique des fluides numérique, les schémas d'intégration explicites sont considérés comme étant simples d'implémentation ainsi que de faibles coûts calculatoires, mais cependant très enclins à des instabilités numériques. Afin de limiter ces dernières et assurer une convergence, des limitations sur le pas de temps imposées par la condition de Courant Friedrichs Lewy (CFL) sont nécessaires. On s'inspire de la condition

de CFL proposée en synthèse d'images pour l'animation de liquides [FM96] en raison des similarités entre leur processus d'advection et celui présenté ici. Cette condition de CFL statue que toute composante des vecteurs de déplacement (soit, $v_t * \delta T$) doit être inférieure à l'espacement inter-cellules $\Delta\tau$, ce qui se traduit par $\delta T < \Delta\tau / \max_{x,y}[v_t]$. De par sa nature, notre processus d'advection prends en considération les cellules distantes au plus de $\sqrt{2}\Delta\tau$, respectant ainsi la précédente condition de CFL. L'advection sera néanmoins faussée dans le cas de vecteurs de déplacement trop longs, cas pour lequel on propose deux stratégies :

1. la longueur des vecteurs de déplacements doit être inférieure ou égal à $\sqrt{2}\Delta\tau$, une normalisation des vecteurs étant nécessaire pour les cellules violant cette contrainte. Cette considération permet d'assurer une advection stable numériquement parlant, mais nuira potentiellement au réalisme de l'animation.
2. une subdivision de δT en pas de temps intermédiaires afin de satisfaire la condition de CFL. Dans un premier temps, une recherche de la composante maximale du champs de vitesses est effectuée et on calcule le rapport $r = (\max_{x,y}[v_t] * \delta T) / \Delta\tau$. $\lfloor r \rfloor + 1$ définit le nombre de pas de temps intermédiaires, $\delta T_{intermediaire} = \delta T / r$ étant le pas de temps à employer pour les $\lfloor r \rfloor$ premières passes intermédiaires tandis que $\delta T_{final} = (\delta T / r) * (r - \lfloor r \rfloor)$ sera le pas de temps à employer pour finaliser l'advection. Ces passes intermédiaires (invisibles à l'utilisateur car internes à la simulation) assurent une certaine convergence tout en respectant la dynamique du flot liquide, la complexité calculatoire résultante étant en revanche plus grande.

Le choix parmi ces deux solutions dépend du contexte dans lequel la simulation est employée, et consiste en un compromis entre réalisme et performance. Si les performances priment sur le réalisme tel que dans un jeu vidéo, la première alternative devrait être préférée. Inversement, si un certain réalisme doit être reproduit tel que dans un simulateur de conduite, la seconde solution serait un choix plus judicieux.

La figure 2.3 montre un exemple de la contribution par advection avant d'une cellule envers son voisinage ainsi qu'à elle-même. Une cellule ne contribue à aucune de ses voisines si la norme du vecteur de déplacement est nulle, et inversement, jusqu'à trois de ses voisines. Les aires d'intersections entre la cellule courante translatée et ses voisines sont calculées relativement aux composantes du vecteur de déplacement, à la localisation des centres des cellules voisines ainsi qu'à l'échantillonnage de la grille. Dans un souci de clarté, on considérera que les centres de cellules voisines sont espacées d'une unité selon chaque dimension.

En inversant le processus, le voisinage 8 connexes d'une cellule est exploré, voisinage dont les contributions individuelles sont calculées de la sorte :

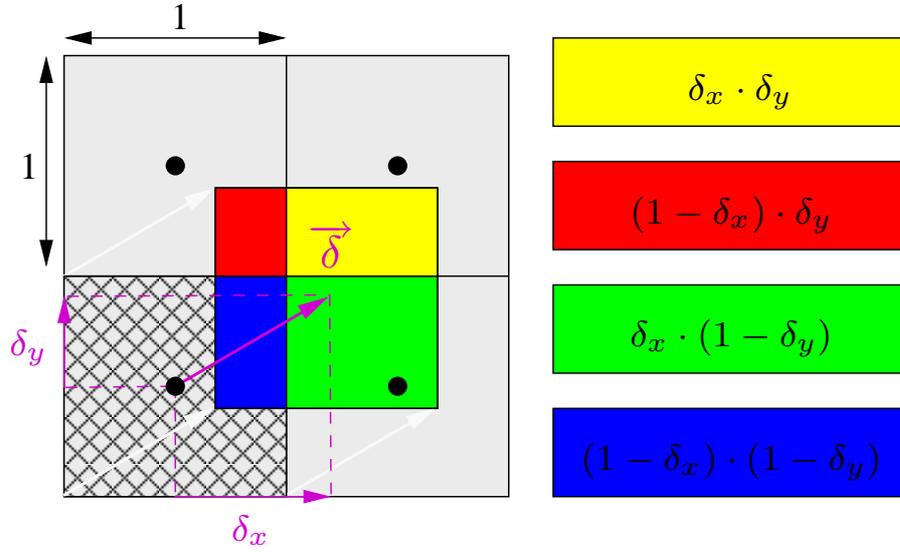


FIG. 2.3: Contribution avant d'une cellule par advection. La cellule en bas à gauche est transportée par (δ_x, δ_y) , et intersecte 3 cellules voisines. Les aires d'intersections avec ses voisines ainsi qu'avec elle-même sont explicitées à droite.

$$\begin{aligned}
 c(i,j) &= (1 - |\delta_x|) * (1 - |\delta_y|) * q_l(i,j) \\
 c(i-1,j+1) &= \max(0, \delta_x) * |\min(0, \delta_y)| * q_l(i-1,j+1) \\
 c(i,j+1) &= (1 - |\delta_x|) * |\min(0, \delta_y)| * q_l(i,j+1) \\
 c(i+1,j+1) &= |\min(0, \delta_x)| * |\min(0, \delta_y)| * q_l(i+1,j+1) \\
 c(i-1,j) &= \max(0, \delta_x) * (1 - |\delta_y|) * q_l(i-1,j) \\
 c(i+1,j) &= |\min(0, \delta_x)| * (1 - |\delta_y|) * q_l(i+1,j) \\
 c(i-1,j-1) &= \max(0, \delta_x) * \max(0, \delta_y) * q_l(i-1,j-1) \\
 c(i,j-1) &= (1 - |\delta_x|) * \max(0, \delta_y) * q_l(i,j-1) \\
 c(i+1,j-1) &= |\min(0, \delta_x)| * \max(0, \delta_y) * q_l(i+1,j-1)
 \end{aligned} \tag{2.4}$$

La mise à jour des propriétés de la cellule de centre (i, j) est considérée ici. $c(m, n)$ dénote le volume de liquide quittant la cellule (m, n) et arrivant en la cellule (i, j) . Le vecteur de déplacement $\vec{\delta}$ est défini pour chaque cellule voisine (m, n) tel que $\vec{\delta} = (\delta_x, \delta_y) = \vec{v}_i(m, n) * \delta T$.

Les termes de l'équation 2.4 peuvent être compris intuitivement :

- $1 - |\delta_{axis}|$ signifie que la cellule voisine demeure sur le même axe que la cellule centrale, sa contribution étant alors inversement proportionnelle à la longueur du déplacement. Dans un tel cas, une contribution est toujours de rigueur sauf dans le cas d'un déplacement nul.
- $\max(0, \delta_{axis})$ signifie que la cellule voisine doit se situer du côté positif de l'axe considéré (soit, Nord ou Est), auquel cas la contribution sera proportionnelle à la

longueur du déplacement, la contribution étant nulle sinon.

- $|\min(0, \delta_{axis})|$ signifie que la cellule voisine doit se situer du côté négatif de l'axe considéré (soit, Sud ou Ouest), auquel cas la contribution sera proportionnelle à la longueur du déplacement, la contribution étant nulle sinon.

Une telle formulation permet une implémentation efficace sur les processeurs graphiques de génération actuelle en raison des instructions nativement supportées (valeur absolue, minimale et maximale) utilisées ici. Des instructions conditionnelles auraient du être employées à défaut, instructions significativement plus lentes à l'exécution et donc pénalisantes.

2.2.4 Effets Spéciaux

Absorption

D'un point de vue phénoménologique, l'absorption d'un liquide par une surface poreuse modifie de façon complexe les propriétés optiques de la surface [DPH96]. Fidèle à nos motivations, on propose de simuler simplement et empiriquement cet effet en considérant que plus une parcelle de surface sera humide, plus ses informations colorimétriques seront atténuées. A cette fin, la texture liquide est dotée d'une quatrième composante (les trois premières restant inchangées) représentant la quantité liquide absorbée localement par la surface, cette valeur étant bornée entre 0 et 1. Le processus de simulation demeure identique excepté qu'avant l'advection, la fraction volumique qui sera absorbée par la surface est déterminée. Pour une cellule donnée, cette quantité liquide retenue dépend du volume liquide courant q_l , du temps, de la nature de la surface $\alpha_{surface}$ (une valeur nulle symbolisant une surface totalement hydrophobe et inversement, une valeur de 1 totalement hydrophile) ainsi que du volume liquide déjà absorbé q_a . La formule que nous avons utilisé est la suivante :

$$q'_a(T, \delta T) = \min(1 - q_a(T), A(q_l(T), \alpha_{surface}) * \delta T) \quad (2.5)$$

$$\begin{aligned} q_l(T) &= q_l(T) - q'_a(T, \delta T) \\ q_a(T + \delta T) &= q_a(T) + q'_a(T, \delta T) \end{aligned} \quad (2.6)$$

où q'_a est la quantité liquide absorbée relativement au laps de temps écoulé δT . Cette quantité dépend de la fonction d'absorption A qui peut soit être définie en tant que donnée utilisateur (fonction pré-calculée et discrétisée), soit en tant que fonction procédurale similairement à l'équation 2.2. La fonction d'absorption A prends en entrée le volume liquide courant ainsi que l'affinité de la surface afin d'approcher le volume liquide absorbé par unité de temps. Une telle formulation permet de décrire de façon non uniforme les

propriétés poreuses d'une surface en usant d'une carte d'affinité (texture 2D à une composante scalaire) afin de rendre certaines régions plus hydrophiles que d'autres (par exemple, une vitre recouverte partiellement de sable). Le paramètre q_l peut être utilisé afin de décrire un profil d'absorption non linéaire : plus le volume liquide sera important dans une cellule, plus la proportion absorbée sera élevée. Une fois le volume liquide théoriquement absorbé calculé par intégration explicite dans le temps du résultat de la fonction d'absorption A , ce dernier est tronqué relativement à la quantité liquide déjà absorbée par la cellule si nécessaire (voir l'équation 2.5), cela afin d'éviter des incohérences tant numériques que visuelles. Finalement, les volumes liquides courant q_l et absorbé q_a sont mis à jour pour chaque cellule selon l'équation 2.6, et le reste de la simulation se déroule sans autres changements.

La valeur d'absorption sera utilisée lors du rendu d'une surface opaque afin de refléter le phénomène (voir la section 2.3.2).

Transport d'encre

Durant la phase d'advection, un effet de transport d'encre peut être achevé en advectant les informations de couleurs de la texture de décoration avec le flot. Une impression de surface se délavant progressivement peut ainsi être obtenue, des gouttes de couleurs se déplaçant sur la surface. Afin de simuler ce phénomène, une nouvelle texture 2D dite d'encre dotée de quatre composantes est nécessaire, la texture liquide ayant quant à elle quatre composantes tel que pour l'effet d'absorption (voir la partie 2.2.4). Les trois premières composantes de la texture d'encre représentent une information de couleur (r, g, b) par cellule, la quatrième déterminant la densité de couleur actuellement transportée par le flot et sera initialisée à 0. La quatrième composante de la texture liquide détermine la densité de couleurs restante dans la texture de décoration par cellule, et sera initialisé à 1. Les texture liquide et décorative pouvant être de résolutions différentes, cette dernière est rediscrétisée afin d'être de même dimension que la texture liquide. Redéfinie par filtrage (un filtre cubique a été utilisé pour les résultats), cette texture sera utilisée lors de l'advection des couleurs, les nouveaux texels étant les moyennes (si réduit) ou les interpolations (si agrandi) de la texture décorative de base. Cette étape n'est effectuée qu'une seule fois préalablement à la simulation.

Durant la simulation, une fois les gouttes ajoutées à la texture liquide et le nouveau champs de vitesses déterminé, une densité de couleur est calculée relativement à la texture décorative qui contribuera à la texture d'encre. Pour une cellule donnée, la texture décorative peut émettre une densité de couleur $\alpha_{pattern}$ calculée par une fonction semblable à la fonction d'absorption de la partie 2.2.4. Cependant, afin de prévenir une saturation des couleurs dans la texture d'encre, la densité de couleur transportée actuellement α_{ink} définit la densité maximale que la cellule peut encore recevoir. La densité finale émise $\alpha_{emitted}$ est déterminée comme étant le minimum des valeurs $\alpha_{pattern}$ et $(1 - \alpha_{ink})$, et sera la valeur utilisée pour mettre à jour les données de la simula-

tion. L'information de couleur de la texture décorative est ensuite pré-multipliée par $\alpha_{emitted}$ et ajoutée au précédent triplet de couleurs (r, g, b) de la texture d'encre. La valeur $\alpha_{emitted}$ est ensuite soustraite de la quatrième composante de la texture décorative ($\alpha_{pattern}(T + \delta T) = \alpha_{pattern}(T) - \alpha_{emitted}(\delta T)$), puis ajoutée à la quatrième composante de la texture d'encre ($\alpha_{ink}(T + \delta T) = \alpha_{ink}(T) + \alpha_{emitted}(\delta T)$).

Lors de l'advection, les quadruplets de la texture d'encre seront transportés par le flot de la même façon que les variables relatives au liquide : dans l'équation 2.4, $q_l(m, n)$ est consécutivement substitué par chaque composante de la texture d'encre, soit r , g , b et α_{ink} . En conséquence, l'advection des textures liquide et d'encre étant dictée par le même champs de vitesses, le flot d'encre suivra rigoureusement le flot liquide, donnant l'illusion de gouttes liquides colorées. La texture d'encre sera utilisée lors du rendu d'une surface opaque (voir la partie 2.3.2) afin de donner de l'impression d'une surface peinte progressivement délavée.

2.3 Implémentation

2.3.1 Simulation

Etant donné que la phase d'advection est la partie la plus lourde de notre méthode d'un point de vue calculatoire, son optimisation est importante afin d'obtenir des performances temps-réel. Remarquant la nature parallélisable de l'algorithme (les variables de la simulation pouvant être mises à jours indépendamment) et souhaitant en tirer partie, une implémentation de la simulation sur GPU a été effectuée pour ses avantages significatifs en terme de performances.

Notre méthode a été implémentée pour la librairie graphique OpenGL en concordance avec le langage de programmation des processeurs graphiques GLSL. La norme « Shader Model 3.0 » est requise, ainsi que les extensions OpenGL suivantes :

- `GL_EXT_framebuffer_object`, afin d'effectuer un rendu dans une texture.
- `GL_ARB_texture_float`, afin de manipuler des textures en précision flottante.
- `GL_ARB_texture_cube_map` pour l'environnement mapping.
- `GL_ARB_draw_buffers` afin d'effectuer plusieurs rendu dans différentes cibles en une seule passe (cette dernière ne sera utilisée que dans le cas de transport d'encre).

La texture liquide est définie en précision flottante afin de prévenir des effets de dissipation numérique due à des accumulations d'erreurs, une précision limitée à 16 bits permettant d'obtenir des valeurs interpolées par la carte graphique à un coût négligeable.

Pour un pas de temps, quatre passes sont nécessaire au déroulement de la simulation ainsi qu'au rendu. Une opération simultanée de lecture et écriture sur une même cible n'étant pas tolérée sur les GPU actuelles, la texture liquide (ainsi que la texture d'encre si

besoin) est allouée deux fois. Ce type de textures sont communément dénommées textures ping pong. Dans le cas d'un effet de transport d'encre, on tire partie du fait que les textures liquide et d'encre sont de même dimensions en utilisant un tampon auxiliaire, ce dernier permettant une mise à jour simultanée des deux textures lors d'une seule passe. L'algorithme multi-passes est alors le suivant :

– **Passe 1** : Ajout de gouttes

Déclarer la texture liquide courante en tant que cible.

Activer le mélange des couleurs en mode additif pour cette passe (fonction GPU de mélange avec les paramètres `GL_ONE` pour la source et `GL_ONE` pour la destination).

Activer l'anti-aliasage de points afin de gérer le cas des fragments en périphérie des gouttes. Projeter des points de tailles et positions données munis d'attributs appropriés (volume et vitesse initiale des gouttes).

Echanger la texture liquide

– **Passe 2** : Calcul du champs de vitesses

Déclarer la texture liquide courante en tant que cible.

Projeter un quadrilatère de la taille de la fenêtre image. Pour chaque fragment, calculer le vecteur d'accélération relativement à δT (temps écoulé depuis la dernière image) et mettre à jour les vitesses.

Dans le cas d'un effet d'absorption, mettre à jour le volume liquide absorbé par le fragment.

Dans le cas d'un effet de transport d'encre, mettre à jour la densité restante de la texture décorative ainsi que les attributs relatifs à la texture d'encre pour le fragment.

Echanger la texture liquide

– **Passe 3** : Advection du flot

Choisir une des deux stratégies pour le pas de temps tel que décrit dans la partie 2.2.3, et effectuer des passes intermédiaires si nécessaire.

Déclarer la texture liquide courante en tant que cible.

Projeter un quadrilatère de la taille de la fenêtre image. Pour chaque fragment, parcourir le voisinage 8 connexes, calculer leurs vecteurs de déplacement selon leurs vitesses et δT afin de déterminer leurs contributions au fragment courant.

Mettre à jour les volumes liquides.

Dans le cas d'un effet de transport d'encre, transporter les quatre attributs d'encre avec le flot.

Echanger la texture liquide

– **Passe 4** : Rendu

Déclarer le tampon d'image standard comme cible, rendre la surface mouillée.

2.3.2 Rendu

Afin de visualiser la surface mouillée, il est nécessaire de connaître la localisation de l'interface liquide-air. La texture liquide utilisée par la simulation représente la distribution volumique du liquide sur une grille et hérite en conséquence de propriétés inhérentes aux méthodes VOF (Volume Of Fluid). Dans le cadre de méthodes VOF, la reconstruction de l'interface 3D nécessite de déterminer l'intersection d'un plan avec un cube soumise à une contrainte volumique, processus coûteux d'un point de vue calculatoire [GLN⁺99]. Etant donné que notre méthode se veut temps-réel, on s'inspire d'une méthode précédente [HK03] où le champs volumique est directement mis en relation avec un champs de distance, et on considère simplement la hauteur liquide de chaque cellule comme étant linéairement proportionnelle à son volume. Cette simplification directe permet alors d'employer un rendu par une approche de displacement mapping basée image dans le même esprit que [POC05] qui opère sur des cartes de hauteurs. En dépit de l'évaluation grossière du champs de hauteur, les résultats obtenus en pratique démontrent que pour de faibles volumes, cette approximation est efficace tant qualitativement qu'en terme de performances.

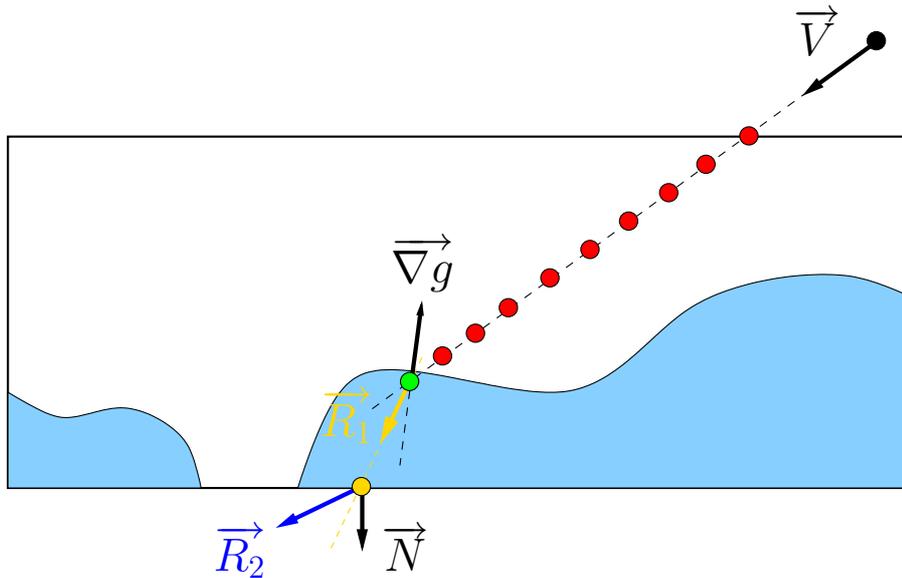


FIG. 2.4: Exemple de rendu d'une surface mouillée. Une recherche linéaire (points rouges) est effectuée jusqu'à ce qu'une intersection approchée avec l'interface liquide soit trouvée (point vert). Après réfraction du vecteur de vue selon le gradient à l'interface, l'intersection avec la surface est déterminée (point jaune) et définit les coordonnées de textures réfractées pour la texture décorative associée à une surface opaque. Dans le cas d'une surface translucide, une seconde réfraction par le vecteur normal à la surface permettra d'indexer (vecteur bleu) une cubemap de l'environnement de la scène.

Le processus de rendu diffère selon l'orientation de la surface relativement au point de

vue ainsi que selon le type de matériau considéré (translucide ou opaque). On définit dans un premier temps les techniques et les formules communes à chacun de ces cas :

- Calcul du gradient

L'information de normale à l'interface liquide-air est extrait de la texture liquide en tant que produit vectoriel de différences centrées, qui résulte en un gradient $\vec{\nabla}g = (-\frac{\partial q_l}{\partial x}, -\frac{\partial q_l}{\partial y}, 1)$.

- Réfraction

Soit un vecteur incident \vec{I} et un vecteur normal \vec{N} à l'interface, le vecteur réfracté \vec{R} selon le rapport eta des indices de réfraction de deux milieux est calculé par une méthode vectorielle respectant la loi de Snell :

$$k = 1 - eta^2 * (1 - (\vec{N} \cdot \vec{I})^2)$$

Si $k < 0$, alors $\vec{R} = (0, 0, 0)$. Sinon :

$$\vec{R} = eta * \vec{I} - (eta * (\vec{N} \cdot \vec{I}) * \sqrt{k}) * \vec{N}$$

Des indices de réfraction de 1 et de 1.333 ont respectivement été utilisés pour l'air et l'eau.

- Intersection d'un rayon avec une surface plane

Considérons une surface plane dans son espace tangent. Cette surface se situe dans le plan (O, x, y) à une profondeur nulle. Pour tout rayon défini en espace tangent par un point $\vec{p} = (x_p, y_p, z_p)$ associé à un vecteur $\vec{D} = (x_D, y_D, z_D)$, l'intersection du rayon avec la surface est calculée telle que $\vec{p}_{int} = \vec{p} + \frac{-z_p}{z_D} \cdot \vec{D}$.

- Intersection d'un rayon avec une carte de hauteur

La procédure utilisée dans le cas présent est inspirée de l'approche [POC05], que l'on décrit brièvement dans ce qui suit. La surface plane associée à un champs de hauteur définit une boite englobante dont les faces avant sont projetés dans un premier temps. Pour chaque fragment activé, un pas de progression approprié est calculé relativement à la résolution de la carte de hauteur et à la pente du vecteur de vision défini en espace tangent. Partant de l'intersection frontale de la boite englobante, le rayon est parcouru incrémentalement jusqu'à ce que sa hauteur devienne inférieure à sa valeur correspondante dans la carte de hauteur. Les auteurs proposent de plus d'effectuer une recherche dichotomique afin de converger avec plus de précision vers le point d'intersection. Cependant, dans le cadre de la présente méthode, un tel raffinement s'est avéré inutile en pratique étant donné une amélioration des résultats non pertinente qualitativement parlant. On explique ce dernier point par la nature fine des cartes de hauteurs découlant des textures liquides.

La boite englobante de la surface mouillée est construite tel que chaque face est orientée afin que son vecteur normal pointe vers l'extérieur de la boite, excepté pour la face inclus dans le plan (O, x, y) , le vecteur normal pointant vers l'intérieur dans ce cas la. Chaque sommet de la boite englobante est doté de coordonnées de texture 3D normalisées équivalentes



FIG. 2.5: Une ville vue de nuit derrière une vitre sur laquelle tombe de la pluie (364 images par seconde). On remarquera la distorsion de la scène induite par la réfraction des gouttes liquides sur la vitre.

aux coordonnées en espace tangent. La matrice de modélisation ainsi que la position de la caméra sont mises à disposition des programmes GPU par passation de variables uniformes.

