

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

MÉMOIRE PRÉSENTÉ À
L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

COMME EXIGENCE PARTIELLE
À L'OBTENTION DE LA
MAÎTRISE EN GÉNIE
CONCENTRATION TECHNOLOGIES DE L'INFORMATION
M.Sc.A.

PAR
Bruno ROY

APPRENTISSAGE PAR DICTIONNAIRE POUR LA SYNTHÈSE DES DÉTAILS DE
SURFACE D'UNE SIMULATION DE FLUIDE PAR PARTICULES

MONTRÉAL, LE 12 MARS 2015

© Tous droits réservés, Bruno Roy, 2015

© Tous droits réservés

Cette licence signifie qu'il est interdit de reproduire, d'enregistrer ou de diffuser en tout ou en partie, le présent document. Le lecteur qui désire imprimer ou conserver sur un autre media une partie importante de ce document, doit obligatoirement en demander l'autorisation à l'auteur.

PRÉSENTATION DU JURY

CE MÉMOIRE A ÉTÉ ÉVALUÉ

PAR UN JURY COMPOSÉ DE:

M. Eric Paquette, directeur de mémoire
Département de génie logiciel et des TI à l'École de technologie supérieure

M. Michael J. McGuffin, président du jury
Département de génie logiciel et des TI à l'École de technologie supérieure

M. Christian Desrosiers, membre du jury
Département de génie logiciel et des TI à l'École de technologie supérieure

IL A FAIT L'OBJET D'UNE SOUTENANCE DEVANT JURY ET PUBLIC

LE 27 FÉVRIER 2015

À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

AVANT-PROPOS

Certaines images utilisées dans ce document sont tirées d'articles publiés dans des journaux ou des conférences scientifiques. La source des images est clairement indiquée sous les figures et la référence au travail est également fournie. Ceci est la note de droit d'auteur pour les travaux tirés d'une publication d'ACM :

ACM COPYRIGHT NOTICE. Copyright © 2014 by the Association for Computing Machinery, Inc. Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org or Publications Dept., ACM, Inc., fax +1 (212) 869-0481.

REMERCIEMENTS

Tout d'abord, j'aimerais remercier spécialement mon directeur de recherche, Eric Paquette, pour sa disponibilité et son implication dans le projet. Ses idées et sa rigueur m'ont permis de mener à terme ce projet de recherche. Malgré les changements d'orientation fréquents des travaux, Eric a toujours su m'aiguiller vers une piste de solution. De plus, je tiens à lui témoigner mon appréciation d'avoir cru en moi et en mes compétences. Enfin, son soutien financier, sous forme de bourses, m'a permis de concentrer mon emploi du temps sur mes travaux de recherche. Par la même occasion, je tiens aussi à remercier Christian Desrosiers pour son aide sur des concepts d'apprentissage automatique et sur certaines notions mathématiques associées. J