Surfaces translucides

Dans le cas de surfaces translucides, on approche la réfraction de la scène considérée comme distante par le biais d'une technique d'environnement mapping. En conséquence d'une telle hypothèse, la couleur des fragments obtenus après réfraction ne dépendent pas de la localisation du point à partir duquel a été réfracté le vecteur incident. Une cubemap (prédéfinie ou pouvant être calculée à la volée à chaque pas de temps de la simulation) a été utilisée à ces fins car supportée matériellement sur GPU. La réfraction par le matériau de la surface est négligée, les surfaces étant considérées comme infiniment fine. Pour des raisons évidentes, les effets d'absorption et de transport d'encre ne seront pas abordés dans la présente partie. On détaille maintenant le rendu d'une surface translucide. Après projection des faces de la boîte englobante, chaque fragment exécutera un programme GPU qui se comportera différemment selon que le fragment émane d'une face arrière ou avant. Si le fragment est originaire d'une face arrière, on teste à la position courante si une valeur positive est présente dans la texture liquide. Si tel n'est pas le cas, cela signifie alors que la surface est localement sèche et on arrête le traitement du fragment. Sinon, un volume liquide étant présent, le vecteur de vue incident est réfracté en espace tangent avec

la normale à la surface $(0, 0, 1)$ selon le ratio d'indices de réfraction air-liquide. S'ensuit une recherche linéaire de l'intersection du nouveau vecteur de vue après réfraction avec l'interface liquide-air. Cette intersection trouvée, le vecteur de vue courant est à nouveau réfracté avec le gradient à l'isosurface en utilisant cette fois-ci le ratio d'indices de réfraction liquide-air. Le vecteur résultant est transformé en coordonnées monde pour finalement indexer la cubemap de la scène, déterminant ainsi la couleur du fragment.

Le processus est similaire dans le cas d'un fragment généré par une face avant, comme illustré dans la figure 2.4. On cherche dans un premier temps l'intersection du vecteur de vue avec la texture liquide. Si une intersection est trouvée (point vert dans la figure), le gradient à l'interface est calculé pour être utilisé dans la réfraction du vecteur de vue selon le rapport d'indices de réfraction air-liquide. Le vecteur résultant (\vec{R}_1 dans la figure) est immédiatement réfracté à nouveau en espace tangent avec la normale de surface $(0, 0, 1)$ selon un rapport d'indices de réfraction liquide-air. Ce vecteur final, illustré en bleu et dénommé \vec{R}_2 sur la figure, a uniquement été translaté du point d'intersection initial pour une clarté de présentation, la profondeur du flot liquide n'étant pas prise en compte dans le cas présent. Après une transformation en coordonnées monde du vecteur, une cubemap est indexée pour simuler la réfraction de l'environnement. Un exemple de rendu pour une surface translucide mouillée est montrée dans la figure 2.5

Surfaces opaques

Une surface opaque se différencie principalement d'une surface translucide par sa texture décorative qui lui est associée ainsi que par la possibilité de simuler un effet d'absorption ou un effet de transport d'encre. La boîte englobante est tout d'abord projetée, les fragments émanant d'une face arrière n'étant pas traités. Dans le cas d'un fragment de face avant, l'intersection du vecteur de vue avec l'interface liquide-air est recherchée en espace tangent. Si aucune intersection n'a été trouvée, le dernier échantillon le long du rayon gît sur une zone sèche, et les deux premières composantes des coordonnées de texture sont utilisées afin d'indexer la texture décorative (on rappelle que les coordonnées de texture de la boîte englobante sont définies en espace tangent). Ce dernier cas est identique à une technique standard de placage de texture 2D, l'absence de liquide impliquant une non réfraction. Dans le cas d'une intersection entre le vecteur de vue et l'interface (point vert sur la figure 2.4), le gradient à l'isosurface est calculé. Ce dernier est utilisé afin de réfracter le vecteur de vue selon le rapport d'indices de réfraction air-liquide (\vec{R}_1 sur la figure). On détermine ensuite l'intersection du vecteur de vue réfracté avec la surface (point jaune sur la figure). Les deux premières coordonnées du point d'intersection sont utilisées afin d'indexer la texture décorative, ainsi que la texture d'absorption ou la texture d'encre si besoin. La couleur finale du fragment est obtenue selon le type de la simulation :

- Simulation standard

La couleur du fragment est simplement la couleur du texel correspondant dans la texture décorative.

- Effet d'absorption

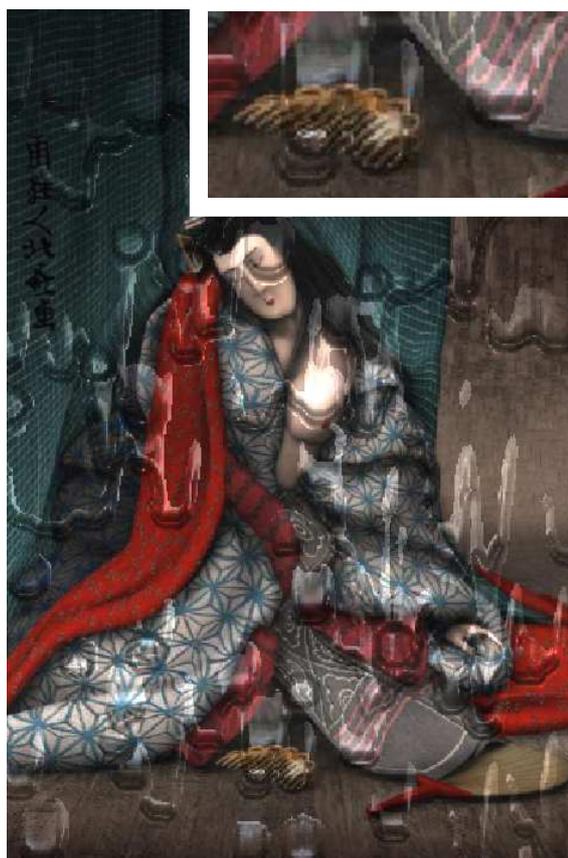


FIG. 2.6: Flot de gouttes non perturbé au dessus de la peinture Nishiki, 309 images par seconde. La réfraction de la texture décorative peut clairement être observée sur la zone agrandie en haut à droite.

Connaissant la couleur issue de la texture décorative ainsi que la quantité de liquide absorbée q_a , le texel de la texture décorative peut être indifféremment atténuée ou mélangée avec une texture dite cachée. Les couleurs des fragments sont respectivement calculés par $((1 - q_a) * texture_{pattern})$ et $((1 - q_a) * texture_{pattern} + q_a * texture_{hidden})$.

– Effet de transport d’encre

La texture d’encre est mélangée avec la texture décorative afin de donner la couleur du fragment $(\alpha_{pattern} * texture_{pattern} + texture_{ink})$. On remarquera qu’aucune pondération n’est appliquée aux composantes de la texture d’encre étant donné que les couleurs sont déjà pré-multipliées.

La figure 2.6 montre une surface opaque où la réfraction par le flot liquide est simulée à l’aide d’une texture décorative, et peut être clairement observée.

Un modèle simple d’illumination globale de la surface mouillée a été implémentée dans la présente méthode, les composantes diffuses et spéculaires pouvant être facilement dérivées à partir du champs de gradients de la texture liquide ainsi que des points d’intersection dénotant l’interface liquide-air.

2.4 Resultats

La méthode proposée a été implémentée sur un processeur Athlon64 3200 Mhz avec une carte graphique GeForce 7800. Les résultats communiqués dans cette partie ont été obtenus pour une fenêtre d’image de 1024*768. Du aux performances de l’algorithme en terme d’images par seconde, le champs de vélocités n’a jamais induit de violation de la condition de CFL, une seule passe ayant été nécessaire à l’advection sans le biais de passes intermédiaires (voir la définition de la troisième passe dans la partie 2.3.1).

	256 ²	512 ²	1024 ²
Standard	2.27ms	3.26ms	8.12ms
Absorption	2.32ms	3.34ms	8.36ms
Ink Transport	3.22ms	5.48ms	13.7ms

TAB. 2.1: Temps de calcul en millisecondes nécessaire à un pas de temps de la simulation hors rendu

Le tableau 2.1 dresse les temps de calcul requis afin qu’un pas de temps complet de la simulation s’achève (hors rendu) en fonction de différentes tailles de grilles. Pour chacun

des types de simulation (soit, standard, avec effet d'absorption ou avec effet de transport d'encre), une carte de normales a été utilisée afin de perturber le flot (nécessitant donc un accès texture supplémentaire par fragment). Pour l'effet d'absorption, une texture cachée a été utilisée qui nécessite un accès mémoire supplémentaire en comparaison avec une simple atténuation. On remarquera que la simulation proposée opère rapidement même pour de larges grilles.



FIG. 2.7: Ville vue de nuit à travers une vitre sur laquelle il pleut de plus en plus, 359 images par seconde

Les figures 2.7 montrent une simulation où une surface translucide est progressivement mouillée, l'animation tournant en moyenne à 359 images par seconde. La grille de simulation est de dimension $170 * 256$, une cubemap de $512 * 512$ de résolution 8 bits ayant été utilisée pour l'environnement de la ville.

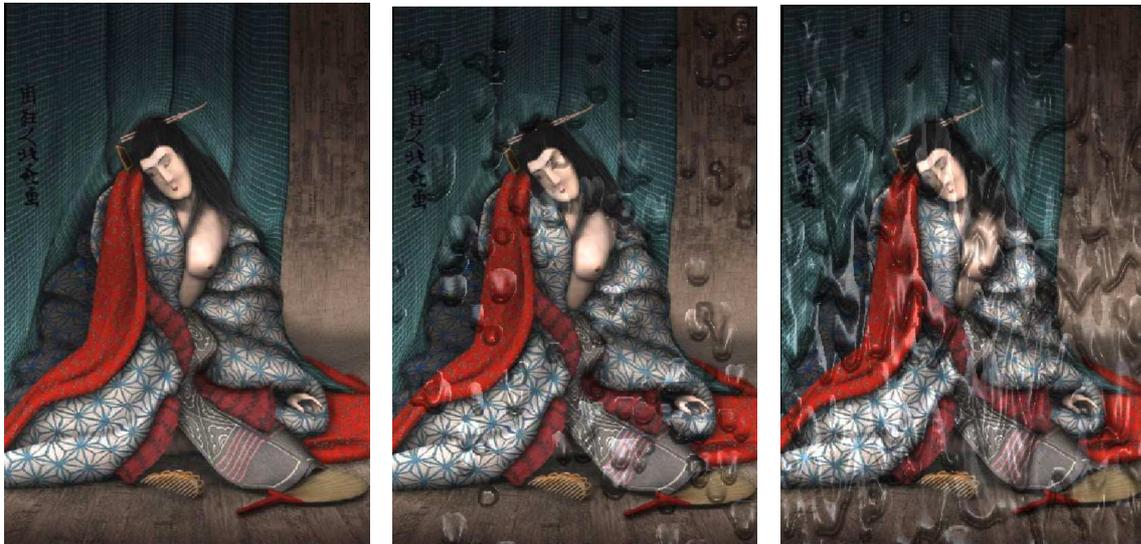


FIG. 2.8: Flot liquide sans texture de normales au dessus de la peinture Nishiki, 311 images par seconde

Les figures 2.8 montrent une simulation standard où une surface mouillée est rendue à une moyenne de 311 images par seconde. La texture décorative est de dimension $512 * 769$, la grille de simulation étant de $170 * 256$. La dynamique du flot liquide est uniquement

influencée par la force gravitationnelle.

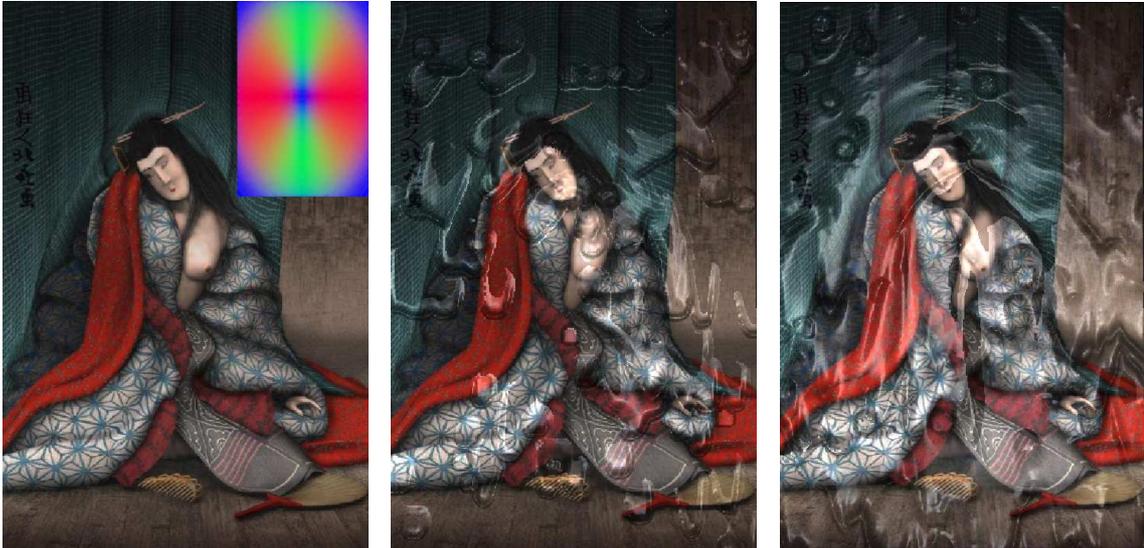


FIG. 2.9: Flot liquide influencé par une texture de normales au dessus de la peinture Nishiki, 302 images par seconde

Les figures 2.9 montre la même simulation que précédemment à la différence qu'une texture de normales a été utilisée afin de perturber le flot liquide. Le nombre d'images par seconde moyen est de 302, la texture de normales étant de dimensions $170 * 256$ et représentant une gaussienne projetée que l'on aura pris soin de déformer afin de garder le même rapport d'aspect que la texture décorative. En comparaison des figures 2.8, on peut clairement observer que le chemin emprunté par les gouttes diffère, ces dernières suivant de façon cohérente la géométrie dictée implicitement par la texture normale.

Les figures 2.10 montrent une peinture de Mona Lisa heureuse (texture décorative, $300 * 433$) disparaissant progressivement à 278 images par seconde. Les figures 2.11 montrent une peinture de Mona Lisa dans un style cubique nuancé de vert (texture décorative, $300 * 433$) laissant progressivement place à une Mona Lisa heureuse (texture cachée, $300 * 433$) à une moyenne de 272 images par seconde. Dans les deux cas, la taille de la grille de simulation est de $178 * 256$, et une texture normale de $44 * 64$ générée par du bruit a été utilisée afin d'obtenir un comportement non uniforme du flot liquide. Dans le premier cas, la valeur d'absorption a été utilisée pour atténuer la texture décorative tandis que dans le second cas, cette dernière a été utilisée pour le mélange entre les deux textures. Une fonction d'absorption simpliste $A = q_l$ a été utilisée pour ces deux résultats.

Finalement, les figures 2.12 montrent une texture décorative ($512 * 512$) délavée dans le temps par le flot liquide à une moyenne de 162 images par seconde. Les dimensions de la grille de simulation sont de $256 * 256$, une texture normale issue de bruit de dimensions



FIG. 2.10: Une peinture de Mona Lisa heureuse disparaissant progressivement dans les zones de forte humidité, 278 images par seconde

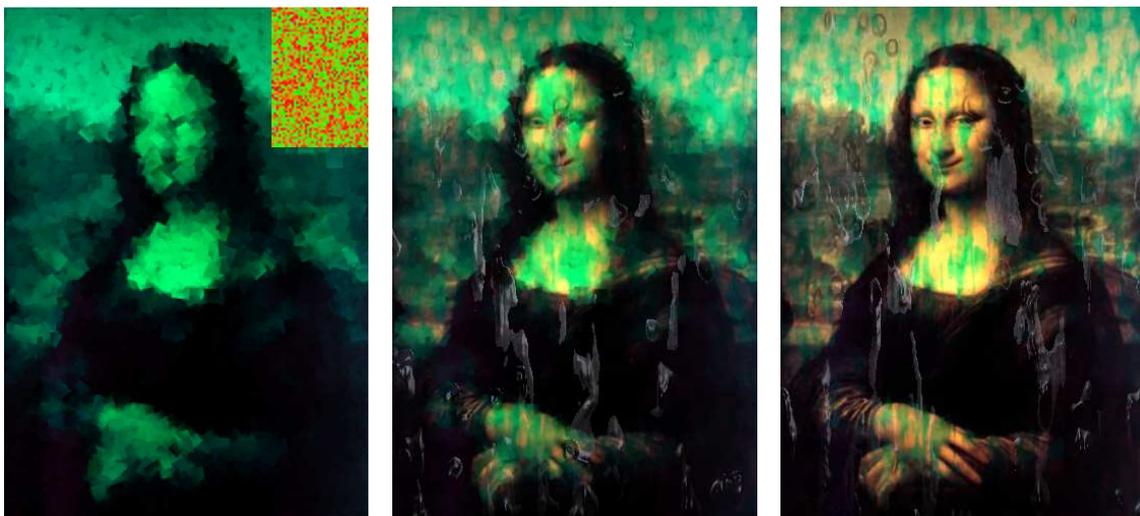


FIG. 2.11: Une peinture de Mona Lisa de style cubique teintée de vert laissant progressivement place à une Mona Lisa heureuse là où l'humidité de la surface est la plus forte, 272 images par seconde



FIG. 2.12: Encre d'une peinture illustrant une nuit étoilée emportées par des gouttes d'eau dévalant la surface, 162 images par seconde

64 * 64 ayant été utilisée. On observe sur les captures différentes couleurs émanant de la texture décorative qui sont transportées par le flot liquide, la texture décorative disparaissant ainsi progressivement là où des densités ont été enlevées.

Les textures normales des figures 2.9,2.10,2.11,2.12 sont visualisées en espace couleur (r, g, b) , la composante rouge prenant la valeur des abscisses, la verte des ordonnées et la bleue des profondeurs.

2.5 Conclusions

Une méthode empirique dédiée aux écoulements surfaciques en temps-réel a été présentée, une implémentation sur processeurs graphiques des algorithmes de simulation ainsi que de rendu ayant été démontrée. Le concept de la méthode est le calcul dynamique d'une carte de hauteur représentant un flot de gouttes liquides sur une surface plane, le matériau de cette dernière pouvant être de nature translucide comme opaque. Le processus de simulation prends en compte des forces extérieures, la viscosité du liquide et la présence d'obstacles. De précédentes méthodes basées sur un maillage du domaine imposent aux gouttes d'être gelées dans le temps avant d'être intégralement déplacées vers leurs nouvelles positions. En conséquence, une discrétisation très fine du domaine s'impose afin d'obtenir une fluidité convaincante pour l'animation. Le schéma d'advection qui a été présenté dans cette partie contourne cette limitation en gérant de courts déplacements de façon continue, des animations fluides étant obtenues indépendamment de la résolution de la grille de simulation. Le flot liquide est de forme lisse et continue en raison d'une modélisation par carte de hauteur ainsi que de la souplesse du schéma d'advection, la réfraction et la réflexion de la lumière étant dérivées selon le champs de gradients de la carte de hauteur. Cela diffère des précédentes approches où un flot est défini par une union de sphère, impliquant une interface liquide-air globalement discontinue ainsi qu'un champs de gradients pouvant varier abruptement. De plus, un effet d'absorption et un effet de transport d'encre ont été proposés, le surcoût calculatoire étant faible et les résultats convaincants. Ces travaux ont

donné lieu à une publication [HJGP08].

Cependant, le réalisme des animations pourraient être amélioré de plusieurs manières. Les effets de tensions de surface ayant été majoritairement négligés, la forme du flot liquide à la triple ligne de contact ne reflète pas le phénomène de l'angle de contact, l'attraction inter-gouttes ayant été elle aussi négligée. Trouver une formulation empirique efficace pour ces derniers permettrait d'accroître grandement le réalisme des animations actuellement obtenues, notamment lorsque le flot est visualisé de près. La méthode pourrait facilement être étendue afin de prendre en compte des surfaces courbes en usant d'une seconde carte de hauteur pour la géométrie de la surface. Cependant, de trop fortes courbures de la surface solide induiraient une distorsion visuelle du flot trop importante ainsi que des déplacements soumis à des accélérations incohérentes.

Une modélisation adaptée aux surfaces courbes tout en permettant une advection stable et rapide des gouttes liquides est en cours de recherche, une meilleure considération des effets de tensions de surface étant de rigueur.

Chapitre 3

Une approche basée physique pour l'animation de gouttes sur des surfaces arbitraires

Lors de l'état de l'art, il a été clairement explicité les raisons pour lesquelles des simulations physiques sont employées en synthèse d'images dans une optique d'animation de liquides. Pour résumer, le comportement d'un liquide est un phénomène naturel extrêmement complexe dont la formalisation algorithmique est -pour le moins que l'on puisse dire- non triviale. A ce jour, on ne peut que constater à travers les résultats de la communauté que seul l'emploi d'équations physiques issues de l'hydrodynamique permet de reproduire sur machine un tel phénomène avec un degré de réalisme saisissant, et donc suffisant (ceci est notamment vrai dans le domaine de la synthèse d'images où une justesse des résultats au sens de la physique n'est pas requise). Il est d'autant plus appréciable que seules de mineures modifications relatives à des paramètres globaux (comme la viscosité ou la densité) ou à des conditions aux bords (comme la présence d'un solide ou une transition de phase) soient nécessaires à l'animation de fluides de diverses natures dans des environnements à géométrie quelconque, certaines approches plus poussées allant même jusqu'à agrémenter les équations de bases de termes supplémentaires afin d'étendre l'éventail des phénomènes reproductibles (par exemple, l'ébullition, les bulles ou les gouttes). En revanche, les contre coûts de ces méthodes basées physiques sont principalement leurs complexités calculatoire et mémoire, les destinant généralement ainsi à des animations et à des rendus en temps différé. Non des moindres, on notera également que l'implémentation de tels simulateurs se révélera être une tâche ardue en raison de l'enchaînement intrinsèque de plusieurs méthodes numériques complexes possiblement non intuitives, une seule négligence pouvant réduire des heures de calculs à du simple temps perdu. En effet, les solveurs mis en jeu résultent d'une combinaison (souvent qualifiée de savante) entre diverses techniques telles que pour la modélisation du liquide, la discrétisation des équations ou l'intégration de ces dernières. Il existe une très vaste littérature (et donc de nombreux choix) pour chacune de ces techniques dont les avantages, inconvénients et complexité diffèrent, et il est diffi-

cile d'estimer avant terme l'impact d'un de ces choix sur la convergence globale du solveur.

Dans le présent chapitre, une méthode basée physique pour l'animation de gouttes est présentée. Les équations de Navier et Stokes ont été choisies pour la formalisation générale de la dynamique d'un liquide, cela en raison de leur longue popularité (on bénéficie ainsi de la part de la communauté physique de leur compréhension ainsi que de nombreuses méthodes de résolution numériques) et de leur adaptabilité (possibilité de prendre en compte les effets de tension de surface). On décrira dans une première partie les équations de Navier et Stokes d'un point de vue purement théorique suivies des choix effectués quant à leur résolution numérique, choix que l'on justifiera. On s'intéressera ensuite à l'inclusion des effets de tensions de surface à l'interface liquide-gaz, puis à ceux agissant à la triple ligne de contact (intersection liquide-gaz-solide). Ces derniers sont à l'origine du comportement et de la forme des gouttes, et donc au coeur de notre problématique. Une unique contribution abordant précisément la présente thématique en synthèse d'images à notre connaissance [WMT05], cette dernière sera décrite et analysée suffisamment en détail afin de la comprendre, d'en ressortir ces principales limitations et finalement en extraire des pistes de recherche. On présentera également quelques modèles de triple ligne de contact élaborés dans le domaine de la physique numérique, modèles que l'on comparera à celui proposé par Wang *et al.* [WMT05]. A partir des précédentes limitations et de la compréhension des modèles physiques découleront quelques nouvelles techniques dédiées à la dynamique de la triple ligne de contact. Ces techniques destinées au domaine de la synthèse d'images font l'objet du présent chapitre, et l'on présentera les diverses théories proposées accompagnées de résultats préliminaires.

Dans la grande majorité des cas, on adoptera les conventions typographiques suivantes. Les grandeurs vectorielles seront notées en gras \mathbf{u} tandis que les grandeurs scalaires seront laissées telles quelles x . On dénotera par \cdot le produit scalaire, \times le produit vectoriel, $| \cdot |$ la valeur absolue et $\| \cdot \|$ la norme d'un vecteur.

3.1 Equations de Navier et Stokes

3.1.1 Description

On se place dans le cadre adéquat d'un fluide incompressible (par exemple, l'air ou l'eau). Dans un tel cas, les équations de Navier et Stokes peuvent se dériver à partir des principes de conservation de la quantité de mouvement (équation 3.1) et de conservation de la masse (équation 3.2), la conservation de l'énergie étant implicite. Dans leurs formes discrètes, on a alors :

$$\frac{\partial \mathbf{u}}{\partial t} = -\mathbf{u} \cdot \nabla \mathbf{u} + \nabla \cdot (\nu \nabla \mathbf{u}) - \frac{1}{\rho} \nabla p + \frac{1}{\rho} \mathbf{F} \quad (3.1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (3.2)$$

où \mathbf{u} décrit le champs vectoriel des vitesses au sein du fluide, ν la viscosité cinématique du fluide, ρ la densité (soit, la masse volumique) du fluide, p le champs scalaire des pressions, \mathbf{F} des forces extérieures et t le temps.

On dénotera par ∇ le gradient $\nabla = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})$ et par ∇^2 le laplacien $\nabla^2 = \nabla \cdot \nabla = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$.

Les quatre termes de droite de l'équation 3.1 peuvent être décrit individuellement et se comprennent intuitivement :

1. $-\mathbf{u} \cdot \nabla \mathbf{u}$ représente l'advection du fluide par lui-même et décrit le transport des vitesses le long de lignes de courants. On notera que ce terme est à la base du caractère non linéaire des équations de Navier et Stokes.
2. $\nabla \cdot (\nu \nabla \mathbf{u})$ dénote le terme visqueux. Plus forte sera la viscosité du fluide, plus atténuée sera l'influence des vitesses environnantes et inversement. Phénoménologiquement parlant, un liquide visqueux se mouvoira plus lentement et aura tendance à demeurer compact.
3. $-\frac{1}{\rho} \nabla p$ est la contribution du champs de pressions sur les vitesses. Conceptuellement, un fluide aura tendance à se déplacer vers les zones de plus faibles pressions.
4. $\frac{1}{\rho} \mathbf{F}$ est l'action de forces extérieures, et se résume généralement à la force gravitationnelle.

Les équations de Navier et Stokes sont des équations différentielles partielles non linéaires pour lesquelles il n'existe toujours pas à ce jour de solutions analytiques, une résolution numérique étant en revanche possible. Typiquement, une simulation numérique part de conditions initiales sur les variables pour successivement décrire à chaque pas de temps leurs évolutions, des conditions aux bords étant imposées et la contrainte de conservation de masse (équation 3.2) se devant d'être respectée à chaque itération. Dans le cadre des présents travaux, on considère une simulation diphasé avec contacts solides. Soit les domaines suivants représentant respectivement un liquide Ω_L , un gaz Ω_G et un solide Ω_S . Il existe alors deux interfaces, l'interface liquide-gaz $\Gamma_{LG} = \Omega_L \cap \Omega_G$ et l'interface solide-fluides $\Gamma_S = (\Omega_L \cap \Omega_S) \cup (\Omega_G \cap \Omega_S)$. La triple ligne de contact sera décrite par l'interface particulière $\Gamma_C = \Omega_L \cap \Omega_G \cap \Omega_S$. En trois dimensions, Γ_{LG} et Γ_S sont des surfaces bidimensionnelles orientées plongées dans \mathbf{R}^3 tandis que Γ_C sera une courbe unidimensionnelle orientée dans \mathbf{R}^3 . Soit Ω le domaine de résolution numérique tel que $\Omega = \Omega \cap (\Omega_L \cup \Omega_G \cup \Omega_S)$. La figure 3.1 illustre cette représentation en deux dimensions.

Avant de décrire la méthode de résolution à proprement parler des équations de Navier et Stokes dans le domaine fictif Ω , il est primordial de décider d'une représentation

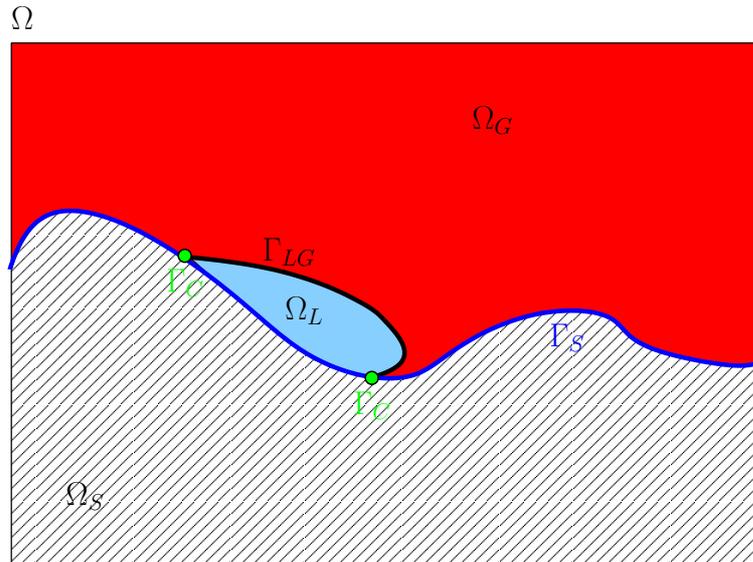


FIG. 3.1: Domaine de résolution Ω pour les équations de Navier et Stokes où Ω_L dénote le domaine liquide (zone bleue), Ω_G dénote le domaine gazeux (zone rouge) et Ω_S dénote le domaine solide (zone hachurée). Les interfaces engendrées sont au nombre de trois, respectivement l'interface liquide-gaz Γ_{LG} (courbe noire), l'interface solide-fluides Γ_S (courbe bleue) et l'interface particulière de la triple ligne de contact Γ_C (points verts).

discrète des différents domaines afin de suivre l'évolution des interfaces dans le temps. C'est le premier point que l'on abordera dans ce qui suit. Puis, on décrira les schémas numériques utilisés pour la résolution de chacun des quatre termes de l'équation 3.1 ainsi que l'imposition des conditions aux bords pour les interfaces respectives solide-fluides Γ_S et liquide-gaz Γ_{LG} . Sera ensuite abordé le point crucial des conditions aux bords pour la triple ligne de contact, soit l'interface liquide-gaz-solide Γ_C . On étudiera les précédents travaux de Wang *et al.* et certains autres modèles issus de la physique numérique afin de dégager de nouveaux moyens d'estimation des conditions aux bords à la triple ligne de contact. L'imposition de ces dernières une fois intégrées au solveur permettra alors de simuler des écoulements de gouttes, la précision et la justesse requises se devant d'être suffisante pour la synthèse d'image (on peut se permettre d'être légèrement approximatif d'un point de vue de la physique tant que l'effet visuel désiré est atteint).

3.1.2 Discrétisation des équations et des interfaces

Discrétisation des équations différentielles

On souhaite discrétiser deux fluides immiscibles dans un environnement pouvant être partiellement solide dans une optique de simulation numérique. Il existe plusieurs méthodes de discrétisation disponibles dans la littérature, discrétisation qui influencera directement la résolution des équations de Navier et Stokes en terme de méthodologie, complexité et convergence. Avant d'énumérer les principales méthodes de discrétisations existantes,

rappelons certains objectifs quant à la méthode développée. Bien que basée physique, il est important de bénéficier d'une méthode aux temps de calculs non prohibitifs ainsi que d'une implémentation la plus simple possible dans la mesure où les résultats satisfont le critère de réalisme en vue de la synthèse d'images. Relativement à la théorie des effets de tensions de surface, il sera important de bénéficier de bonnes approximations des normales et des courbures aux interfaces, ces dernières faisant parties intégrantes du théorème de Laplace et de la relation de Young. Décrivons succinctement les différentes classes de discrétisation eulériennes communément rencontrées en physique numérique :

1. Méthode des différences finies

Cette méthode effectue une approximation des dérivées intervenant dans le système d'équations différentielles à l'aide d'opérateurs différentiels discrets, ces derniers étant issus de développements en séries de Taylor tronqués et donc d'un certain ordre. Les variables peuvent ainsi être mises à jour par application de ces opérateurs sur les champs de variables passées (schéma explicite) ou par supposition du champs actuel (schéma implicite). La méthode des différences finies est une méthode populaire car simple d'implémentation et de complexités calculatoire et mémoire raisonnables comparée à ses consœurs, la qualité de la solution étant en revanche fortement dépendante du degré de raffinement du maillage (usuellement une grille) ainsi que de l'ordre des opérateurs employés. On notera que la représentation discrète des interfaces n'est pas imposée dans une méthode aux différences finies.

2. Méthode des volumes finis

Contrairement à la méthode des différences finies qui effectue une approximation des équations différentielles, la méthode des volumes finis est basée sur une approximation de l'intégration de ces dernières. Le domaine est discrétisé en une grille représentant des sous-volumes auxquels sont appliquées individuellement les lois de conservation. En utilisant le théorème de la divergence qui lie une intégration volumique à une intégration surfacique et usant de la connexité des sous-volumes, les variables peuvent être intégrées numériquement dans le temps sans dissipation aucune quelque soit le raffinement du maillage. Cependant, de par sa nature, l'interface ne peut être obtenue trivialement et des schémas numériques de reconstruction interfaciale sont nécessaires à une visualisation. On citera par exemple le schéma SLIC [NW76] (de l'anglais, « Simple Line Interface Calculation »), le schéma donneur-accepteur de Hirt et Nichols [HN81] ainsi que le schéma PLIC [Rud97] (de l'anglais, « Piecewise Linear Interface Calculation »), ce dernier étant le plus populaire pour la qualité de la reconstruction qu'il procure en dépit d'une complexité calculatoire non négligeable. De plus, il est courant d'observer des variations brutales cycliques dans la localisation de l'interface lors d'une fusion ou d'une séparation d'interfaces [HN81] (phénomène communément dénommé en anglais « flotsam and jetsam »), pouvant ainsi gravement nuire au réalisme d'une animation. Finalement, et non des moindres, on notera que l'estimation standard des normales et des courbures dans une méthode aux volumes finis est généralement de piètre qualité, des schémas d'approximation

d'ordre supérieur mais plus lourds pouvant néanmoins remédier en partie à cette lacune [RR10, JEPP04, CFK05, FCD⁺06].

3. Méthode des éléments finis

La méthode des éléments finis permet de résoudre une équation différentielle partielle dans sa forme variationnelle (soit, dans sa forme intégrale). La discrétisation emploie un pavage du domaine en éléments finis auquel sont associées des fonctions de bases permettant de recouvrir la continuité du champs (souvent, il s'agira de fonctions d'interpolation telles que les polynômes de Lagrange). En général, les éléments sont discrétisés sous la forme de triangles en 2D (respectivement des tétraèdres en 3D) afin d'aligner au mieux les bords du domaine réel sur ceux du domaine fictif, un raffinement local du maillage proche de l'interface étant de rigueur pour une meilleure précision numérique. Une conséquence directe est la nécessité de remailler le domaine à chaque déplacement des interfaces afin de coïncider à nouveau sur les bords, processus très coûteux notamment en 3D. Bien que la justesse des résultats soit souvent supérieure dans une méthode aux éléments finis en comparaison avec les méthodes des différences finies ou des volumes finis, elle est indubitablement plus difficile d'implémentation et engendre des complexités calculatoire et mémoire la rendant peu appropriée au domaine de la synthèse d'images.

En synthèse des points précédents, la méthode des différences finies semble être un choix adéquat pour la discrétisation des équations de Navier et Stokes, cette dernière offrant à première vue une implémentation tant efficace que relativement simple. Cela est d'autant plus vrai si l'on considère l'existence de schémas numériques traitant les discontinuités interfaciales (par exemple, le saut de pression lors d'une transition de phase) ainsi que l'existence de méthodes de discrétisation régulière du domaine permettant la localisation précise de l'interface même lorsque non alignée sur les bords de la grille. On s'intéresse tout d'abord au second point, les discontinuités interfaciales étant traitées dans la partie dédiée à la résolution numérique.

Discrétisation des interfaces

Les choix possibles quant à la discrétisation des interfaces sont principalement les suivants (voir la figure 3.2) :

- Des particules inertielles inter-connectées transportées par le flot. Cette discrétisation lagrangienne souffre de plusieurs lacunes, notamment de la difficulté à gérer les processus de fusion ou de séparation des interfaces, mais aussi de l'obligation de distribuer réciproquement les forces agissant sur les particules aux voxels environnants après un *lissage* empirique des variables (par exemple, à l'aide d'une fonction de Heaviside). Il s'agit d'une approche dite de *suivi d'interfaces* contrairement aux deux approches suivantes qui sont des approches dites de *capture d'interfaces*.
- Une discrétisation eulérienne en sous-volumes. On hérite alors des processus coûteux

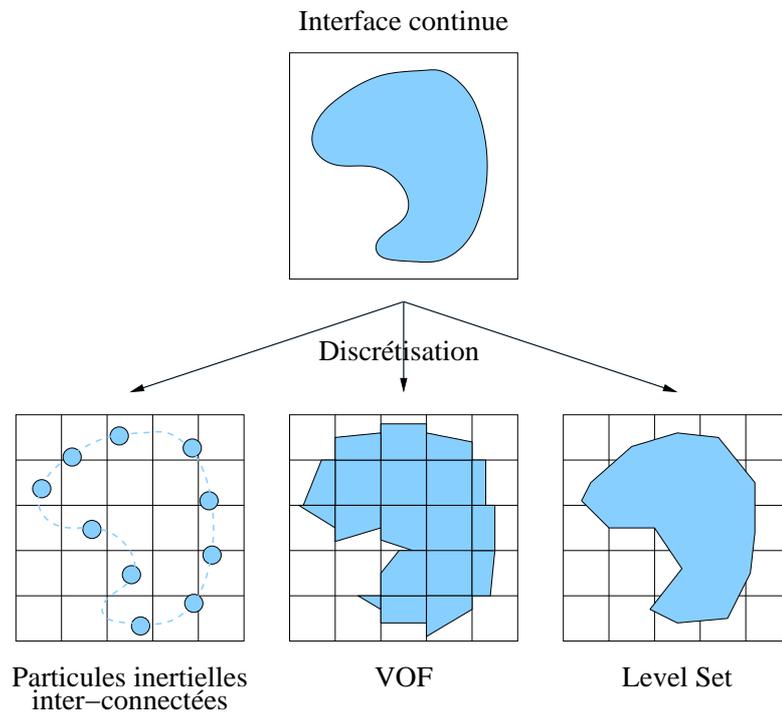


FIG. 3.2: Trois modèles de discrétisation d'une interface continue (centrée en haut). A gauche, utilisation de particules inertielles inter-connectées, au milieu, discrétisation en volumes finis et à droite, discrétisation en un level set. Le level set permet de conserver la continuité de l'interface (contrairement à une méthodologie VOF) et est apte à gérer aisément fusions et séparations interfaciales (contrairement aux particules inertielles inter-connectées).

pour la reconstruction interfaciale sans pour autant bénéficier de la conservation volumique étant donné la méthodologie adoptée des différences finies, rendant ce choix à priori superflu.

- Une discrétisation eulérienne en un champs de distances signées, communément dénommé level set [OS88, OF02]. Chaque voxel de la grille contient la distance de son centre à l’interface, le signe servant à départager l’intérieur de l’extérieur du domaine. Cette discrétisation permet de localiser précisément une interface en dessous de la résolution de la grille, et présente l’avantage majeur dans notre cas de pouvoir approcher avec une bonne précision les normales et les courbures. Cette discrétisation constitue notre choix et est décrite dans ce qui suit.

Un level set permet de représenter une seule interface, soit, uniquement deux domaines adjacents. Dans la méthode présentée, deux level sets seront utilisés conjointement afin de dénoter respectivement l’interface solide-fluides et l’interface liquide-gaz. La localisation de la triple ligne de contact découlera de la fusion des informations relatives à ces deux champs de distances. Dans le cas d’un domaine simple (par exemple, domaine liquide initialisé en tant qu’une conique pleine), le level set peut être déterminé de façon procédurale à partir du calcul de la distance des centres des voxels à une surface paramétrique. Dans le cas d’un domaine de forme arbitraire possiblement complexe (par exemple, un objet solide de la forme d’un lapin), on initialise le level set dans une bande étroite proche de l’interface à partir d’un maillage triangulaire du domaine en utilisant une méthode robuste pour l’estimation des distances signées [BA05]. Cette méthode nécessite des maillages représentant une deux variété sans bords plongée dans \mathbf{R}^3 . La majorité des maillages de modèles 3D disponibles en synthèse d’images répondent rarement à cette contrainte car issus de créations infographiques ou de numérisations d’objets réels. Afin de parer à cette éventualité, un algorithme de réparation [CL96] est employé pour combler des trous de surface parasites ainsi que supprimer auto-intersections et points singuliers. Le champs de distances signées est finalement complété à l’aide d’une technique d’extrapolation rapide [Set96] en tirant partie du fait que la propagation des caractéristiques s’effectue progressivement le long des normales.

Soit ϕ un level set tel que $\phi(\mathbf{x}, t)$ représente la distance du point \mathbf{x} à l’interface $\Gamma(t)$ à un instant t donné. Relativement au level set ϕ_{LG} de l’interface liquide-gaz (respectivement ϕ_S de l’interface solide-fluides), les distances nulles ou de signe négatif définissent le domaine du liquide (respectivement du solide) et par complémentarité, les distances strictement positives définissent le domaine du gaz (respectivement des fluides). L’interface Γ_{LG} (respectivement Γ_S) est définie par l’ensemble $\{\mathbf{x}, \phi(\mathbf{x}, t) = 0\}$ où on substitue ϕ par ϕ_{LG} (respectivement ϕ_S). On suppose une interpolation linéaire entre la valeur des centres des voxels, ce qui permet d’évaluer une interface par l’isosurface de valeur nulle à l’aide d’un algorithme de triangulation rapide et fiable [LC87, NH91]. Le vecteur normal est obtenu numériquement par application de l’opérateur du gradient au level set $\vec{n} = \nabla\phi$

tandis que la courbure le sera par application de l'opérateur du laplacien $\kappa = \nabla^2 \phi$.

L'atout principal d'un level set réside en sa capacité aisée à gérer des interfaces dynamiques, c'est à dire des interfaces sujettes à un champs de vélocités quelconques. Considérons la dérivée totale d'un level set ϕ au cours du temps t . Par définition :

$$\frac{D\phi(\mathbf{x}, t)}{Dt} = 0 \quad (3.3)$$

Après application du théorème de dérivation des fonctions composées, on obtient :

$$\frac{\partial \phi(\mathbf{x}, t)}{\partial t} + \frac{\partial \phi(\mathbf{x}, t)}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial t} = 0 \quad (3.4)$$

Sachant que $\mathbf{u} = \frac{\partial \mathbf{x}}{\partial t}$ et que le gradient ∇ correspond à une dérivation spatiale, on obtient finalement une expression de la dérivée eulérienne :

$$\frac{\partial \phi}{\partial t} = -\mathbf{u} \cdot \nabla \phi \quad (3.5)$$

Cette équation permettra au sein d'une simulation de fluides basée grille une intégration temporelle des interfaces en prenant \mathbf{u} comme étant le champs de vélocités des fluides. Simple et efficace, ce schéma d'évolution des interfaces démontre la puissance des level set en terme de support des processus de fusion et de séparation interfaciales. Cependant, il existe deux failles importantes lors de l'advection d'un level set : la distortion du champs de distances signées et la conservation du volume délimité par les interfaces.

Bien que la localisation des interfaces (soit, les isosurfaces définies par $\phi(\mathbf{x}, t) = 0$) demeure valide d'un certain point de vue, le level set ne représentera plus un champs de distances signées après advection du à une accumulation des erreurs numériques. Comme mentionné par Gomes et Faugera [GF99], cette violation de la propriété du champs de distances signées va à l'encontre de la théorie des level set, mais est pourtant bien réelle. Dans le cas de la présente méthode, il est inconcevable de se contenter d'un level set approximatif étant donné que la qualité des estimations des normales et des courbures dépend fortement de la qualité du level set en terme d'approximation du champs de distances signées. Une *réinitialisation* du level set est alors nécessaire, processus pour lequel il existe deux philosophies distinctes.

Une formulation à partir des valeurs initiales [SSO94] décrit une équation d'Hamilton-Jacobi pour ϕ au cours d'un temps fictif τ . La finesse de discrétisation de τ en des pas de temps intermédiaires Δ_τ détermine la qualité de la reconstruction ainsi que la largeur de la bande autour de l'interface dans laquelle les distances sont affectées qui est de $2\tau\Delta_\tau$:

$$\frac{\partial \phi}{\partial \tau} = \text{sgn}(\phi_{\tau=0})(1 - \|\nabla \phi\|) \quad (3.6)$$

où sgn est une fonction du signe du level set :

$$\text{sgn}(\phi) = \begin{cases} 1 & \text{si } \phi > 0 \\ 0 & \text{si } \phi = 0 \\ -1 & \text{si } \phi < 0 \end{cases} \quad (3.7)$$

possiblement atténuée afin de réduire les erreurs numériques [SSO94] :

$$\text{sgn}(\phi) = \frac{\phi}{\sqrt{\phi^2 + (\Delta x)^2}} \quad (3.8)$$

Cette approche nécessite de parcourir toute la grille $\frac{\tau}{\Delta \tau}$ fois et est de complexité calculatoire en $O(N^3)$ où N est le nombre total de voxels.

La seconde méthode de réinitialisation est celle qui a déjà été utilisée pour l'initialisation des level set [Set96], et propose une formulation du problème à partir de conditions aux bords de complexité calculatoire en $O(N \log N)$. Si l'advection de l'interface s'effectue uniformément vers l'intérieur ou vers l'extérieur de l'interface, alors il est possible de reformuler l'équation d'Hamilton-Jacobi en une équation eikonale. Le champs de distances signées est obtenue pour $F = 1$ grâce à un schéma incrémental démarré de l'interface où les distances sont estimées selon un polynôme du second degré imposant la norme du gradient du level set :

$$\|\nabla \phi\|^F = 1 \quad (3.9)$$

Ces deux approches ont été implémentées et testées car présentant des avantages différents, rappelons l'importance de cette réinitialisation relativement au solveur qui se basera en partie sur les estimations des normales et des courbures pour intégrer les effets de tensions de surfaces.

Le dernier point à considérer est celui de la conservation volumique, d'importance triviale dans le cadre d'une animation d'un liquide. Il existe de nombreuses alternatives afin de remédier à cette lacune des level set, et l'on énumère les plus courantes dans ce qui suit :

- Une première solution émane de l'observation qu'une approximation d'ordre faible des gradients du level set lors de l'advection est fortement responsable de la dissipation volumique, notamment lors de la présence de discontinuités. Des schémas de discrétisation d'ordre supérieur pour le gradient ont ainsi été proposés dans le cadre d'une équation de type Hamilton-Jacobi [SO88, SO89], plus tard améliorés en proposant une formulation adaptative [JS96]. Bien que les volumes soit mieux conservés en usant de tels schémas, une perte volumique est néanmoins constatable notamment lors de longues simulations. De plus, les stencils de discrétisation des gradients en question nécessitent un voisinage plus étendus (typiquement quatre voxels selon

- chaque dimension) ce qui rends leur implémentation plus lourde calculatoirement parlant mais aussi problématique lors de l'intersection avec les bords de la grille.
- En évaluant numériquement les volumes à l'itération précédente et en intégrant cette information sous la forme d'une contrainte lors du processus de réinitialisation du level set, les volumes délimités par les interfaces peuvent être conservés dans le temps. Ce type d'approches use d'heuristiques quant à l'évaluation individuelle des volumes pour relocaliser les interfaces, et présente l'inconvénient majeur d'une implémentation difficile [MEPS98, SF99, RS00].
 - Un autre type d'approche combine les avantages respectifs d'un level set avec ceux d'une méthode aux volumes finis, soit respectivement les informations topologiques et la conservation volumique. Le level set est utilisé lors de la résolution numérique du système d'équations différentielles pour ensuite être reconstruit. Cette reconstruction se base sur la répartition des sous-volumes qui auront été transportés indépendamment, et donc qui auront conservé le volume global délimités par les interfaces. Bien que relativement coûteuse en terme de calculs, cette méthode [SP00, Sus03] est efficace et a précédemment été utilisée pour la synthèse d'images [MMS04, MUM⁺06].
 - Mentionnée lors de l'état de l'art, la méthode des particules level set [EMF02] constitue une autre alternative à la résolution du problème de la conservation volumique. En peuplant de particules inertielles les deux cotés de l'interface dans une fine bande, les particules sont advectées selon un schéma explicite indépendamment du level set pour lequel une technique semi-lagrangienne (simple et stable) est utilisée. Les particules servent alors à reconstruire l'isosurface nulle du level set grâce à une formulation en surfaces implicites des particules. Cette méthode est simple d'implémentation mais permet surtout de se passer de schémas d'advection d'ordre élevé et coûteux grâce aux particules.

L'implémentation de la méthode des particules level set proposée dans [ELF05] est celle utilisée dans les présents travaux, et on détaille dans ce qui suit la résolution numérique des équations de Navier et Stokes.

3.1.3 Résolution numérique

La résolution numérique des équations de Navier et Stokes requiert d'intégrer les quatre termes de droite de l'équation de conservation de la quantité de mouvement 3.1, la solution étant soumise à des conditions aux bords ainsi qu'à la contrainte de conservation de masse 3.2.

On utilise une grille régulière 3D à arrangement délocalisée des variables où pressions et level sets sont stockés en les centres des voxels, les vitesses étant stockées sur leurs bords respectifs des voxels (voir la figure 3.3). Cette discrétisation permet de coupler fortement un champ de pressions à la divergence du champ de vitesses sans passer par une interpolation approximative supplémentaire, et présente de meilleures propriétés de

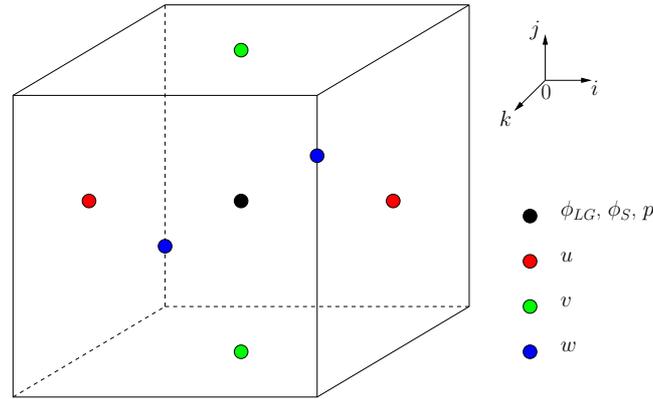


FIG. 3.3: Grille 3D à arrangement delocalisé. Les informations de distance des level sets ainsi que les pressions sont stockées en les centres des voxels, les composantes des vélocités $\mathbf{u} = (u, v, w)$ étant quant à elles discrétisées sur les bords de la grille relativement à leurs dimensions.

convergence comparée à un arrangement colocalisé des variables [HW65].

On utilise la méthodologie des pas fractionnaires pour la souplesse qu'elle autorise quant aux choix indépendants pour la résolution chaque terme ainsi que pour son implémentation intuitive. Chaque terme de l'équation 3.1 (hormis la pression qui sera traité à part) est donc intégré successivement à un champs de vélocités temporaire de divergence non nulle suivi d'une projection du champs de pression [Sta99] afin de répondre à la contrainte de conservation de la masse 3.2. On remarquera qu'ici, la pression joue le rôle d'un multiplicateur de Lagrange et n'a pas de sens d'un point de vue de la physique. Etant donné que notre intérêt se situe dans l'obtention d'un champs de vélocités de divergence nulle et que le champs de pressions résultant de la projection n'est pas destiné à une visualisation scientifique, cette approche rapide et puissante conceptuellement nous satisfait parfaitement (sinon, une approche par relaxation aux faibles propriétés de convergence et possiblement divergente aurait du être adoptée pour la résolution du terme de pression [FM96, CdVLHM97]).

La méthode proposée se veut diphasé, soit, on simule deux fluides incompressibles et immiscibles en interaction. Dans un tel cas, des considérations importantes sont à prendre en compte relativement à la formulation standard des équations de Navier et Stokes simple phase. Ces considérations ont attrait aux propriétés des fluides (la densité et la viscosité), à l'aspect physique d'une transition de phase (comportement des variables à l'interface) ainsi qu'à la formulation de la contrainte relative à la conservation de la masse. Ces différents aspects seront traités au fur et à mesure qu'on les rencontrera lors de la description de la résolution numérique de chacun des termes, description chronologique qui suit.

La méthode des caractéristiques est utilisée pour intégrer le terme d'advection $-\mathbf{u} \cdot \nabla \mathbf{u}$

comme introduit en synthèse d'images par Stam [Sta99] qui employa la terminologie d'advection semi-lagrangienne. Ce schéma d'advection de complexité calculatoire négligeable est inconditionnellement stable étant donné que le champs de vitesses est mis à jour à partir d'une interpolation (trilinéaire dans le cas de notre implémentation) du champs de vitesses à l'itération précédente :

$$\mathbf{u}^*(\mathbf{x}) = \mathbf{u}^t(\mathbf{x} - \mathbf{u}^t(\mathbf{x})\Delta t) \quad (3.10)$$

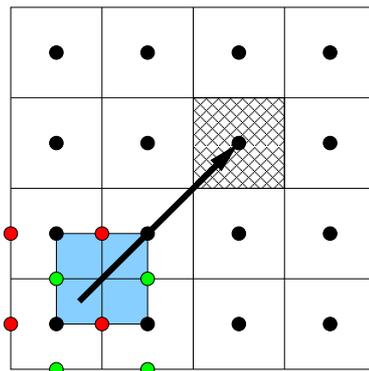


FIG. 3.4: Illustration d'une advection semi-lagrangienne. La case hachurée est mise à jour selon la méthode des caractéristiques. Un déplacement inverse (flèche noire) est calculé comme étant la vitesse de la case pondérée par le pas de temps. Les variables de la case sont alors intégrées dans le temps par interpolation trilinéaire des champs de variables à la position obtenue comme étant la translation du centre de la case par le vecteur de déplacement inverse.

L'advection du champs de vitesses est traitée prioritairement car un schéma semi-lagrangien convergera mieux dans le cas d'un champs de vitesses de divergence nulle [CMT04] (ce qui ne sera plus le cas jusqu'à la prochaine projection du champs de pression). Une illustration d'une advection semi-lagrangienne est montrée dans la figure 3.4.

La seconde étape est de loin la plus simple, et consiste à intégrer les forces extérieures $\frac{1}{\rho}\mathbf{F}$ selon un schéma explicite d'euler. En considérant la densité locale comme étant celle définie par la valeur du level set ϕ_{LG} en les centres des voxels, on a alors :

$$\mathbf{u}^{**}(\mathbf{x}) = \mathbf{u}^*(\mathbf{x}) + \frac{1}{\rho(\phi_{LG}(\mathbf{x}))}\mathbf{F}(\mathbf{x})\Delta t \quad (3.11)$$

Dans le cas de notre implémentation, seule la gravitation a été prise en compte étant donné qu'il n'est pas nécessaire de simuler des phénomènes externes (les objets solides sont immobiles, la formulation diphasé intégrant implicitement au solveur l'action d'un second champs de forces sans passer par l'ajout d'un champs de forces artificiel).

Le terme de viscosité $\nabla \cdot (\nu \nabla \mathbf{u})$ est ensuite traité implicitement comme introduit en synthèse d'images par Stam [Sta99] pour le cas d'un fluide simple phase. Dans ce cas, la

viscosité cinématique ν est constante dans tout le domaine et le terme de viscosité $\nabla \cdot (\nu \nabla \mathbf{u})$ se simplifie donc en l'expression $\nu \nabla^2 \mathbf{u}$ après factorisation par la constante visqueuse ν . On a à résoudre alors un système d'équation linéaire :

$$(I - \Delta t \nu \nabla^2) \mathbf{u}^{***} = \mathbf{u}^{**} \quad (3.12)$$

Dans une formulation diphasé des équations de Navier et Stokes, cette discrétisation est correcte dans le cas où le stencil du laplacien n'est pas entrecoupé par l'interface (soit, loin de l'interface). On utilise une discrétisation spatiale du laplacien d'ordre 1. Soit f une fonction de l'espace, le laplacien de f relativement à une dimension x est alors :

$$\begin{aligned} \nabla^2 f(x) &= \frac{\nabla f(x+\frac{1}{2}) - \nabla f(x-\frac{1}{2})}{\Delta x} \\ &= \frac{f(x+1) - 2f(x) + f(x-1)}{(\Delta x)^2} \end{aligned} \quad (3.13)$$

Dans le cas où une interface intersecte le stencil, ν ne peut plus être factorisé et on dérive une expression du laplacien prenant en compte la discontinuité de la viscosité :

$$\begin{aligned} \nabla \cdot (\nu \nabla f(x)) &= \frac{\nu(x+\frac{1}{2}) \nabla f(x+\frac{1}{2}) - \nu(x-\frac{1}{2}) \nabla f(x-\frac{1}{2})}{\Delta x} \\ &= \frac{\nu(x+\frac{1}{2}) f(x+1) - (\nu(x+\frac{1}{2}) + \nu(x-\frac{1}{2})) f(x) + \nu(x-\frac{1}{2}) f(x-1)}{(\Delta x)^2} \end{aligned} \quad (3.14)$$

$\nu(x)$ dépend du fluide occupant l'espace en x et est donc fonction du level set. En conséquence, sa valeur est bien définie en les centres des voxels mais pas sur les bords. Afin de prendre en compte l'intervalle spatial du stencil pour redéfinir la viscosité ν , il existe plusieurs méthodologies. Un lissage de la viscosité par le biais d'un opérateur de Heaviside H est souvent employé dont on explicite un exemple en une dimension ci-dessous :

$$\nu(x) = \nu(\phi(\lfloor x \rfloor)) + (\nu(\phi(\lfloor x \rfloor + 1)) - \nu(\phi(\lfloor x \rfloor))) H(\phi(x)) \quad (3.15)$$

où différents opérateurs de Heaviside [SSO94, MEPS98, SF99] basés sur une interpolation linéaire de ϕ sont possibles. On retire x de la formulation pour une meilleure lisibilité, et on cite comme exemples :

$$H(\phi) = \begin{cases} 0 & \text{si } \phi < 0 \\ \frac{1}{2} & \text{si } \phi = 0 \\ 1 & \text{si } \phi > 0 \end{cases} \quad (3.16)$$

$$H(\phi) = \begin{cases} 0 & \text{si } \phi < -\epsilon \\ \frac{1}{2} \left(1 + \frac{\phi}{\epsilon} + \frac{1}{\pi} \sin\left(\frac{\pi\phi}{\epsilon}\right) \right) & \text{si } |\phi| \leq \epsilon \\ 1 & \text{si } \phi > \epsilon \end{cases} \quad (3.17)$$

Dans l'équation 3.17, ϵ définit une bande autour de l'interface dans laquelle les valeurs sont lissées, et est une valeur typiquement de l'ordre de l'espacement inter-voxels (par exemple, $\epsilon = 1.5\Delta x$). Ce lissage empirique est nécessaire afin de simuler une continuité des constantes des fluides alors qu'en réalité, ces constantes sont discontinues à l'interface.

En lieu de ce lissage empirique, on part de propriétés physiques pour dériver une moyenne harmonique des constantes [LSSF06] (telle que la viscosité ν). Phénoménologiquement parlant, la viscosité est discontinue à l'interface, mais l'accélération visqueuse est continue. On considère la dimension x dans ce qui suit, l'extension à trois dimensions étant intuitive et directe. On dénote par l'index $-$ la valeur des variables à gauche de l'interface, et respectivement par l'index $+$ la valeur des variables à droite de l'interface. La continuité de l'accélération visqueuse peut alors être formulée :

$$\nu^- u_x^- = \nu^+ u_x^+ \quad (3.18)$$

Soit x_Γ le point d'intersection entre le stencil et l'interface (et donc, $\phi_{LG}(\lfloor x_\Gamma \rfloor + 1)\phi_{LG}(\lfloor x_\Gamma \rfloor) \leq 0$), et α la distance relative au bord gauche où l'intersection intervient telle que $\alpha = \frac{-\phi_{LG}(\lfloor x_\Gamma \rfloor)}{\phi_{LG}(\lfloor x_\Gamma \rfloor + 1) - \phi_{LG}(\lfloor x_\Gamma \rfloor)}$. On a alors :

$$\nu^- \frac{u(x_\Gamma) - u(\lfloor x_\Gamma \rfloor)}{\alpha \Delta x} = \nu^+ \frac{u(\lfloor x_\Gamma \rfloor + 1) - u(x_\Gamma)}{(1 - \alpha) \Delta x} \quad (3.19)$$

Par développement, on obtient une expression pour $u(x_\Gamma)$:

$$u(x_\Gamma) = \frac{\alpha \nu^+ u(\lfloor x_\Gamma \rfloor + 1) + (1 - \alpha) \nu^- u(\lfloor x_\Gamma \rfloor)}{\alpha \nu^+ + (1 - \alpha) \nu^-} \quad (3.20)$$

On réinjecte $u(x_\Gamma)$ tel que formulé dans l'équation 3.20 dans l'équation 3.19, pour finalement obtenir la moyenne harmonique $\nu(x + \frac{1}{2})$ pour l'intervalle $[\lfloor x_\Gamma \rfloor, \lfloor x_\Gamma \rfloor + 1]$ lorsque ce dernier est intersecté par l'interface Γ :

$$\nu(x + \frac{1}{2}) = \frac{\nu^+ \nu^-}{\alpha \nu^+ + (1 - \alpha) \nu^-} \quad (3.21)$$

On remarquera que lorsqu'il n'y a pas d'intersection entre l'interface et le stencil, cette formulation revient à une formulation simple phase si l'on prend soin de normaliser α entre 0 et 1. En substituant l'expression de la viscosité dans l'équation 3.14 par celle obtenue dans l'équation 3.21 et en réitérant le processus à gauche ainsi que selon les deux autres dimensions, on obtient une formulation du terme visqueux diphasé après avoir pris soin de remplacer f dans l'équation 3.14 par successivement chacune des trois composantes du champs de vitesses \mathbf{u} . Ce terme nouvellement formulé est exprimé dans sa forme implicite afin d'être intégré. Le système d'équations linéaires en découlant est symétrique défini positif, l'ordre faible employé pour l'opérateur du laplacien impliquant une matrice faiblement bandée et clairesemée. Ce type de système d'équations peut être rapidement inversé par une méthode itérative basée sur les espaces de Krylov, et on utilise dans le cadre de notre implémentation un algorithme du gradient conjugué avec un préconditionneur

obtenu par factorisation incomplète de Cholesky [Saa03]. Le pré-conditionnement n'est pas nécessaire en théorie, mais se révèle en pratique indispensable pour accélérer (parfois dramatiquement) la convergence de l'algorithme du gradient conjugué, notamment lors d'une forte différence de viscosités entre les deux fluides. Plus grande sera cette différence, moins bien conditionnée sera la matrice définissant le système d'équations linéaires et plus l'emploi d'un pré-conditionneur sera justifié.

L'étape finale du solveur est celle de la projection du champs de pression. Le théorème de Helmholtz-Hodge statue que tout champs vectoriel peut se décomposer en un champs vectoriel de divergence nulle et des gradients d'un champs scalaire. En faisant l'analogie entre le champs vectoriel quelconque et le champs des vitesses actuel \mathbf{u}^{***} , le champs scalaire et le champs de pressions p et finalement entre le champs vectoriel de divergence nulle avec le champs de vitesses solution des équations de Navier et Stokes \mathbf{u}^{t+1} , une projection peut être dérivée telle que :

$$\mathbf{u}^{t+1} = \mathbf{u}^{***} - \frac{\Delta t}{\rho} \nabla p \quad (3.22)$$

\mathbf{u}^{***} est connu, il s'agit du champs actuel des vitesses. Cependant, il demeure deux champs de variables \mathbf{u}^{t+1} et p à déterminer. En tirant partie du fait que \mathbf{u}^{t+1} sera de divergence nulle, on applique l'opérateur de divergence $\nabla \cdot$ des deux cotés de l'équation 3.22 afin d'éliminer \mathbf{u}^{t+1} , et ainsi obtenir une équation résolvable pour p :

$$\Delta t \nabla \cdot \left(\frac{1}{\rho} \nabla p \right) = \nabla \cdot \mathbf{u}^{***} \quad (3.23)$$

On se retrouve dans la même situation que pour le terme visqueux étant donné que l'on se doit de considérer la discontinuité interfaciale de la densité ρ , que l'on traite de la même façon en constatant que le déplacement de l'interface est identique quelque soit le fluide où l'on se place (sinon un vide serait créé et l'interface ne serait plus une interface). En d'autres termes, la vitesse et donc l'accélération sont continues à l'interface :

$$\frac{1}{\rho^-} \nabla p^- = \frac{1}{\rho^+} \nabla p^+ \quad (3.24)$$

Par souci de complétude, on notera que par analyse dimensionnelle, $[\frac{1}{\rho} \nabla p]$ correspond à $[\frac{1}{kg.m^{-3}} \frac{N.m^{-2}}{m}]$, que l'on développe $[\frac{1}{kg.m^{-3}} kg.m.s^{-2}.m^{-3}]$ pour réaliser qu'il s'agit d'une accélération $[m.s^{-2}]$. Une accélération étant une distance inversement proportionnelle au temps élevé au carré, le terme $\frac{1}{\rho} \nabla p$ dénote bien un déplacement interfacial.

A partir de l'équation 3.24 qui est similaire à l'équation pour la continuité de l'accélération visqueuse 3.19, on dérive une moyenne harmonique pour la densité ρ qui sera utilisée dans la discrétisation du terme de pression. On omet les détails ici, le processus de dérivation étant identique à celui utilisé pour le terme visqueux, pour ne donner uniquement l'expression de la moyenne harmonique de la densité :

$$\frac{1}{\rho(x + \frac{1}{2})} = \frac{\frac{1}{\rho^-} \frac{1}{\rho^+}}{\frac{\alpha}{\rho^+} + \frac{1-\alpha}{\rho^-}} \quad (3.25)$$

avec la discrétisation du terme de pression selon une seule dimension (l'extension à trois dimension est trivial) qui lui est associée :

$$\nabla \cdot \left(\frac{1}{\rho} \nabla p \right) = \frac{\frac{1}{\rho(x+\frac{1}{2})}p(x+1) - \left(\frac{1}{\rho(x+\frac{1}{2})} + \frac{1}{\rho(x-\frac{1}{2})} \right) p(x+1) + \frac{1}{\rho(x-\frac{1}{2})}p(x-1)}{(\Delta x)^2} \quad (3.26)$$

La divergence du champs de vélocités s'obtient simplement grâce à l'arrangement dé-localisé des variables, et est dans leur forme tridimensionnelle :

$$\begin{aligned} \nabla \cdot \mathbf{u}(\mathbf{x}) &= \frac{u(\mathbf{x}+(\frac{1}{2},0,0))-u(\mathbf{x}-(\frac{1}{2},0,0))}{\Delta x} \\ &+ \frac{v(\mathbf{x}+(0,\frac{1}{2},0))-v(\mathbf{x}-(0,\frac{1}{2},0))}{\Delta x} \\ &+ \frac{w(\mathbf{x}+(0,0,\frac{1}{2}))-w(\mathbf{x}-(0,0,\frac{1}{2}))}{\Delta x} \end{aligned} \quad (3.27)$$

où \mathbf{x} dénote les coordonnées des centres des voxels.

L'équation 3.23 est une équation de poisson où le système d'équation linéaires est défini à gauche par la discrétisation de la pression (équation 3.26) en concordance avec l'expression de la moyenne harmonique pour la densité (équation 3.25), le terme de droite étant défini par la divergence du champs de vélocités (équation 3.27). Ce système d'équations linéaires possède les mêmes propriétés algébriques que celui formé pour le terme visqueux, et on emploie ainsi la méthode itérative du gradient conjugué pré-conditionné afin de l'inverser rapidement. Le champs de pression résultant se substitue dans l'équation 3.22 afin d'obtenir le champs de vélocités \mathbf{u}^{t+1} dont la divergence est nulle. Ce champs de vélocités constitue *une* solution des équations de Navier et Stokes à l'itération $t + 1$.

Durant les étapes précédemment décrites, des conditions aux bords sont nécessaires afin de prendre en compte la présence d'un solide (immobile dans le cadre de notre implémentation). Ces conditions aux bords sont relatives au champs de vélocités \mathbf{u} et au champs de pression p et sont de types respectifs Dirichlet et Neumann [FM96] :

$$\left\{ \begin{array}{l} \mathbf{u}|_{\Gamma_S} = 0 \\ \frac{\partial p}{\partial \vec{n}}|_{\Gamma_S} = 0 \end{array} \right. \quad (3.28)$$

où \vec{n} dénote la normale à l'interface solide-fluides. On utilise pour la discrétisation des conditions aux bords relatives au champs de vélocités la méthode de Gibou *et al.* [GFCK02] alors que l'on utilisera la méthodologie présentée dans [FF01] pour la discrétisation des conditions aux bords relatives au champs de pression. Lors d'une intégration implicite

comme pour le terme de viscosité et le terme de de pression, ces deux formulations permettent de conserver un système d'équations linéaires symétrique défini positif tout en prenant en compte la présence de solides, et donc permet une résolution rapide et efficace des champs de variables concernés.

Les interfaces sont évoluées dans le temps avant que le solveur réitère le processus de résolution décrit ci-dessus. Comme précédemment mentionné, on utilise la méthode des particules level set [ELF05]. Dans cette méthode, le level set est évolué indépendamment des particules inertielles à l'aide d'un schéma d'advection semi-lagrangien. Ainsi, en considérant une interpolation trilinéaire du level set et en usant de l'équation 3.5 pour l'advection du level set, on a :

$$\phi(\mathbf{x}, t + 1) = \phi(\mathbf{x} - \mathbf{u}(\mathbf{x})\Delta t, t) \quad (3.29)$$

Après l'advection du level set, les particules sont évoluées à leur tour selon une intégration d'euler afin d'être utilisées comme décrit dans [ELF05] pour reconstruire l'interface. Une réinitialisation du level est ensuite nécessaire afin de réparer le champs de distance. Cette réinitialisation est basée sur l'équation différentielle 3.6 en concordance avec l'estimation du signe explicitée dans l'équation 3.8. Dans le cadre de notre implémentation, un temps fictif de $\tau = 10$ a au plus été utilisé en concordance avec un pas de temps de $\Delta\tau = 0.5\Delta x$, ce qui correspond à la réparation locale d'une bande épaisse de 5 voxels autour de l'interface. Cette épaisseur est suffisante étant donné que normales et courbures seront estimées (pour les effets de tensions de surface) uniquement à proximité de l'interface. On note que les particules sont sollicitées une deuxième fois après la réinitialisation afin de reconstruire l'interface, le processus de réinitialisation pouvant amener les interfaces à se déplacer faiblement.

3.1.4 Tensions interfaciales

La résolution des équations de Navier et Stokes telle que présentée précédemment permet de simuler deux fluides immiscibles incompressibles en interaction (c'est à dire qu'il existe une interface délimitant les deux domaines relatifs aux fluides). Comme on a pu le voir lors de l'état de l'art, il existe dans la nature des forces agissant dans le proche voisinage d'une telle interface. Ces forces sont communément dénommées tensions de surface. A une faible échelle, ce sont ces forces qui dominent et qui sont donc responsables de la forme des bulles et des gouttes ainsi que leurs dynamiques. Inversement, lorsque de grands volumes sont considérés, la force gravitationnelle domine et ces forces peuvent être négligées tant leur impact est insignifiant.

Dans le cadre de la présente méthode, on se place à une échelle suffisamment petite pour que les tensions de surface jouent un rôle préminent dans la forme et la dynamique

d'un liquide en interaction avec un gaz. On se doit ainsi de les intégrer au solveur numérique sous une forme discrétisée en rapport direct avec l'aspect théorique des tensions de surface. Cette partie traite de cet aspect là du solveur. On présente les deux formalisations dédiées aux effets de tensions de surface les plus populaires en physique numérique pour finalement décrire en détail celle que l'on aura choisie.

Conceptuellement, les tensions de surfaces existantes dans la proximité de l'interface sont des forces agissant le long de la normale à l'interface, leur sens dépendant du signe de la courbure tandis que leur intensité dépendra du degré de courbure pondérée par une constante. Cette constante γ est communément dénommée coefficient de tensions de surface, et se mesure comme étant le rapport d'une force sur une distance [$N.m^{-1}$]. Si l'on dénote le vecteur normal à l'interface \vec{n} , la courbure par κ , et par dS une surface infinitésimale, la tension de surface F_{TS} à l'interface peut alors s'exprimer mathématiquement de la sorte :

$$\mathbf{F}_{TS} = \gamma \kappa \vec{n} dS \quad (3.30)$$

et l'action globale des tensions de surface sur l'interface peut s'exprimer par le biais d'une intégration surfacique \oint :

$$\oint_S \mathbf{F}_{TS}(\mathbf{x}) d\mathbf{x} = \gamma \oint_S \kappa(\mathbf{x}) \vec{n}(\mathbf{x}) d\mathbf{x} \quad (3.31)$$

Le premier modèle de tensions de surface en physique numérique est du à Brackbill *et al.* [BKZ92] et est communément dénommé en anglais « Continuum Surface Force » (CSF). Une discrétisation de l'équation 3.31 est proposée en vue d'une incorporation en tant que force extérieure dans l'équation 3.1 de conservation de la quantité de mouvement des équations de Navier et Stokes. Sa formulation est la suivante :

$$\mathbf{F}_{CSF}(\mathbf{x}) = \gamma \kappa(\mathbf{x}) \vec{n}(\mathbf{x}) \delta(\mathbf{x}) \quad (3.32)$$

où δ est une dirac permettant d'estimer une portion de surface selon la localisation de \mathbf{x} relativement à l'interface. Cette évaluation surfacique correspond le plus souvent à un lissage empirique des valeurs similairement à une fonction de Heaviside (voir l'équation 3.17), et dépend donc directement du schéma de discrétisation employé pour le domaine de résolution (par exemple, un level set où des volumes finis). Le modèle CSF a été utilisé dans de nombreux travaux en physique numérique, et l'on citera à titre d'exemples de façon non exhaustive [CHMO96, Zuo96, SU98, MEPS98, BMC99, SP00, Spe05, vdPSVW08]. En synthèse d'images, le modèle CSF a été utilisé dans [CM04, SSK05, ZYP06] et on notera que bien qu'empirique, Hong et Kim [HK03] adoptent une approche conceptuellement très proche en traitant les effets de tensions de surface en tant que forces extérieures (leurs évaluations étant différentes).

L'approche CSF est une approche de type *diffuse*, c'est à dire qu'elle distribue volumiquement une force dans les alentours d'une interface alors que ces forces n'ont lieu qu'à l'interface même. Comme le montrent les résultats [BMC99], une oscillation de l'interface est couramment observée due au schéma diffus trop approximatif employé, l'équilibre des forces à l'interface au sens de la physique étant difficilement atteignable numériquement parlant. Ces oscillations portent atteintes à la convergence du solveur, mais sont surtout observables dans le cadre d'une visualisation, rendant ainsi cette approche peu appropriée pour la synthèse d'images.

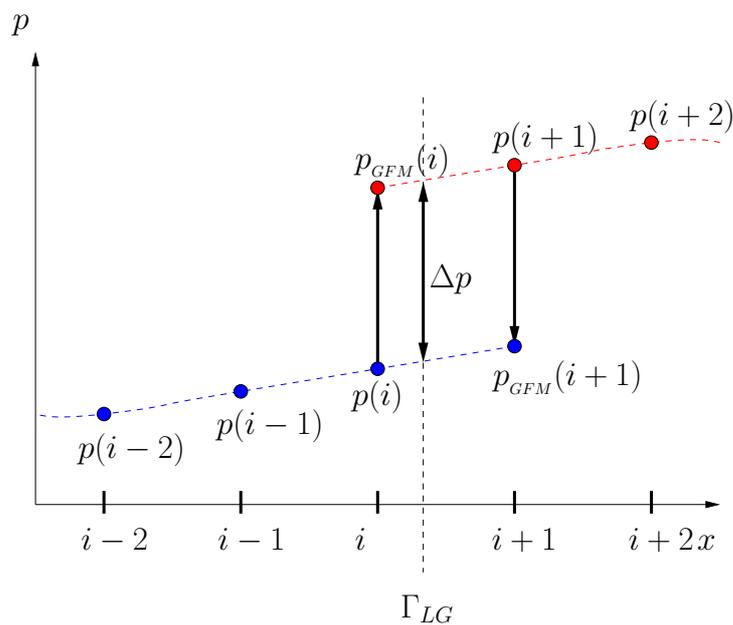


FIG. 3.5: Illustration du concept de la méthode des fluides fantômes (GFM). Lors d'une transition de phase à l'interface Γ_{LG} , une discontinuité de la pression existe comme formalisée dans le théorème de Laplace. A partir de ce théorème, des pressions fantômes ($p_{GFM}(i)$ et $p_{GFM}(i+1)$) de part et d'autre de l'interface sont déterminées à partir des pressions réelles et du saut de pression théorique Δp . Cette extrapolation interfaciale permet de simuler la continuité de chacune des phases (et donc une dérivation des pressions à l'interface est possible) tout en prenant en compte dans la simulation les effets de tensions de surface.

Une seconde approche pour la discrétisation des effets de tensions de surface est la méthode dite des fluides fantômes [FAMO99] (de l'anglais « Ghost Fluid Method », GFM). Conceptuellement, cette approche extrapole à l'interface le domaine d'un fluide dans celui de son voisin en se basant sur des propriétés physiques. Dans le cas des effets de tensions de surface, le théorème de Laplace est mis à contribution afin de décrire à proximité de l'interface un champ de pression virtuel où le saut de pression théorique interfacial $\Delta p = \gamma\kappa$ est pris en compte (voir la figure 3.5). Considérons uniquement l'axe des abscisses,

et soit x_Γ le point d'intersection entre une interface et deux voxels connexes selon x . Si l'on extrapole le fluide de gauche dans celui de droite, on a alors :

$$\begin{aligned} p_{GFM}(\lfloor x_\Gamma \rfloor + 1) &= p(\lfloor x_\Gamma \rfloor + 1) + \frac{\phi_{LG}(\lfloor x_\Gamma \rfloor)}{|\phi_{LG}(\lfloor x_\Gamma \rfloor)|} \gamma \kappa \\ \kappa &= \alpha \kappa(\lfloor x_\Gamma \rfloor) + (1 - \alpha) \kappa(\lfloor x_\Gamma \rfloor + 1) \\ \alpha &= \frac{-\phi_{LG}(\lfloor x_\Gamma \rfloor)}{\phi_{LG}(\lfloor x_\Gamma \rfloor + 1) - \phi_{LG}(\lfloor x_\Gamma \rfloor)} \end{aligned} \quad (3.33)$$

En substituant dans l'équation 3.26 les pressions des fluides voisins par celle d'un fluide fantôme si une transition de phase est de rigueur, les effets de tensions de surface sont incorporés dans la discrétisation du terme de pression des équations de Navier et Stokes. Les gradients du champs de pressions solution une fois soustrait au champs de vitesses temporaire comme dicté par l'équation 3.22 permet alors d'obtenir un champs de vitesses non seulement de divergence nulle, mais reflétant aussi l'action de forces minimisant l'énergie de surface.

La GFM est une approche agissant dimension par dimension et est de type *fine* du à la considération du saut de pression à l'interface même, les courbures à l'interface étant obtenues par une interpolation linéaire des courbures en les centres des voxels. Le signe du saut de pression dépend du fluide en considération, et étant donné qu'entre deux voxels connexes la même courbure interpolée est utilisée indépendamment du coté où l'on se place, le stencil de discrétisation demeure symétrique et défini positif (seul le terme source de droite est modifié). Cette dernière remarque est de prime importance car elle autorise l'utilisation de méthodes itératives rapides pour la résolution du système d'équations linéaires découlant d'une discrétisation GFM. L'autre atout majeur de la GFM est l'extrapolation à la volée du champs de pression à l'interface, la complexité mémoire du solveur restant ainsi inchangée comparée à une approche ne prenant pas en compte les tensions de surface. Récemment utilisée en synthèse d'images pour l'animation de liquides [HK05, LSSF06], la GFM a été choisie dans le cadre de notre implémentation pour sa stabilité tant numérique que visuelle (les effets d'oscillations d'interface à l'équilibre sont ainsi évités) mais aussi pour la meilleure justesse des résultats qu'elle engendre au sens de la physique et de la phénoménologie (la forme des bulles et des gouttes obtenues sont en meilleur accord avec la réalité comparée à une approche CSF).

On notera pour conclure que les premières approches en physique numérique proposaient d'estimer les effets de tensions de surface en imposant des conditions aux bords sur le champs de pression d'un des fluides. Par exemple, dans le cas de l'eau en interaction avec de l'air, une pression nulle pour l'air est supposée à laquelle est ajoutée un saut de pression lorsqu'à proximité de l'interface. Ce genre d'approches présente deux failles. Tout d'abord, le saut de pression est calculé au centre des voxels sans prendre en compte la localisation exacte de l'interface, mais il est surtout nécessaire de poser comme hypothèse la valeur de la pression à l'équilibre pour le fluide auquel on souhaite imposer des conditions

aux bords. Ainsi, une voxélisation des effets de tensions de surface peut être observée si la discrétisation du domaine n'est pas assez fine, l'hypothèse d'une pression à l'équilibre limitant grandement le réalisme et la diversité des simulations réalisables. Cependant, du à son implémentation facile et directe, ce type d'approche a précédemment été utilisé en synthèse d'images où des animations convaincantes ont pu être obtenues au prix d'une discrétisation très fine [FM97, LGF04, WMT05].

3.1.5 Tensions de surface à la triple ligne de contact

Si l'on analyse succinctement la théorie des effets de tensions de surface agissant à une interface, l'équilibre d'une interface entre deux fluides est atteint lorsque la courbure de cette dernière est nulle, soit lorsque l'interface représente un plan en trois dimensions. Cette formulation de l'équilibre interfacial est en accord avec les observations phénoménologiques pour une interface simple (par exemple, pour la surface de séparation existante entre de l'eau et de l'air lorsque éloignée de tout solide). Cependant, lorsqu'une faible quantité de liquide est déposée sur un solide, on se rends compte que l'équilibre à la triple ligne de contact est atteint pour un certain angle de contact et non plus pour un plan. L'équilibre hydrostatique à la triple ligne de contact est alors décrit par la relation de Young et non plus par le théorème de Laplace, le solveur précédemment présenté ne pouvant pas en conséquence reproduire le phénomène des gouttes car cherchant à établir un équilibre non valide à la triple ligne de contact d'un point de vue de la physique. Cette partie traite de la formalisation et de la discrétisation de la relation de Young en physique numérique, et on présentera dans la partie 3.2 quelques nouveaux modèles spécifiques à la synthèse d'images afin d'animer des flots de gouttes sur des surfaces solides. Rappelons que le phénomène de la triple ligne de contact -et notamment sa dynamique- demeure à ce jour mal compris de la communauté physique, la majorité des modèles ayant été proposés en physique numérique étant empiriques et se basant sur des observations phénoménologiques ainsi que sur des mesures expérimentales.

En dynamique des fluides, lorsqu'un fluide est en interaction avec un solide, des conditions aux bords sont imposées sur le champs de vitesses afin que la vitesse du fluide soit égale à celle du solide. Plus précisément, cette condition aux bords (dénommée en anglais « no-slip ») impose que le fluide ne peut se déplacer tangentiellement au solide. Bien que correcte dans le cadre de simulations où il est question de grandes étendues liquides, cette condition aux bords va à l'encontre du phénomène de la triple ligne de contact où un déplacement tangentiel du liquide peut être observé. Cette contradiction [CL71] a soulevée de nombreuses interrogations au sein de la communauté physique, certains allant même jusqu'à remettre en cause la validité des équations de Navier et Stokes. Afin de remédier à cette incohérence, une condition aux bords du solide plus laxiste dite de non-pénétration (en anglais « no-penetration » ou « free-slip ») a été introduite en pratique où la composante normale au solide de la vitesse du fluide est égale à la vitesse du solide (et donc où un déplacement tangentiel au solide est autorisé). A notre connaissance, cette condition au

bord est utilisée dans tous les travaux dédiés à la dynamique de la triple ligne de contact, d'où son importance. Dans les travaux que l'on présentera ci-dessous, une condition aux bords « no-slip » sera imposée sur les bords des voxels solides connexes à des voxels liquides excepté à la triple ligne de contact où une condition aux bords « free-slip » sera de rigueur afin de permettre le déplacement des gouttes sur le solide (typiquement, plusieurs voxels avoisinant la triple ligne de contact seront concernés).

On énumère dans ce qui suit les modèles de triple ligne de contact issus de la physique numérique :

1. Angle de contact constant [Zuo96, ZGZ97]

A partir de conditions initiales où la forme des gouttes est discrétisée comme étant déjà à l'équilibre (soit, l'angle de contact stable dicté par la relation de Young est respecté), une vitesse pour la triple ligne de contact est imposée afin que son déplacement soit égal au déplacement de la couche supérieure du liquide selon la normale au solide. Ainsi, l'équilibre est rétabli immédiatement à chaque pas de temps. Bien que ce modèle permet la simulation de l'écoulement d'une goutte sur une surface solide, sa simplicité résulte en son inaptitude à gérer des gouttes de formes arbitraires (soit, possiblement non à l'équilibre) ainsi qu'à reproduire des fusions ou des séparations de plusieurs gouttes.

2. Retour à l'équilibre [Zuo96, ZGZ97]

Connaissant la valeur de l'angle de contact stable θ_e et ayant déterminé la valeur de l'angle de contact pour le pas de temps courant θ_d , la vitesse de la triple ligne de contact \mathbf{u}_C est estimée comme étant la différence des deux angles précédents le tout pondéré par un coefficient λ dit de longueur de glissement. Une formulation linéaire $\mathbf{u}_C = \lambda(\theta_e - \theta_d)$ et une formulation quadratique $\mathbf{u}_C = \lambda(\theta_e^2 - \theta_d^2)$ ont ainsi été proposées, des résultats convaincants ayant été obtenus en pratique pour de faibles vitesses (et donc pour une faible différence entre les deux angles). Les principales limitations de cette méthode sont sa forte inexactitude lors de grandes différences entre l'angle de contact stable et l'angle de contact dynamique, la nécessité de déterminer empiriquement la constante λ ainsi que l'estimation à chaque pas de temps de l'angle de contact dynamique dont la justesse et la précision dépendront fortement de la finesse de la discrétisation ainsi que de la méthodologie numérique adoptée.

3. Condition aux bords de Navier [Zuo96, ZGZ97]

Proposée il y a près de deux siècles par Navier, une condition aux bords de Navier peut être imposée à proximité de la triple ligne de contact afin d'engendrer sa mobilité. La vitesse tangentielle $\mathbf{u}_C = \lambda \frac{\partial \mathbf{u}}{\partial \mathbf{n}}$ ainsi calculée dépendra de la variation des vitesses selon la normale au solide \vec{n} ainsi que d'un coefficient de glissement λ . Les résultats obtenus par cette méthode diffèrent de trop lorsque comparés à des mesures expérimentales, notamment en ce qui concerne la vitesse de déplacement de la triple ligne de contact.

4. Condition aux bords de Navier généralisée

Dans plusieurs travaux importants, Qian *et al.* [QWS03, QWS04, QWS06] proposent une condition aux bords de Navier généralisée pour la vitesse de la triple ligne de contact où le comportement de la triple ligne de contact à l'échelle microscopique est reproduit à travers un modèle hydrodynamique continu à l'échelle macroscopique (soit, un modèle utilisable dans une simulation classique des équations de Navier et Stokes). Bien que récents, ces travaux prometteurs semblent proposer pour la première fois une condition aux bords générale pour le problème d'interaction fluide-solide, les résultats relatifs à la simulation de gouttes étant en excellent accord avec les mesures expérimentales. Les limitations de ce modèle sont principalement sa difficulté d'implémentation et de compréhension (une formulation hydrodynamique diphasique de Cahn-Hilliard étant de rigueur) ainsi que la nécessité de déterminer des constantes relatives à chaque couple de fluides par des simulations de dynamique moléculaires (lentes et complexes) qui serviront de paramètres d'entrée lors de la simulation macroscopique. Cependant, cette piste sera explorée dans de futurs travaux pour la synthèse d'images du à son aspect fortement prometteur et généraliste.

5. Saut de pression à la triple ligne de contact

Afin de reproduire le phénomène de l'angle de contact, une adaptation du théorème de Laplace à la triple ligne de contact a été proposée, le théorème de Laplace étant initialement valide uniquement à l'interface. Le concept majeur derrière ces méthodes est la construction d'une interface virtuelle creusée dans la partie solide afin que les courbures calculées à la triple ligne de contact reflètent l'angle de contact stable désiré. Proposée pour la première fois en 2D par Sussman et Uto [SU98] et repris dans de nombreux travaux tels que [RRL01, Spe05, LKMU05, AB08], cette méthode a été adaptée en 3D pour la synthèse d'images par Wang *et al.* [WMT05]. On détaille cette dernière dans ce qui suit étant donnée son importance quant au contexte présent et l'on rappelle qu'il s'agit de la seule méthode dédiée à l'animation de gouttes sur un solide. Dans [WMT05], deux level sets sont utilisés pour la discrétisation de l'interface liquide-air et de l'interface solide-fluides. Afin de prendre en compte le phénomène de la triple ligne de contact, une surface virtuelle est construite dans un troisième level set. La surface virtuelle est dans un premier temps initialisée comme étant l'interface liquide-air. Pour chaque intersection de la triple ligne de contact avec deux voxels connexes dans la phase liquide, de nouvelles distances sont calculées pour les voxels directement connexes dans le solide à l'aide d'une estimation analytique. Cette estimation analytique calcule la distance signée entre le centre d'un voxel et un vecteur reflétant l'angle de contact stable relativement au point de contact. Lorsqu'une fine bande a été peuplée dans la partie solide pour tous les points de contact de la triple ligne, une extrapolation rapide [Set96] du champ de distances signées est effectuée dans une bande plus large du solide (typiquement trois voxels) afin que lors des calculs des courbures, toutes les distances utilisées dans les stencils soient valides. La surface virtuelle finale est alors obtenue. Lors de la prise en

compte des effets de tensions de surface où des conditions aux bords dans les voxels de l'air voisins à des voxels liquides seront imposées, cette surface virtuelle permettra de calculer des courbures valides même pour la triple ligne de contact. Dans un deuxième temps, Wang *et al.* proposent de prendre en compte la dynamique de la triple ligne de contact ainsi que le phénomène d'hystérésis de l'angle où vitesses et angles de contact sont intrinsèquement liées. Les paramètres d'angle de reculée et d'angle d'avancée sont alors nécessaires à l'instar de l'angle de contact stable. Afin de déterminer quel angle de contact imposer, deux surfaces virtuelles sont construites, l'une reflétant l'angle de reculée et la seconde l'angle d'avancée. Selon les signes respectifs des deux pressions obtenues à la triple ligne de contact, la pression adéquate pour chaque point de contact est choisie afin d'imposer l'angle de contact comme étant celui de reculée, d'avancée ou bien comme étant non contraint (phénomène de l'hystérésis).

Cette famille de méthodes présente de nombreux avantages. Tout d'abord, l'intégration des tensions de surface à l'interface comme à la triple ligne de contact est effectuée indifféremment à partir d'un saut de pression comme dicté par le théorème de Laplace. Aussi, en n'imposant pas la vitesse de la triple ligne de contact en tant que conditions aux bords comme lors de précédents modèles, aucune hypothèse n'est faite sur l'échelle du liquide et donc sur les forces qui dominent. Ainsi, dans le cas de fortes vitesses ou de grands volumes liquides, les effets de tensions de surface bien que pris en compte n'auront pas d'influences significatives sur le flot. En revanche, la complexité mémoire engendrée est accrue du à l'usage d'une troisième grille (angle de contact stable) voire d'une quatrième grille (dynamique de la triple ligne de contact et phénomène de l'hystérésis de l'angle). La complexité calculatoire est aussi grandement augmentée du à l'estimation analytique employée suivie d'une extrapolation (d'après les auteurs, une simulation typique pour une animation de moins d'une minute requiert de 5 à 8 jours de calculs sur un Pentium Xeon 2.8GHz). Ceci peut devenir une sévère limitation en 3D, notamment pour de fortes résolutions de grilles. Les résultats présentés dans [WMT05] nécessitent malheureusement une finesse de discrétisation élevée afin d'être convaincants, et nous pensons que cette limitation provient principalement de l'intégration des effets de tensions de surface en tant que conditions aux bords voxelisées. De plus, la technique de la surface virtuelle échoue pour des angles de contacts trop petits ou trop grands du à l'estimation analytique des distances dans le solide exacte en 2D mais approximative en 3D. Pour finir, on notera que les auteurs affirment que des conditions aux bords « no-slip » ont été utilisées tout le long du solide (et donc à la triple ligne de contact). Théoriquement, une pression infinie à la triple ligne de contact est alors requise pour que son déplacement soit possible [CL71] tandis qu'en physique numérique, les méthodes dédiées aux gouttes imposent une condition aux bords « free-slip » pour éviter cette singularité aux conséquences graves. Ce dernier point -important- est possiblement un oubli de la part des auteurs.

Synthèse :

On se focalise exclusivement sur les limitations de l'approche de Wang *et al.* (groupe n°5) car propre et unique au domaine de la synthèse d'images, les autres groupes de méthodes contribuant possiblement à les dépasser et aidant à une meilleure compréhension du problème. On énumère les limitations majeures de [WMT05] ci-dessous :

1. Du à une simulation simple phase, seule une interface fluide-air peut être considérée. Le phénomène de l'angle de contact est pourtant existant pour toute interface fluide-fluide tel que de l'eau avec de l'huile dans un récipient. Il serait intéressant dans le cadre de la synthèse d'images d'étendre l'éventail des combinaisons de fluides possibles afin de bénéficier d'une approche généraliste.
2. La prise en compte des effets de tensions de surface est effectuée en tant que conditions aux bords centrées en les voxels. Comme expliqué précédemment, cette méthodologie est trop approximative, la qualité des résultats dépendant fortement du degré de finesse de la grille.
3. Les angles de contacts modélisables ne peuvent être extrêmes en raison de l'estimation analytique trop approximative en 3D des distances dans le solide.
4. La création de la surface virtuelle requiert une complexité mémoire pouvant être très pénalisante en 3D, un angle de contact stable nécessitant une grille supplémentaire et la dynamique de la triple ligne de contact deux grilles supplémentaires. Une estimation à la volée des conditions aux bords à la triple ligne de contact permettrait de diviser par deux les ressources mémoires nécessaires à la simulation.
5. Dans le cas où le concept de surface virtuelle est maintenu, les ressources mémoires pourraient être néanmoins réduite d'un quart si une heuristique différente (soit, non basée sur les pressions) était utilisée quant à la détermination de l'angle de contact à imposer (soit, l'angle d'avancée ou l'angle de reculée).
6. Lever l'ambiguïté pour la communauté graphique sur la condition aux bords « no-slip » ou « free-slip » à adopter pour les vitesses à l'interface solide. En effet, d'après la communauté physique et ses résultats -mais intuitivement aussi- une condition « no-slip » interdit le déplacement de gouttes sur une surface solide (alors que cette même condition a été employée dans [WMT05]). Bien que n'étant pas une contribution en soit, il semble important de rectifier ce point crucial à la convergence de la simulation, point spécifique à l'animation de gouttes et non aux grandes étendues de fluides.

Le solveur décrit jusqu'à présent a été conçu dès le départ avec ces limitations en vue. Ainsi, la limitation n°1 est dépassée grâce à la formulation diphasique des équations de Navier et Stokes, permettant ainsi de prendre en compte toute interface fluide-fluide en contact avec un solide. De même pour la limitation n°2 où la GFM (méthode des fluides fantômes) intègre les effets de tensions de surface au solveur avec une meilleure justesse et précision comparé à des conditions aux bords voxelisées, permettant ainsi de réduire la

résolution des grilles. La limitation n°6 (qui n'est pas réellement une limitation en soit) est considérée afin d'imposer des conditions aux bords adéquates pour les vitesses dans le solide, des problèmes de convergence voire des explosions numériques ayant été rencontrés en pratique sinon. Les limitations n°3-4-5 font l'objet de la partie suivante et constituent en soit les réelles contributions des présents travaux de recherche.

3.2 Modèles proposés

3.2.1 Dynamique de la triple ligne de contact

On s'inspire des travaux [LKMU05, Spe05] afin de dicter le comportement de la triple ligne de contact et reproduire le phénomène d'hystérésis de l'angle de contact. Contrairement à Wang *et al.* [WMT05] où leur heuristique quant à l'angle d'avancée θ_a ou de reculée θ_r à imposer dépend des deux pressions respectivement obtenues, on se base sur la valeur de l'angle de contact dynamique θ_d actuel afin de choisir l'angle de contact adéquat. La méthode est simple et efficace, et se résume à deux étapes :

1. Calculer l'angle de contact dynamique θ_d actuel pour chaque point de contact de la triple ligne de contact. Grâce aux deux level sets ϕ_{LG} et ϕ_S , cette information est obtenue facilement et efficacement avec $\theta_d = \cos^{-1}(\nabla\phi_{LG} \cdot \nabla\phi_S)$ (ceci est vrai car ϕ_{LG} et ϕ_S sont des champs de distances signées normalisées à proximité de l'interface).
2. Si $\theta_r \leq \theta_d \leq \theta_a$, alors l'angle est en hystérésis et la triple ligne de contact n'est pas autorisée à se déplacer en imposant une condition aux bords « no-slip » sur les champs de vitesses $\mathbf{u}_C = 0$ (soit, dans une formulation par saut de pression, $\Delta p = 0$). Si au contraire $\theta_d > \theta_a$ (respectivement $\theta_d < \theta_r$), on impose l'angle de contact comme étant θ_a (respectivement θ_r) et une condition aux bords de « free-slip » sur les champs de vitesses est considérée (soit, dans une formulation par saut de pression, $\Delta p \neq 0$).

Cette évaluation de l'angle de contact ne se base que sur les champs de variables actuels des équations de Navier et Stokes contrairement à [WMT05] où deux champs de variables supplémentaires sont nécessaires. Ainsi, même dans le cadre d'une méthode basée sur la surface virtuelle, la dynamique de la triple ligne de contact ne nécessitera plus qu'un seul champ de variables supplémentaires, impliquant une baisse non négligeable d'un quart de la complexité mémoire. Les modèles présentés dans ce qui suit concernent uniquement l'évaluation des conditions aux bords à proximité de la triple ligne de contact, tous usant de la méthodologie que l'on vient de présenter pour la détermination de l'angle de contact à imposer.

3.2.2 Estimation des sauts de pression à la triple ligne de contact

Afin de prendre en compte les effets de tensions de surface à la triple ligne de contact en usant du théorème de Laplace, les courbures usuelles à l'interface doivent être réévaluées relativement à l'angle de contact que l'on souhaite imposer. La technique de la surface

virtuelle constitue un moyen d'arriver à cette fin. On propose ici différentes méthodes d'estimation des courbures où la surface virtuelle globale est remplacée par une approximation surfacique locale, ce qui permet une évaluation des courbures à la volée. Ainsi, les nouvelles méthodes présentées permettront de garder la même complexité mémoire du solveur de base au prix d'un faible surcoût calculatoire comme nous allons le voir.

Approximation de l'interface à la triple ligne de contact par deux sphères d'axes principaux orthogonaux

Soit une surface 2D orientée plongée dans \mathbf{R}^3 . En un point donné de cette surface, il existe un vecteur normal qui définit un plan tangent local à la surface. Une courbure pour cette surface est calculée comme étant l'inverse du rayon du cercle osculateur à la courbe 2D obtenue par section de la surface selon un plan perpendiculaire au plan tangent. Les deux courbures principales en un point de la surface sont définies comme étant respectivement le minimum et le maximum de toutes ses courbures (soit, pour toutes les sections perpendiculaires au plan tangent). La courbure moyenne κ (qui est celle utilisée en dynamique des fluides pour quantifier les effets de tensions de surface) est alors la moyenne de ces deux courbures. On s'inspire de [Mar98], et on propose d'estimer la courbure moyenne à la triple ligne de contact par une approximation numérique des courbures principales à partir des deux grilles interfaciales ϕ_{LG} et ϕ_S .

Considérons une goutte sur un solide et plaçons nous en un point de la triple ligne de contact. Un tel point de contact existe si deux voxels de coordonnées \mathbf{v}_1 et \mathbf{v}_2 connexes selon une dimension ont comme propriétés :

- les deux voxels ne sont pas dans le solide, soit $\phi_S(\mathbf{v}_1) > 0$ et $\phi_S(\mathbf{v}_2) > 0$.
- il y a une transition de phase entre les deux voxels, soit $\phi_{LG}(\mathbf{v}_1)\phi_{LG}(\mathbf{v}_2) \leq 0$.
- au moins un des 8 voxels 6-connexe selon les deux autres dimensions au bloc \mathbf{v}_1 - \mathbf{v}_2 est dans la phase solide.

Si ces conditions sont remplies, alors les coordonnées \mathbf{p}_C du point de contact sont estimées comme étant :

$$\mathbf{p}_C = \mathbf{v}_1 + \frac{-\phi(\mathbf{v}_1)}{\phi(\mathbf{v}_2) - \phi(\mathbf{v}_1)}(\mathbf{v}_2 - \mathbf{v}_1) \quad (3.34)$$

Intuitivement, une bonne approximation des deux directions principales de courbure pour une telle surface en \mathbf{p}_C est le plan tangent à l'interface solide Γ_S ainsi que le plan perpendiculaire à l'interface solide Γ_S passant par le vecteur normal à l'interface liquide-gaz Γ_{LG} . On définit un repère orthonormé (ξ, η, τ) local à chaque point \mathbf{p}_C de la triple ligne de contact tel que :

$$\begin{cases} \eta = \nabla\phi_S \\ \tau = \nabla\phi_{LG} - (\nabla\phi_{LG} \cdot \eta)\eta \\ \xi = \eta \times \tau \end{cases} \quad (3.35)$$

Les deux axes principaux sont donc ξ et η . Sachant que trois points suffisent à définir un cercle et que les deux cercles osculateurs passent tous deux par le point de contact \mathbf{p}_C , il nous faut déterminer les quatre autres points qui permettront le calcul des courbures principales.

Le calcul de la courbure κ_ξ relativement à l'axe ξ nécessitent de localiser deux points \mathbf{p}_ξ^+ et \mathbf{p}_ξ^- situés sur l'interface solide (les informations relatives à l'interface liquide-gaz ϕ_{LG} demeurant valides) de part et d'autre du plan $(\mathbf{p}_C, \eta, \tau)$. Dans le cas de la courbure κ_η relative à l'axe η , le point \mathbf{p}_η^+ sera déterminé dans la phase des fluides de la même façon que pour l'axe ξ tandis que le point \mathbf{p}_η^- situé dans le solide sera construit afin de refléter l'angle de contact θ à imposer. On explicite les calculs de ces quatre points :

$$\begin{aligned}
\mathbf{p}_\xi^+ &= (\mathbf{p}_C + \xi) - \phi_{LG}(\mathbf{p}_C + \xi) \nabla \phi_{LG}(\mathbf{p}_C + \xi) \\
\mathbf{p}_\xi^- &= (\mathbf{p}_C - \xi) - \phi_{LG}(\mathbf{p}_C - \xi) \nabla \phi_{LG}(\mathbf{p}_C - \xi) \\
\mathbf{p}_\eta^+ &= (\mathbf{p}_C + \eta) - \phi_{LG}(\mathbf{p}_C + \eta) \nabla \phi_{LG}(\mathbf{p}_C + \eta) \\
\mathbf{p}_\eta^- &= \mathbf{p}_C + \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & \sin(\theta) \\ 0 & -\sin(\theta) & \cos(\theta) \end{pmatrix} \tau^T
\end{aligned} \tag{3.36}$$

Les points dans une phase non solide (soient \mathbf{p}_ξ^+ , \mathbf{p}_ξ^- et \mathbf{p}_η^+) sont en fait calculés par projection orthogonale sur l'interface Γ_{LG} du point de contact translaté en usant des gradients de la phase des fluides ainsi que de leur information de distance. Le point virtuel \mathbf{p}_η^- (qui sera toujours situé dans le solide) est obtenu par translation du point de contact selon l'axe τ (tangent au solide et orienté selon le gradient de la phase des fluides) après rotation de θ degrés selon l'axe ξ .

Afin de calculer les courbures principales κ_ξ et κ_η , il nous faut déterminer les cercles passant respectivement par les triplets de points $(\mathbf{p}_\xi^-, \mathbf{p}_C, \mathbf{p}_\xi^+)$ et $(\mathbf{p}_\eta^-, \mathbf{p}_C, \mathbf{p}_\eta^+)$ ainsi que les signes respectifs des courbures.

On utilise la méthode de López-López [LL92] qui permet de trouver efficacement le centre du cercle circonscrit à un triangle. Soit un triangle défini par ses trois sommets \mathbf{s}_1 , \mathbf{s}_2 et \mathbf{s}_3 ainsi que ses trois segments $\mathbf{S}_1 = \mathbf{s}_2 - \mathbf{s}_3$, $\mathbf{S}_2 = \mathbf{s}_3 - \mathbf{s}_1$ et $\mathbf{S}_3 = \mathbf{s}_1 - \mathbf{s}_2$. Le centre de gravité du triangle est $\mathbf{s}_G = \frac{\mathbf{s}_1 + \mathbf{s}_2 + \mathbf{s}_3}{3}$. On calcule les valeurs scalaires $d_1 = -\mathbf{S}_2 \cdot \mathbf{S}_3$, $d_2 = -\mathbf{S}_3 \cdot \mathbf{S}_1$ et $d_3 = -\mathbf{S}_1 \cdot \mathbf{S}_2$. Le centre du cercle circonscrit \mathbf{c} est alors :

$$\mathbf{c} = \frac{3}{2} \mathbf{s}_G - \frac{d_2 d_3 \mathbf{s}_1 + d_3 d_1 \mathbf{s}_2 + d_1 d_2 \mathbf{s}_3}{2(d_2 d_3 + d_3 d_1 + d_1 d_2)} \tag{3.37}$$

Le rayon du cercle est alors déduit de façon triviale $R = \|\mathbf{s}_i - \mathbf{c}\|$ avec $i \in \{1, 2, 3\}$. Pour des

raisons de stabilité numérique, on notera qu'il est nécessaire de déclencher cette procédure uniquement lorsque les trois points ne sont pas alignés, auquel cas le rayon sera considéré comme infini et la courbure donc nulle.

En substituant respectivement $(\mathbf{p}_\xi^-, \mathbf{p}_C, \mathbf{p}_\xi^+)$ et $(\mathbf{p}_\eta^-, \mathbf{p}_C, \mathbf{p}_\eta^+)$ à $(\mathbf{s}_1, \mathbf{s}_3, \mathbf{s}_2)$ pour obtenir respectivement R_ξ et R_η , on pose pour les approximations des courbures principales :

$$\begin{aligned} \kappa_\xi &= \left(\frac{((\mathbf{p}_C - \mathbf{p}_\xi^-) \times (\mathbf{p}_\xi^+ - \mathbf{p}_C))}{\|((\mathbf{p}_C - \mathbf{p}_\xi^-) \times (\mathbf{p}_\xi^+ - \mathbf{p}_C))\|} \cdot \boldsymbol{\eta} \right) \frac{1}{R_\xi} \\ \kappa_\eta &= - \left(\frac{((\mathbf{p}_C - \mathbf{p}_\eta^-) \times (\mathbf{p}_\eta^+ - \mathbf{p}_C))}{\|((\mathbf{p}_C - \mathbf{p}_\eta^-) \times (\mathbf{p}_\eta^+ - \mathbf{p}_C))\|} \cdot \boldsymbol{\xi} \right) \frac{1}{R_\eta} \end{aligned} \quad (3.38)$$

où le signe de la courbure est déterminé par déduction géométrique de la convexité (signe positif) ou concavité (signe négatif) de la phase liquide. La courbure moyenne qui servira à imposer des conditions aux bords pour la pression à la triple ligne de contact selon le théorème de Laplace est alors obtenue par :

$$\kappa = \frac{1}{2} (\kappa_\xi + \kappa_\eta) \quad (3.39)$$

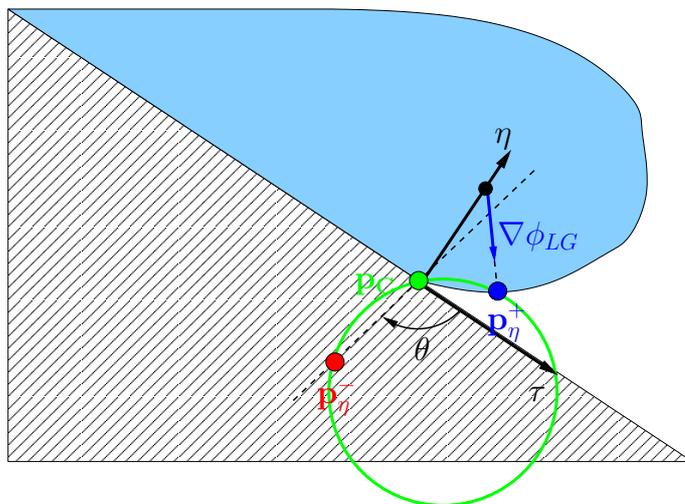


FIG. 3.6: Approximation de la courbure par deux sphères d'axes principaux orthogonaux. Ici, un exemple en 2D est montré (soit, seule la courbure selon l'axe principal η est calculée à l'aide d'un cercle) où deux points de l'interface \mathbf{p}_C et \mathbf{p}_η^+ ainsi qu'un point virtuel \mathbf{p}_η^- imposant l'angle de contact définissent un cercle (en vert) dont la rayon sera celui de courbure.

La figure 3.6 illustre la méthode proposée en 2D.

Approximation de l'interface à la triple ligne de contact par une quadrique

En géométrie différentielle, les courbures d'une surface paramétrisée selon la longueur d'arc sont définies par la dérivée seconde de son équation paramétrique. Intuitivement,

une équation d'ordre deux pour la description d'une surface suffit ainsi aux calculs des courbures. Une surface 2D plongée dans \mathbf{R}^3 définie par une équation quadratique appartient à la famille géométrique des quadriques. On présente ici une méthode permettant d'approcher l'interface à la triple ligne de contact en concordance avec un angle de contact à imposer par une quadrique, la courbure moyenne qui sera utilisée pour le calcul des effets de tensions de surface étant dérivée analytiquement à partir de l'équation de la quadrique [Pet02].

Pour tout point de contact \mathbf{p}_C à la triple ligne de contact, on détermine un repère orthonormé local (ξ, η, τ) ainsi qu'un triplet de points à l'interface liquide-gaz $(\mathbf{p}_\xi^-, \mathbf{p}_\xi^+, \mathbf{p}_\eta^+)$ en concordance avec un point virtuel \mathbf{p}_η^- reflétant l'angle de contact θ à imposer comme précédemment décrit dans la partie 3.2.2. On souhaite trouver une quadrique qui passera au plus près de ces points afin d'y calculer la courbure moyenne en \mathbf{p}_C . On procède comme décrit ci-dessous.

Pour tout point \mathbf{p} de l'espace, ses coordonnées \mathbf{p}' dans le repère $(\mathbf{p}_C, \xi, \eta, \tau)$ sont obtenues selon la formule de changement de repère suivante :

$$\mathbf{p}' = \begin{pmatrix} \xi \\ \eta \\ \tau \end{pmatrix} (\mathbf{p} - \mathbf{p}_C)^T \quad (3.40)$$

On applique ce changement de repère au quintuplet de points $(\mathbf{p}_C, \mathbf{p}_\xi^-, \mathbf{p}_\xi^+, \mathbf{p}_\eta^-, \mathbf{p}_\eta^+)$ et l'on considère la forme quadratique de l'équation de la surface à déterminer :

$$z' = ax'^2 + bx'y' + cy'^2 + dx' + ey' \quad (3.41)$$

On forme un système d'équations linéaires en substituant les coordonnées (x', y', z') dans l'équation 3.41 par les coordonnées respectives de chacun des points $\mathbf{p}'_C, \mathbf{p}'_\xi^-, \mathbf{p}'_\xi^+, \mathbf{p}'_\eta^-$ et \mathbf{p}'_η^+ où les variables à déterminer sont a, b, c, d et e . Par souci de lisibilité, on dénote par (x'_i, y'_i, z'_i) avec $i \in \{1, 2, 3, 4, 5\}$ les coordonnées de chacun de ces points. Le système d'équations linéaires est alors :

$$\begin{cases} z'_1 = ax'_1{}^2 + bx'_1y'_1 + cy'_1{}^2 + dx'_1 + ey'_1 \\ z'_2 = ax'_2{}^2 + bx'_2y'_2 + cy'_2{}^2 + dx'_2 + ey'_2 \\ z'_3 = ax'_3{}^2 + bx'_3y'_3 + cy'_3{}^2 + dx'_3 + ey'_3 \\ z'_4 = ax'_4{}^2 + bx'_4y'_4 + cy'_4{}^2 + dx'_4 + ey'_4 \\ z'_5 = ax'_5{}^2 + bx'_5y'_5 + cy'_5{}^2 + dx'_5 + ey'_5 \end{cases} \quad (3.42)$$

Sous forme matricielle, le système d'équations linéaires

$$\begin{pmatrix} x_1'^2 & x_1'y_1' & y_1'^2 & x_1' & y_1' \\ x_2'^2 & x_2'y_2' & y_2'^2 & x_2' & y_2' \\ x_3'^2 & x_3'y_3' & y_3'^2 & x_3' & y_3' \\ x_4'^2 & x_4'y_4' & y_4'^2 & x_4' & y_4' \\ x_5'^2 & x_5'y_5' & y_5'^2 & x_5' & y_5' \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \\ e \end{pmatrix} = \begin{pmatrix} z_1' \\ z_2' \\ z_3' \\ z_4' \\ z_5' \end{pmatrix} \quad (3.43)$$

Etant donné que nous sommes en présence d'un système de cinq équations linéaires à résoudre pour déterminer la valeur de cinq variables, ce système est déterminé et possède donc au moins une solution. On pourrait procéder par élimination de Gauss afin de résoudre ce système, mais en pratique, une telle démarche est sujette à des instabilités numériques ainsi qu'à l'obligation de gérer certains cas particuliers. On préfère alors une solution approchée obtenue par la méthode des moindres carrés qui minimise la norme du résidu $b - Ax$ pour un système d'équations linéaires de la forme $Ax = b$.

Une fois les variables déterminées, la courbure moyenne à la triple ligne de contact est obtenue par simple dérivation de degré deux de la forme quadratique de la surface :

$$\kappa = \frac{a + c + ae^2 + cd^2 - bde}{(1 + d^2 + e^2)^{\frac{3}{2}}} \quad (3.44)$$

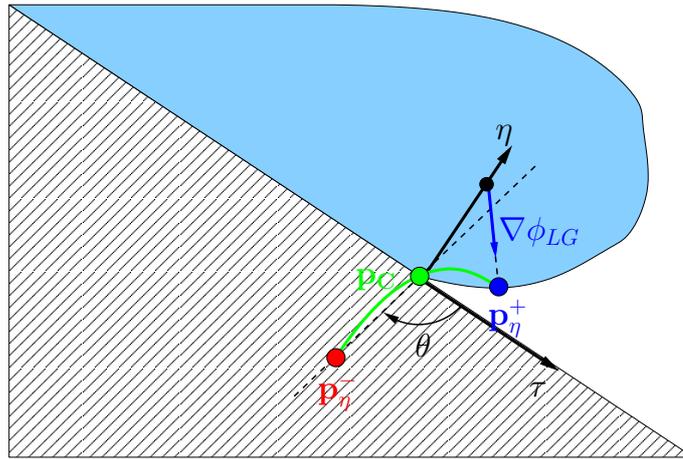


FIG. 3.7: Approximation de la courbure par une quadrique. Ici, un exemple en 2D est montré (soit, seule la courbure selon l'axe principal η est calculée à l'aide d'une conique) où deux points de l'interface \mathbf{p}_C et \mathbf{p}_η^+ ainsi qu'un point virtuel \mathbf{p}_η^- imposant l'angle de contact définissent une parabole (en vert) dont la courbure sera déterminée analytiquement au point de contact \mathbf{p}_C .

La figure 3.7 illustre la méthode proposée en 2D.

Redéfinition de l'opérateur de courbure pour le cas de la triple ligne de contact

Dans un level set, nous avons vu que la courbure moyenne peut être obtenue par application du laplacien sur le champs de distance $\nabla^2\phi$. Comme nous allons le voir, la courbure peut être définie de différentes et diverses manières [LJLC05], et nous proposons de prendre en compte ces formulations afin d'y intégrer l'angle de contact à imposer à la triple ligne de contact. On utilise les informations de repère orthonormé (ξ, η, τ) local en un point de contact \mathbf{p}_C de la triple ligne de contact ainsi que les quatre points interfaciaux $(\mathbf{p}_\xi^-, \mathbf{p}_\xi^+, \mathbf{p}_\eta^-, \mathbf{p}_\eta^+)$ comme définis dans la partie 3.2.2.

- La courbure en un point d'une surface selon un axe peut être défini en tant que la variation d'angle relativement à la longueur d'arc $\frac{\partial\theta}{\partial s}$. Connaissant les deux axes principaux ξ et η pour le calcul des courbures principales κ_ξ et κ_η , on calcule les variations d'angles ainsi :

$$\begin{aligned}\frac{\partial\theta}{\partial s}(\xi) &= \frac{\cos^{-1}\left(\frac{\mathbf{p}_C - \mathbf{p}_\xi^- \cdot \mathbf{p}_\xi^+ - \mathbf{p}_C}{\|\mathbf{p}_C - \mathbf{p}_\xi^-\| \cdot \|\mathbf{p}_\xi^+ - \mathbf{p}_C\|}\right)}{\|\mathbf{p}_C - \mathbf{p}_\xi^-\| + \|\mathbf{p}_\xi^+ - \mathbf{p}_C\|} \\ \frac{\partial\theta}{\partial s}(\eta) &= \frac{\cos^{-1}\left(\frac{\mathbf{p}_C - \mathbf{p}_\eta^- \cdot \mathbf{p}_\eta^+ - \mathbf{p}_C}{\|\mathbf{p}_C - \mathbf{p}_\eta^-\| \cdot \|\mathbf{p}_\eta^+ - \mathbf{p}_C\|}\right)}{\|\mathbf{p}_C - \mathbf{p}_\eta^-\| + \|\mathbf{p}_\eta^+ - \mathbf{p}_C\|}\end{aligned}\quad (3.45)$$

En réutilisant l'évaluation du signe de courbure de la partie 3.2.2, les courbures principales sont alors :

$$\begin{aligned}\kappa_\xi &= \left(\frac{((\mathbf{p}_C - \mathbf{p}_\xi^-) \times (\mathbf{p}_\xi^+ - \mathbf{p}_C))}{\|((\mathbf{p}_C - \mathbf{p}_\xi^-) \times (\mathbf{p}_\xi^+ - \mathbf{p}_C))\|} \cdot \eta \right) \frac{\partial\theta}{\partial s}(\xi) \\ \kappa_\eta &= - \left(\frac{((\mathbf{p}_C - \mathbf{p}_\eta^-) \times (\mathbf{p}_\eta^+ - \mathbf{p}_C))}{\|((\mathbf{p}_C - \mathbf{p}_\eta^-) \times (\mathbf{p}_\eta^+ - \mathbf{p}_C))\|} \cdot \xi \right) \frac{\partial\theta}{\partial s}(\eta)\end{aligned}\quad (3.46)$$

La courbure moyenne est finalement obtenue par $\kappa = \frac{1}{2}(\kappa_\xi + \kappa_\eta)$.

- Une définition alternative pour la courbure selon un axe est celle qui met en relation la dérivée tangentielle et le vecteur normal à l'interface $\frac{\partial\mathbf{T}}{\partial s} = \kappa\vec{N}$. Dans le repère $(\mathbf{p}_C, \xi, \eta, \tau)$, les dérivées tangentielles relatives aux deux axes principaux ξ et η sont approchées comme étant :

$$\begin{aligned}\frac{\partial\mathbf{T}}{\partial s}(\xi) &= \frac{\mathbf{p}_C - \mathbf{p}_\xi^-}{\|\mathbf{p}_C - \mathbf{p}_\xi^-\|} - \frac{\mathbf{p}_\xi^+ - \mathbf{p}_C}{\|\mathbf{p}_\xi^+ - \mathbf{p}_C\|} \\ \frac{\partial\mathbf{T}}{\partial s}(\eta) &= \frac{\mathbf{p}_C - \mathbf{p}_\eta^-}{\|\mathbf{p}_C - \mathbf{p}_\eta^-\|} - \frac{\mathbf{p}_\eta^+ - \mathbf{p}_C}{\|\mathbf{p}_\eta^+ - \mathbf{p}_C\|}\end{aligned}\quad (3.47)$$

On pourrait utiliser l'axe τ comme approximation du vecteur normal à l'interface \vec{N} comme il a été fait dans de précédentes méthodes. Cependant, dans le cas présent, la justesse des courbures dépendant directement de la qualité de l'approximation de \vec{N} , une meilleure précision est de rigueur. On détermine alors le plan passant au mieux à travers les cinq points interfaciaux par une méthode des moindres carrés dont

le vecteur normal orienté selon τ constitue une meilleure approximation du vecteur normal à l'interface \vec{N} . La courbure moyenne est alors obtenue comme étant :

$$\kappa = \frac{1}{2} \left(\frac{\partial \mathbf{T}}{\partial s}(\xi) \cdot \vec{N}^{-1} + \frac{\partial \mathbf{T}}{\partial s}(\eta) \cdot \vec{N}^{-1} \right) \quad (3.48)$$

3.3 Etat des travaux et Conclusion

A l'instant de rédaction du présent manuscrit, les modèles théoriques pour la prise en compte des effets de tension de surface à la triple ligne de contact présentés dans la partie 3.2 ne bénéficient pas d'une implémentation finalisée. Afin de pouvoir mettre en pratique et tester intensivement ces nouveaux modèles numériques, un solveur pour les équations de Navier et Stokes 2D a été développé selon la méthode de solution globale explicite dans la partie 3.1.3, l'implémentation de la méthode des particules level set venant d'être achevée (mais non intégrée au solveur encore). Le choix quant au développement préliminaire d'un solveur 2D (et non directement du solveur 3D final) est justifié par des raisons techniques (erreurs de code plus facilement repérables) mais surtout par la meilleure évaluation de la qualité et de la justesse des résultats qu'elle permet de façon intuitive. Ce choix est d'autant plus justifié par le fait que nous avons adopté une approche générale dite de *dimension par dimension* pour les divers schémas numériques utilisés dans le solveur, rendant l'extension de ce dernier à la troisième dimension directe, simple et efficace.

Malgré la non prise en compte actuelle des effets de tensions de surface à la triple ligne de contact, nous présentons néanmoins quelques résultats préliminaires de simulations 2D où les effets de tensions de surface sont considérés à une interface fluide-gaz. Les résultats présentés ont été obtenus pour des grilles de dimensions 64^2 avec $\Delta x = 1/64$ m pour un pas de temps de $1/24$ secondes contraint à une CFL de $5\Delta x$. Le couple de fluides considéré est l'eau (en bleu) et l'air (en rouge) de propriétés respectives 1000 kg.m^{-3} et 2 kg.m^{-3} pour la densité ρ , $10^{-3} \text{ Pa.s}^{-1}$ et 0 Pa.s^{-1} pour la viscosité dynamique μ et de propriété commune le coefficient de tensions de surface $\gamma = 0.073 \text{ N.m}^{-1}$. Les bords de la grille ont été considérés comme solides avec l'imposition d'une condition aux bords « no-slip ». Le temps de calcul pour un pas de temps complet de la simulation est de l'ordre du dixième de seconde, la projection du champs de pression occupant près de 80% des temps de calcul. On considère que la méthode du gradient conjugué pré-conditionné par une décomposition incomplète de Cholesky a convergé lorsque la norme du résidu devient inférieure à 10^{-8} relativement à la norme du terme de droite du système d'équations linéaires. En pratique, une petite centaine d'itérations en moyenne ont été nécessaires à la convergence de la méthode du gradient conjugué avec le champs de pression toujours initialisé en tant qu'un champs de valeurs nulles. Dans les figures 3.8 3.9 3.10 3.11, les couleurs employées pour la représentation des différentes phases ont été atténuées relativement à la distance interfaciale (soit, plus un voxel sera éloigné de l'interface, plus sa couleur sera rendue opaque).

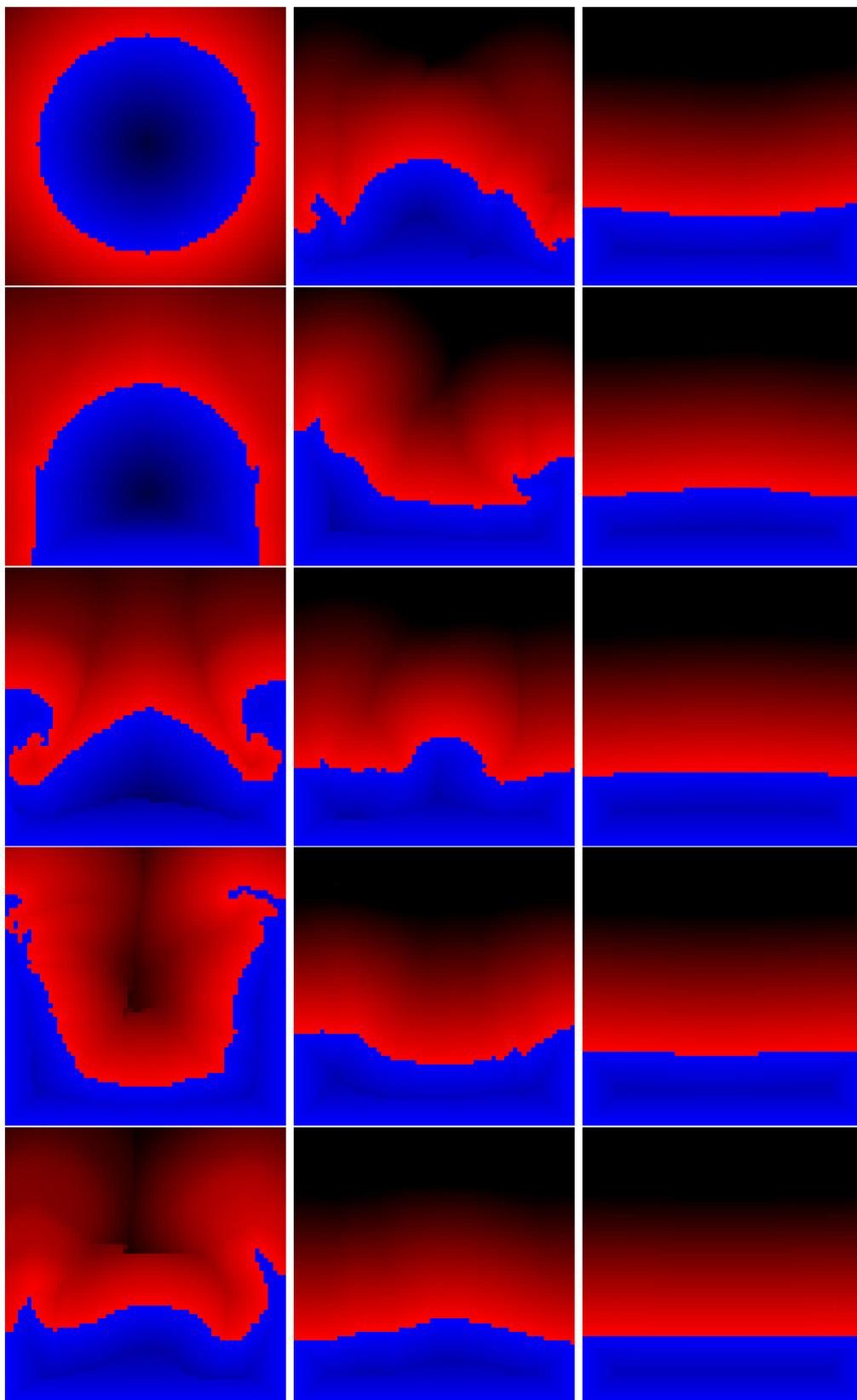


FIG. 3.8: De gauche à droite et de haut en bas, un splash d'eau sans effets de tensions de surface.

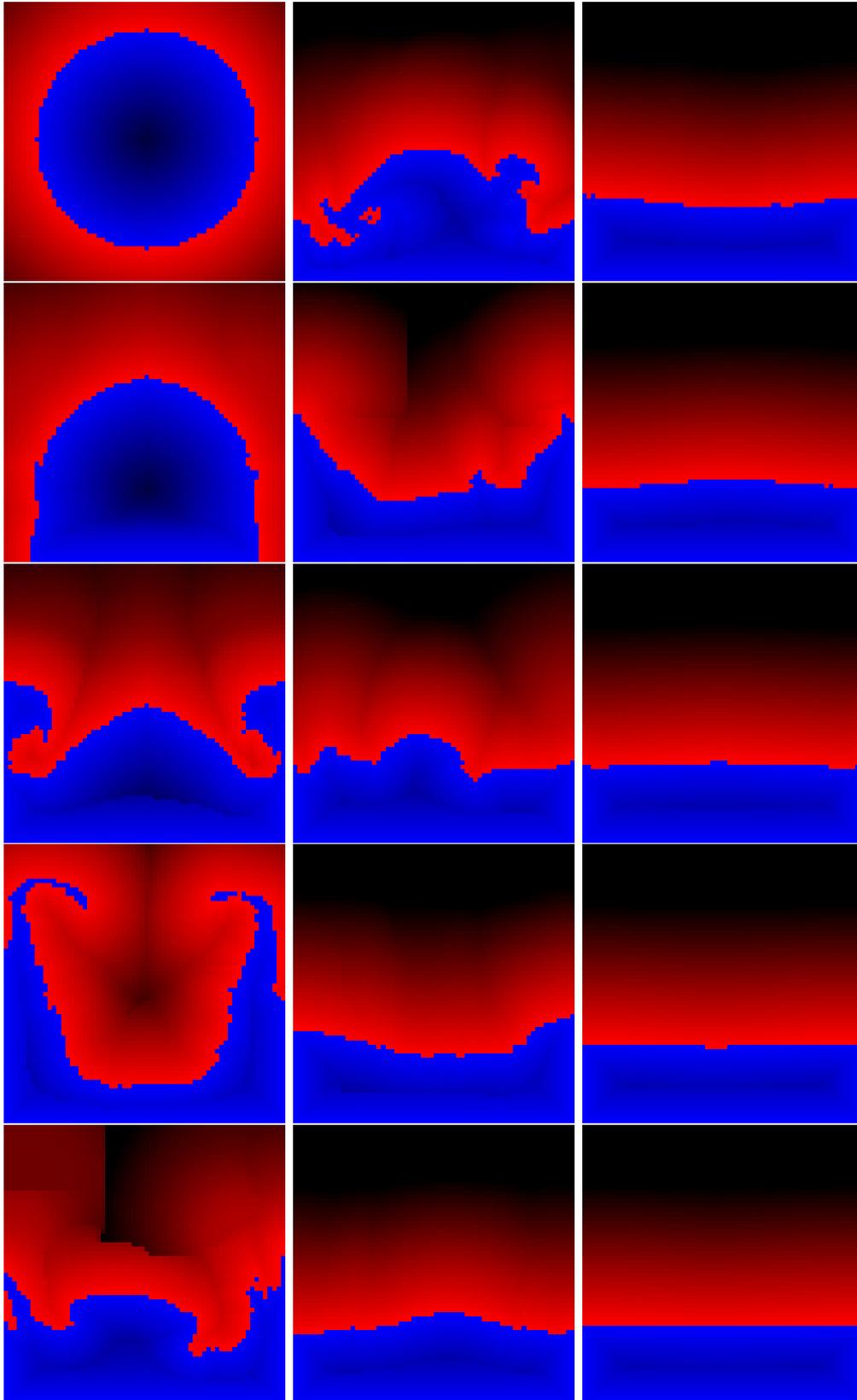


FIG. 3.9: De gauche à droite et de haut en bas, un splash d'eau avec effets de tensions de surface

La figure 3.8 montre un splash d'eau sans les effets de tensions de surface activés pour plusieurs pas de temps successifs, tandis que la figure 3.9 montre la même scène avec les effets de tensions de surface activés à l'aide de la GFM. La dynamique de la goutte liquide est principalement régie par la force gravitationnelle du à sa grande densité comparé à celle de l'air.

La figure 3.10 montre une bulle d'air dans de l'eau sans les effets de tensions de surface activés pour plusieurs pas de temps successifs, tandis que la figure 3.11 montre la même scène avec les effets de tensions de surface activés à l'aide de la GFM. La dynamique de la bulle est principalement régie par le champs de pressions du à la forte différence de densité entre l'air et l'eau, ce qui cause sa remontée à la surface. On remarquera que l'impact des effets de tensions de surface sur la forme de la bulle est plus flagrant comparé à son impact sur la forme d'une goutte. On notera aussi la présence de faibles quantités liquides restant collées sur les parois du solide, puis disparaissant du à une dissipation numérique. Ceci est certainement du respectivement à la condition aux bords « no-slip » imposée sur les bords solides ainsi qu'à la violation de la conservation volumique (ce qui devrait être résolu avec l'intégration au solveur de la méthode des particules level set).

Ces résultats illustrent les effets de tensions de surface sur le comportement d'une interface. Concrètement, la présence de cette force accélère l'atteinte de l'équilibre au sens de la physique, soit, l'interface oscillera moins longtemps si les effets de tensions de surface sont considérés.

Les modèles que l'on a présenté théoriquement se focalisent sur la prise en compte des effets de tension de surface à la triple ligne de contact, ils devraient permettre dans un avenir proche de simuler deux fluides en interaction avec contacts solides comme c'est le cas pour un flot de gouttes. De telles méthodes sont nécessaires si l'on souhaite obtenir des formes et des déplacements de gouttes réalistes, et il n'en existe qu'une seule à ce jour en synthèse d'images [WMT05]. Nous avons proposé plusieurs modèles afin de remédier à certaines des limitations de [WMT05]. En résumé, on propose dans un premier temps de diviser la complexité mémoire du solveur par deux grâce à une évaluation des effets de tensions de surface à la volée, et non à l'aide de la détermination de deux champs de variables supplémentaires à chaque pas de temps. La dynamique de la triple ligne de contact a été simplifiée conceptuellement (mais non qualitativement) grâce à un comportement dorénavant dicté par une heuristique simple issue de travaux reconnus en physique numérique, cette heuristique ne se basant que sur les champs de variables standards des équations de Navier et Stokes (et non sur deux champs de variables supplémentaires comme dans [WMT05]). On utilise la méthode des fluides fantômes (GFM) afin d'intégrer les effets de tensions de surface au solveur contrairement à une imposition de conditions aux bords voxelisées trop approximatives. Les conséquences sont doubles. La GFM permet une intégration des effets de tensions de surface plus juste, ce qui se traduit par des animations plus

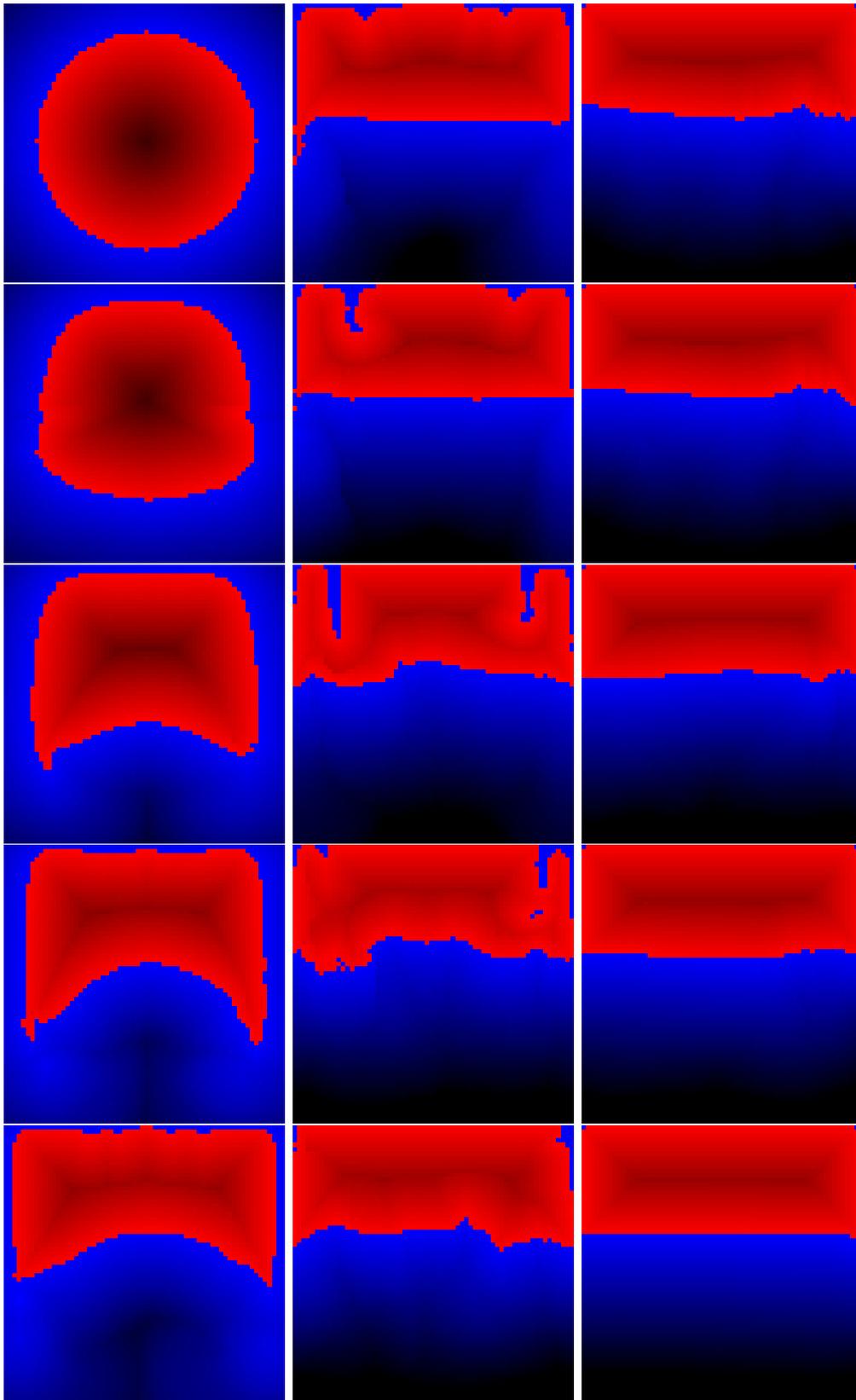


FIG. 3.10: De gauche à droite et de haut en bas, une bulle d'air prisonnière dans de l'eau remontant à la surface sans effets de tensions de surface.

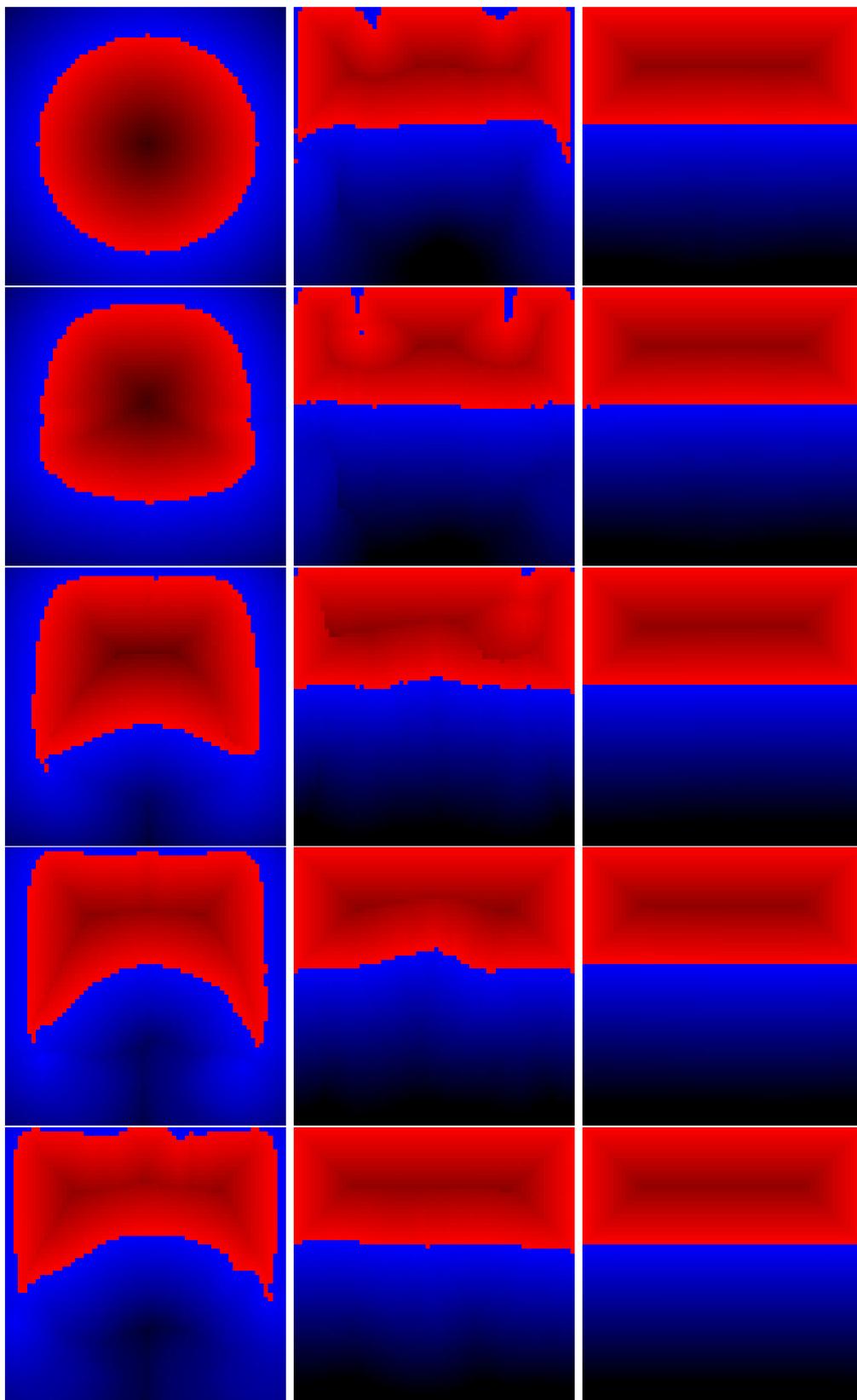


FIG. 3.11: De gauche à droite et de haut en bas, une bulle d'air prisonnière dans de l'eau remontant à la surface avec effets de tensions de surface.

réalistes, avec une précision en dessous de la résolution de la grille, ce qui se traduit par la possibilité d'utiliser des grilles moins fines (soit, une baisse de la complexité mémoire et calculatoire). Finalement, et à titre subsidiaire, dans les travaux de Wang *et al.* [WMT05], une condition aux bords « no-slip » est imposée tout le long du solide. Théoriquement, le déplacement d'un liquide tangentiellement au solide est alors rendu impossible ce qui est plus que problématique dans le cas d'une simulation de faibles écoulements. Reconnu comme une sévère incohérence par la communauté physique depuis plus de trente ans [CL71], il est d'usage depuis de relaxer la contrainte « no-slip » en « free-slip » localement à la triple ligne de contact afin d'autoriser le déplacement de gouttes le long d'un solide. Bien que semblant n'être qu'un détail d'un certain point de vue, nous pensons que cette information est importante à communiquer à la communauté de la synthèse d'images intéressée par l'animation de faibles écoulements surfaciques, la négligence de cette information ayant des conséquences dramatiques sur la convergence de la simulation ainsi que sur la crédibilité des résultats.

Nous espérons présenter sous peu des résultats appuyant tout ce qui a été dit ci-dessus.

Conclusion

L'animation de phénomènes naturels (tels que l'océan, le feu ou les nuages) est réputée dans le domaine de la synthèse d'images pour son aspect non trivial et difficile, des méthodes sophistiquées étant souvent de rigueur en vue de résultats visuellement convaincants. La dynamique des gouttes sur des surfaces solides n'échappe pas à cette généralité et se situe même parmi les phénomènes les plus complexes à reproduire si l'on se place à une échelle suffisamment petite. En synthèse d'images, deux classes de méthodes à part existent quant à l'animation de gouttes.

La première classe de méthodes tente une approche empirique pour la simulation d'un flot de gouttes. De par nature, la conception d'un algorithme empirique requiert de recenser ce que l'on souhaite reproduire afin de lui trouver intuitivement une formalisation (peut-être inspirée de connaissances scientifiques ou autres). En d'autres termes, il est essentiel de se focaliser sur un sous-ensemble des caractéristiques du phénomène (forme, dynamique, propriétés du liquide et du solide, etc.), caractéristiques judicieusement choisies car jugées prédominantes dans un contexte donné (perceptuel ou photo-réaliste par exemple). Les précédentes méthodes empiriques dédiées aux gouttes sont principalement axées sur les performances avec comme objectif la visualisation d'un flot à travers une surface translucide (une vitre ou un pare-brise). De par la nature du contexte, de premières limitations émergent telles que le matériau de la surface, sa géométrie, sa mobilité ainsi que la localisation du point de vue. De plus, afin d'accéder à des performances interactives pour le moins, la forme des gouttes ainsi que leur dynamique sont souvent discontinues (respectivement dans l'espace et le temps).

Dans le chapitre 2, une méthode empirique temps-réel a été proposée afin de lever certaines de ces limitations. Notamment, nous avons fait abstraction sur le matériau de la surface (translucide ou opaque), son orientation dans l'espace et sur le point de vue de l'observateur. De plus, la forme du flot de gouttes est globalement continue, ce qui est rendu possible grâce au schéma de discrétisation employé pour le flot de gouttes associé à un schéma d'advection empirique et simple préservant cette continuité dans le temps. Les gouttes sont visualisables en tant que volumes en dépit de la nature 2D de la simulation grâce à une technique de lancer de rayons temps-réel, ce qui n'était pas le cas pour les précédentes approches. De par ses performances et les résultats visuels convaincants, nous

pensons que cette approche pourrait être intégrée à des applications telles que des simulateurs de conduite ou des jeux vidéos.

Cependant, la forme ainsi que la dynamique obtenues pour les gouttes ne reflètent pas le phénomène de l'angle de contact, la surface étant contrainte à être plane d'autre part. Il serait intéressant et enrichissant de réalisme de trouver une méthodologie où le volume des gouttes est reconstruit relativement à certains paramètres afin de reproduire le phénomène de l'angle de contact, des surfaces paramétriques ou une union de carrés de surface devant permettre de simuler un flot de gouttes sur des surfaces courbes.

Généralement, une approche empirique permettra de prétendre à de hautes performances au prix d'un réalisme sacrifié. Les méthodes basées physiques pour l'animation de phénomènes naturels vont à l'inverse de ce principe en mettant l'accent sur le réalisme des résultats quitte à pâtir de longs temps de calculs. C'est dans cet état d'esprit qu'a été conçue la méthode du chapitre 3. Une des cibles récurrentes de ces simulations numériques est l'animation de grandes étendues d'eau, cas pour lequel les fortes complexités calculatoire et mémoire engendrées les destinent à du rendu en temps différé. Sachant que d'un point de vue équationnel, la complexité d'une simulation pour les gouttes est d'autant plus augmentée, il est d'intérêt pour la communauté de bénéficier d'une méthode la moins prohibitive possible en terme de calculs, soit, que les termes équationnels supplémentaires soient traités le plus efficacement possible. Dans l'unique méthode basée physique dédiée à l'écoulement de gouttes en synthèse d'images, la complexité mémoire d'un solveur physique standard est doublée et la complexité calculatoire augmentée de façon non négligeable au point qu'une animation d'une minute demande plusieurs jours de calculs. Les résultats visuels étant en revanche de qualité et impressionnant de réalisme, nous nous sommes focalisés sur la principale limitation du solveur relative aux gouttes, à savoir l'évaluation numérique des effets de tensions de surface à la triple ligne de contact. En proposant une nouvelle heuristique quant à la dynamique de la goutte en concordance avec différents schémas d'évaluation des tensions de surface tous *à la volée*, la complexité mémoire est divisée par deux et la complexité calculatoire théoriquement plus basse. De plus, les schémas d'évaluation pour les effets de tensions de surface sont issus de la géométrie différentielle et ne constituent pas une approximation 3D d'un modèle 2D comme précédemment. Nous nous attendons en conséquence à ce que des formes de gouttes plus variées et plus justes soient atteintes comme devraient le confirmer les résultats dans un avenir proche.

Une des particularités d'un écoulement de gouttes sur surfaces solides est la faible proportion de l'espace occupée par ces dernières. La méthode basée physique existante ainsi que celle explicitée dans le présent manuscrit usent d'une discrétisation 3D complète de l'espace, ce qui constitue un gaspillage mémoire non négligeable. En perspectives, il serait intéressant d'adapter ou du moins de s'inspirer des techniques de simulation de gaz sur des surfaces comme vu lors de l'état de l'art. Les principales difficultés auxquelles remé-

dier seraient alors la représentation d'une fine bande volumique autour de la surface 2D (afin de modéliser de faibles volumes liquides), l'advection des interfaces (problématique inexistante dans le cadre d'une simulation gazeuse) ainsi que de la prise en compte des effets de tensions de surface à la triple ligne de contact. Les approches surfaciques basées physiques dédiées aux gaz usent soit d'un maillage traditionnel soit d'une discrétisation en carreaux de surfaces pour la résolution des équations physiques. Ce type de discrétisation est responsable de distorsions visuelles et nécessite un solveur adapté complexe d'implémentation. En mathématiques, la paramétrisation conforme est un puissant outil pouvant servir à la résolution d'équations différentielles sur une surface 2D. Des travaux relatifs à ce dernier point devraient être entamer sous peu.

Bibliographie

- [AB08] S. Afkhami and M. Bussmann. Height functions for applying contact angles to 2D VOF simulations. *International Journal for Numerical Methods in Fluids*, 57 :453–472, June 2008.
- [BA05] J. Andreas Baerentzen and Henrik Aanaes. Signed distance computation using the angle weighted pseudonormal. *IEEE Transactions on Visualization and Computer Graphics*, 11(3) :243–253, 2005.
- [BBB07] Christopher Batty, Florence Bertails, and Robert Bridson. A fast variational framework for accurate solid-fluid coupling. *ACM Trans. Graph.*, 26(3) :100, 2007.
- [BKZ92] J. U. Brackbill, D. B. Kothe, and C. Zemach. A continuum method for modeling surface tension. *J. Comput. Phys.*, 100(2) :335–354, 1992.
- [Bli78] James F. Blinn. Simulation of wrinkled surfaces. In *SIGGRAPH '78 : Proceedings of the 5th annual conference on Computer graphics and interactive techniques*, pages 286–292, New York, NY, USA, 1978. ACM.
- [BMC99] M. Bussmann, J. Mostaghimi, and S. Chandra. On a three-dimensional volume tracking model of droplet impact. *Physics of Fluids*, 11(6) :1406–1417, 1999.
- [CdVL95] Jim X. Chen and Niels da Vitoria Lobo. Toward interactive-rate simulation of fluids with moving obstacles using navier-stokes equations. *Graph. Models Image Process.*, 57(2) :107–116, 1995.
- [CdVLHM97] Jim X. Chen, Niels da Vitoria Lobo, Charles E. Hughes, and J. Michael Moshell. Real-time fluid simulation in a dynamic virtual environment. *IEEE Comput. Graph. Appl.*, 17(3) :52–61, 1997.
- [CFK05] Sharen J. Cummins, Marianne M. Francois, and Douglas B. Kothe. Estimating curvature from volume fractions. *Computers & Structures*, 83 :425–434, February 2005.
- [CHMO96] Y. C. Chang, T. Y. Hou, B. Merriman, and S. Osher. A level set formulation of eulerian interface capturing methods for incompressible fluid flows. *J. Comput. Phys.*, 124(2) :449–464, 1996.
- [CL71] Huh C. and Scriven L.E. Hydrodynamic model of steady movement of a solid/liquid/fluid contact line. *J. Colloid Interface*, 35 :85–101, 1971.

- [CL96] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *SIGGRAPH '96 : Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 303–312, New York, NY, USA, 1996. ACM.
- [CM04] Jonathan M. Cohen and M. Jeroen Molemaker. Practical simulation of surface tension flows. In *SIGGRAPH '04 : ACM SIGGRAPH 2004 Sketches*, page 70, New York, NY, USA, 2004. ACM.
- [CMRBVHT02] Mark Carlson, Peter J. Mucha, III R. Brooks Van Horn, and Greg Turk. Melting and flowing. In *SCA '02 : Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 167–174, New York, NY, USA, 2002. ACM Press.
- [CMT04] Mark Carlson, Peter J. Mucha, and Greg Turk. Rigid fluid : animating the interplay between rigid bodies and fluid. In *SIGGRAPH '04 : ACM SIGGRAPH 2004 Papers*, pages 377–384, New York, NY, USA, 2004. ACM Press.
- [dG85] Pierre-Gilles de Gennes. Wetting : statics and dynamics. *Rev. Mod. Phys.*, 57(3) :827–863, Jul 1985.
- [dGBWQ05] Pierre-Gilles de Gennes, Françoise Brochard-Wyart, and David Quéré. *Gouttes, bulles, perles et ondes*. Belin, 2005.
- [DPH96] Julie Dorsey, Hans Köhling Pedersen, and Pat Hanrahan. Flow and changes in appearance. In *SIGGRAPH '96 : Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 411–420, New York, NY, USA, 1996. ACM Press.
- [ELF05] Douglas Enright, Frank Losasso, and Ronald Fedkiw. A fast and accurate semi-lagrangian particle level set method. *Computers & Structures*, 83(6-7) :479–490, February 2005.
- [EMF02] Douglas Enright, Stephen Marschner, and Ronald Fedkiw. Animation and rendering of complex water surfaces. In *SIGGRAPH '02 : Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 736–744, New York, NY, USA, 2002. ACM Press.
- [ETK⁺07] Sharif Elcott, Yiyang Tong, Eva Kanso, Peter Schröder, and Mathieu Desbrun. Stable, circulation-preserving, simplicial fluids. *ACM Trans. Graph.*, 26(1) :4, 2007.
- [FAMO99] Ronald P. Fedkiw, Tariq Aslam, Barry Merriman, and Stanley Osher. A non-oscillatory eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *J. Comput. Phys.*, 152(2) :457–492, 1999.
- [FCD⁺06] Marianne M. Francois, Sharen J. Cummins, Edward D. Dendy, Douglas B. Kothe, James M. Sicilian, and Matthew W. Williams. A balanced-force

- algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework. *J. Comput. Phys.*, 213(1) :141–173, 2006.
- [FF01] Nick Foster and Ronald Fedkiw. Practical animation of liquids. In *SIGGRAPH '01 : Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 23–30, New York, NY, USA, 2001. ACM Press.
- [FHP98] Patrick Fournier, Arash Habibi, and Pierre Poulin. Simulating the flow of liquid droplets. In *Proceedings of the Graphics Interface 1998 Conference*, pages 133–142, Vancouver, BC, Canada, June 1998. Canadian Human-Computer Communications Society.
- [FM96] Nick Foster and Dimitri Metaxas. Realistic animation of liquids. *Graph. Models Image Process.*, 58(5) :471–483, 1996.
- [FM97] Nick Foster and Dimitris Metaxas. Controlling fluid animation. In *CGI '97 : Proceedings of the 1997 Conference on Computer Graphics International*, page 178, Washington, DC, USA, 1997. IEEE Computer Society.
- [FR86] Alain Fournier and William T. Reeves. A simple model of ocean waves. In *SIGGRAPH '86 : Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pages 75–84, New York, NY, USA, 1986. ACM Press.
- [FSJ01] Ronald Fedkiw, Jos Stam, and Henrik Wann Jensen. Visual simulation of smoke. In *SIGGRAPH '01 : Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 15–22, New York, NY, USA, 2001. ACM Press.
- [FZKH05] Z. Fan, Y. Zhao, A. Kaufman, and Y. He. Adapted unstructured lbm for flow simulation on curved surfaces. In *SCA '05 : Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 245–254, New York, NY, USA, 2005. ACM Press.
- [GF99] José Gomes and Olivier D. Faugeras. Reconciling distance functions and level sets. In *SCALE-SPACE '99 : Proceedings of the Second International Conference on Scale-Space Theories in Computer Vision*, pages 70–81, London, UK, 1999. Springer-Verlag.
- [GFCK02] Frederic Gibou, Ronald P. Fedkiw, Li-Tien Cheng, and Myungjoo Kang. A second-order-accurate symmetric discretization of the poisson equation on irregular domains. *J. Comput. Phys.*, 176(1) :205–227, 2002.
- [GH04] S. T. Greenwood and D. H. House. Better with bubbles : enhancing the visual realism of simulated fluid. In *SCA '04 : Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 287–296, Aire-la-Ville, Switzerland, Switzerland, 2004. Eurographics Association.

- [GLN⁺99] Denis Gueyffier, Jie Li, Ali Nadim, Ruben Scardovelli, and Stéphane Zaleski. Volume-of-fluid interface tracking with smoothed surface stress methods for three-dimensional flows. *J. Comput. Phys.*, 152(2) :423–456, 1999.
- [GM77] R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics - Theory and application to non-spherical stars. *Royal Astronomical Society, Monthly Notices*, 181 :375–389, November 1977.
- [Gre86] Ned Greene. Environment mapping and other applications of world projections. *IEEE Comput. Graph. Appl.*, 6(11) :21–29, 1986.
- [GSLF05] Eran Guendelman, Andrew Selle, Frank Losasso, and Ronald Fedkiw. Coupling water and smoke to thin deformable and rigid shells. In *SIGGRAPH '05 : ACM SIGGRAPH 2005 Papers*, pages 973–981, New York, NY, USA, 2005. ACM Press.
- [HJGP08] Jean-François El Hajjar, Vincent Jolivet, Djamchid Ghazanfarpour, and Xavier Pueyo. A model for real-time on-surface flows. *The Visual Computer*, (to appear), 2008.
- [HK03] Jeong-Mo Hong and Chang-Hun Kim. Animation of bubbles in liquid. *Computer Graphics Forum*, 22(3) :253–262, 2003.
- [HK05] Jeong-Mo Hong and Chang-Hun Kim. Discontinuous fluids. *ACM Trans. Graph.*, 24(3) :915–920, 2005.
- [HLYK08] Jeong-Mo Hong, Ho-Young Lee, Jong-Chul Yoon, and Chang-Hun Kim. Bubbles alive. *ACM Trans. Graph.*, to appear, 2008.
- [HN81] C. W. Hirt and B. D. Nichols. Volume of fluid /VOF/ method for the dynamics of free boundaries. *Journal of Computational Physics*, 39 :201–225, January 1981.
- [HW65] Francis H. Harlow and J. Eddie Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids*, 8(12) :2182–2189, 1965.
- [IGLF06] Geoffrey Irving, Eran Guendelman, Frank Losasso, and Ronald Fedkiw. Efficient simulation of large bodies of water by coupling two and three dimensional techniques. *ACM Trans. Graph.*, 25(3) :805–811, 2006.
- [IGPP01] A. Iglesias, A. Gálvez, and J. Puig-Pey. Generating drop trajectories on parametric surfaces. In J. Yu Q. Peng, W. Li, editor, *Proceedings of the Seventh CAD/Graphics'2001*, pages 350–357, Kuming, China, 2001. International Academic Publishers/World Publishing Corporation, Beijing.
- [JEPP04] Jr. James Edward Pilliod and Elbridge Gerry Puckett. Second-order accurate volume-of-fluid algorithms for tracking material interfaces. *J. Comput. Phys.*, 199(2) :465–502, 2004.

- [JS96] Guang-Shan Jiang and Chi-Wang Shu. Efficient implementation of weighted eno schemes. *J. Comput. Phys.*, 126(1) :202–228, 1996.
- [KIY99] Kazufumi Kaneda, Shinya Ikeda, and Hideo Yamashita. Animation of water droplets moving down a surface. *Journal of Visualization and Computer Animation*, 10(1) :15–26, 1999.
- [KKY93] Kazufumi Kaneda, Takushi Kagawa, and Hideo Yamashita. Animation of water droplets on a glass plate. In *Proc. Computer Animation '93*, pages 177–189. Springer-Verlag, 1993.
- [KLLR07] ByungMoon Kim, Yingjie Liu, Ignacio Llamas, and Jarek Rossignac. Advections with significantly reduced dissipation and diffusion. *IEEE Transactions on Visualization and Computer Graphics*, 13(1) :135–144, 2007.
- [KM90] Michael Kass and Gavin Miller. Rapid, stable fluid dynamics for computer graphics. *SIGGRAPH Comput. Graph.*, 24(4) :49–57, 1990.
- [KO96] S. Koshizuka and Y. Oka. Moving-particle semi-implicit method for fragmentation of incompressible fluid. *Nuclear Science Engineering*, 123 :421–434, July 1996.
- [KWF⁺01] A. Kunimatsu, Y. Watanabe, H. Fujii, T. Saito, K. Hiwada, T. Takahashi, and H. Ueki. Fast simulation and rendering techniques for fluid objects. *Computer Graphics Forum*, 20(3) :57–67, 2001.
- [KZYN96] Kazufumi Kaneda, Yasuhiko Zuyama, Hideo Yamashita, and Tomoyuki Nishita. Animation of water droplet flow on curved surfaces. In *Proc. PACIFIC GRAPHICS '96*, pages 50–65, 1996.
- [LC87] William E. Lorensen and Harvey E. Cline. Marching cubes : A high resolution 3d surface construction algorithm. *SIGGRAPH Comput. Graph.*, 21(4) :163–169, 1987.
- [Lew87] J. P. Lewis. Generalized stochastic subdivision. *ACM Trans. Graph.*, 6(3) :167–190, 1987.
- [LGF04] Frank Losasso, Frédéric Gibou, and Ron Fedkiw. Simulating water and smoke with an octree data structure. *ACM Trans. Graph.*, 23(3) :457–462, 2004.
- [LJLC05] Thomas Lewiner, João D. Gomes Jr., Hélio Lopes, and Marcos Craizer. Curvature and torsion estimators based on parametric curve fitting. *Computers & Graphics*, 29(5) :641–655, 2005.
- [LKMU05] H. Liu, S. Krishnan, S. Marella, and H. S. Udaykumar. Sharp interface cartesian grid method ii : a technique for simulating droplet interactions with surfaces of arbitrary shape. *J. Comput. Phys.*, 210(1) :32–54, 2005.
- [LL92] Fernando J. López-López. Triangles revisited. pages 215–218, 1992.
- [LSSF06] Frank Losasso, Tamar Shinar, Andrew Selle, and Ronald Fedkiw. Multiple interacting liquids. *ACM Trans. Graph.*, 25(3) :812–819, 2006.

- [LTKF08] Frank Losasso, Jerry Talton, Nipun Kwatra, and Ronald Fedkiw. Two-way coupled sph and particle level set fluid simulation. *IEEE Transactions on Visualization and Computer Graphics*, 14(4) :797–804, 2008.
- [Luc77] L. B. Lucy. A numerical approach to the testing of the fission hypothesis. *Astronomical Journal*, 82 :1013–1024, December 1977.
- [LZLW05] Y. Q. Liu, H. B. Zhu, X. H. Liu, and E. H. Wu. Real-time simulation of physically based on-surface flow. *The Visual Computer*, 21(8-10) :727–734, 2005.
- [Mar98] R. R. Martin. Estimation of principal curvatures from range data. *Int. J. Shape Modeling*, 3-4(4) :99–109, 1998.
- [MCG03] Matthias Müller, David Charypar, and Markus Gross. Particle-based fluid simulation for interactive applications. In *SCA '03 : Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 154–159, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.
- [MEPS98] Sussman M., Fatemi E., Smereka P., and Osher S. An improved level set method for incompressible two-phase flows. *Computers and Fluids*, 27 :663–680, June 1998.
- [MM99] Alan Murta and James Miller. Modelling and rendering liquids in motion. In V. Skala, editor, *WSCG'99 Conference Proceedings*, pages 194–201, 1999.
- [MMS04] Viorel Mihalef, Dimitris Metaxas, and Mark Sussman. Animation and control of breaking waves. In *SCA '04 : Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 315–324, Aire-la-Ville, Switzerland, Switzerland, 2004. Eurographics Association.
- [MSKG05] Matthias Müller, Barbara Solenthaler, Richard Keiser, and Markus Gross. Particle-based fluid-fluid interaction. In *SCA '05 : Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 237–244, New York, NY, USA, 2005. ACM Press.
- [MST⁺04] Matthias Müller, Simon Schirm, Matthias Teschner, Bruno Heidelberger, and Markus Gross. Interaction of fluids with deformable solids : Research articles. *Comput. Animat. Virtual Worlds*, 15(3-4) :159–171, 2004.
- [MUM⁺06] V. Mihalef, B. Unlusu, D. Metaxas, M. Sussman, and M. Y. Hussaini. Physics based boiling simulation. In *SCA '06 : Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 317–324, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association.
- [MWM87] Gary A. Mastin, Peter A. Watterberg, and John F. Mareda. Fourier synthesis of ocean scenes. *IEEE Comput. Graph. Appl.*, 7(3) :16–23, 1987.

- [NFJ02] Duc Quang Nguyen, Ronald Fedkiw, and Henrik Wann Jensen. Physically based modeling and animation of fire. In *SIGGRAPH '02 : Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 721–728, New York, NY, USA, 2002. ACM Press.
- [NH91] Gregory M. Nielson and Bernd Hamann. The asymptotic decider : resolving the ambiguity in marching cubes. In *VIS '91 : Proceedings of the 2nd conference on Visualization '91*, pages 83–91, Los Alamitos, CA, USA, 1991. IEEE Computer Society Press.
- [NW76] W. F. Noh and P. Woodward. SLIC /simple line interface calculation/. In A. I. van de Vooren and P. J. Zandbergen, editors, *Some Methods of Resolution of Free Surface Problems*, volume 59 of *Lecture Notes in Physics*, Berlin Springer Verlag, pages 330–340, 1976.
- [OF02] Stanley Osher and Ronald Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer-Verlag, 2002.
- [OS88] Stanley Osher and James A. Sethian. Fronts propagating with curvature-dependent speed : algorithms based on hamilton-jacobi formulations. *J. Comput. Phys.*, 79(1) :12–49, 1988.
- [Pea86] Darwyn R. Peachey. Modeling waves and surf. In *SIGGRAPH '86 : Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pages 65–74, New York, NY, USA, 1986. ACM Press.
- [Per85] Ken Perlin. An image synthesizer. In *SIGGRAPH '85 : Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pages 287–296, New York, NY, USA, 1985. ACM Press.
- [Pet02] Sylvain Petitjean. A survey of methods for recovering quadrics in triangle meshes. *ACM Comput. Surv.*, 34(2) :211–262, 2002.
- [POC05] Fábio Policarpo, Manuel M. Oliveira, and João L. D. Comba. Real-time relief mapping on arbitrary polygonal surfaces. *ACM Trans. Graph.*, 24(3) :935–935, 2005.
- [PTB⁺03] Simon Premoze, Tolga Tasdizen, James Bigler, Aaron Lefohn, and Ross T. Whitaker. Particle-based simulation of fluids. In *Proceedings of Eurographics 2003*, pages 401–410, 2003.
- [QWS03] Tiezheng Qian, Xiao-Ping Wang, and Ping Sheng. Molecular scale contact line hydrodynamics of immiscible flows. *Phys. Rev. E*, 68(1) :016306, Jul 2003.
- [QWS04] Tiezheng Qian, Xiao-Ping Wang, and Ping Sheng. Power-law slip profile of the moving contact line in two-phase immiscible flows. *Phys. Rev. Lett.*, 93(9) :094501, Aug 2004.
- [QWS06] T. Qian, X.-P. Wang, and P. Sheng. A variational approach to moving contact line hydrodynamics. *Journal of Fluid Mechanics*, 564 :333–360, October 2006.

- [RRL01] Michael Renardy, Yuriko Renardy, and Jie Li. Numerical simulation of moving contact line problems using a volume-of-fluid method. *J. Comput. Phys.*, 171(1) :243–263, 2001.
- [RR10] Yuriko Renardy and Michael Renardy. Prost : a parabolic reconstruction of surface tension for the volume-of-fluid method. *J. Comput. Phys.*, 183(2) :400–421, r 10.
- [RS00] Giovanni Russo and Peter Smereka. A remark on computing distance functions. *J. Comput. Phys.*, 163(1) :51–67, 2000.
- [Rud97] M. Rudman. Volume-Tracking Methods for Interfacial Flow Calculations. *International Journal for Numerical Methods in Fluids*, 24 :671–691, April 1997.
- [Saa03] Y. Saad. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2003.
- [Sch80] Bruce J. Schachter. Long crested wave models. *Comput Graphics Image Process*, 12(2) :187–201, February 1980.
- [SDY02] Tomoya Sato, Yoshinori Dobashi, and Tsuyoshi Yamamoto. A method for real-time rendering of water droplets taking into account interactive depth of field effects. In Ryohei Nakatsu and Jun’ichi Hoshino, editors, *Entertainment Computing : Technologies and Applications, IFIP First International Workshop on Entertainment Computing (IWEC 2002)*, pages 125–132, Makuhari, Japan, may 2002. Kluwer.
- [Set96] J. Sethian. A fast marching level set method for monotonically advancing fronts. In *Proc. Nat. Acad. Sci.*, volume 93, pages 1591–1595, 1996.
- [SF99] Mark Sussman and Emad Fatemi. An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow. *SIAM J. Sci. Comput.*, 20(4) :1165–1191, 1999.
- [SO88] Chi-Wang Shu and Stanley Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys.*, 77(2) :439–471, 1988.
- [SO89] Chi-Wang Shu and Stanley Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes,ii. *J. Comput. Phys.*, 83(1) :32–78, 1989.
- [SP00] Mark Sussman and Elbridge Gerry Puckett. A coupled level set and volume-of-fluid method for computing 3d and axisymmetric incompressible two-phase flows. *J. Comput. Phys.*, 162(2) :301–337, 2000.
- [Spe05] Peter D. M. Spelt. A level-set approach for simulations of flows with multiple moving contact lines with hysteresis. *J. Comput. Phys.*, 207(2) :389–404, 2005.

- [SSK05] Oh-Young Song, Hyuncheol Shin, and Hyeong-Seok Ko. Stable but non-dissipative water. *ACM Trans. Graph.*, 24(1) :81–97, 2005.
- [SSO94] Mark Sussman, Peter Smereka, and Stanley Osher. A level set approach for computing solutions to incompressible two-phase flow. *J. Comput. Phys.*, 114(1) :146–159, 1994.
- [Sta99] Jos Stam. Stable fluids. In *SIGGRAPH '99 : Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 121–128, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.
- [Sta03] Jos Stam. Flows on surfaces of arbitrary topology. In *SIGGRAPH '03 : ACM SIGGRAPH 2003 Papers*, pages 724–731, New York, NY, USA, 2003. ACM Press.
- [SU98] Mark Sussman and Shotaro Uto. A computational study of the spreading of oil underneath a sheet of ice. Technical Report 98-32, CAM, july 1998.
- [Sus03] Mark Sussman. A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles. *J. Comput. Phys.*, 187(1) :110–136, 2003.
- [SY04] Lin Shi and Yizhou Yu. Inviscid and incompressible fluid simulation on triangle meshes : Research articles. *Comput. Animat. Virtual Worlds*, 15(3-4) :173–181, 2004.
- [TB87] Pauline Y. Ts'o and Brian A. Barsky. Modeling and rendering waves : wave-tracing using beta-splines and reflective and refractive texture mapping. *ACM Trans. Graph.*, 6(3) :191–214, 1987.
- [TIR06] N. Thürey, K. Iglberger, and U. Rüde. Free surface flows with moving and deforming objects for lbm. In Aka GmbH, editor, *Proceedings of Vision, Modeling and Visualization 2006*, pages 193–200. RWTH Aachen, IOS Press, Nov 2006.
- [TKY02] Ruofeng Tong, Kazufumi Kaneda, and Hideo Yamashita. A volume-preserving approach for modeling and animating water flows generated by metaballs. *The Visual Computer*, 18(8) :469–480, 2002.
- [TR04] N. Thürey and U. Rüde. Free Surface Lattice-Boltzmann fluid simulations with and without level sets. In *Workshop on Vision, Modelling, and Visualization VMV*, pages 199–208. IOS Press, 2004.
- [vdPSVW08] S. P. van der Pijl, A. Segal, C. Vuik, and P. Wesseling. Computing three-dimensional two-phase flows with a mass-conserving level set method. *Computing and Visualization in Science*, 11(4) :221–235, 2008.
- [WMT05] Huamin Wang, Peter J. Mucha, and Greg Turk. Water drops on surfaces. *ACM Transactions on Graphics*, 24(3) :921–929, jul 2005.

- [WMT07] Huamin Wang, Gavin Miller, and Greg Turk. Solving general shallow wave equations on surfaces. In *SCA '07 : Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 229–238, Aire-la-Ville, Switzerland, Switzerland, 2007. Eurographics Association.
- [YJC98] Young-Jung Yu, Ho-Youl Jung, and Hwan-Gue Cho. A new rendering technique for water droplet using metaball in the gravitation force. In V. Skala, editor, *WSCG'98 Conference Proceedings*, pages 432–439, Plzen-Bory, Czech Republic, 1998.
- [YJC99] Young-Jung Yu, Ho-Youl Jung, and Hwan-Gue Cho. A new water droplet model using metaball in the gravitational field. *Computers & Graphics*, 23(2) :213–222, 1999.
- [YZZ04] Yonggao Yang, Changqian Zhu, and Hua Zhang. Real-time simulation : Water droplets on glass windows. *Computing in Science and Engg.*, 6(4) :69–73, 2004.
- [ZGZ97] L. Zuowei, C. Guixiang, and Z. Zhaoshun. Numerical simulation of the flow with contact lines. *Acta Mechanica Sinica*, 13 :210–217, August 1997.
- [Zuo96] L. Zuowei. The numerical simulation of the flow with contact lines. *Communications in Nonlinear Science and Numerical Simulations*, 1 :50–56, July 1996.
- [ZYP06] Wen Zheng, Jun-Hai Yong, and Jean-Claude Paul. Simulation of bubbles. In *SCA '06 : Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 325–333, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association.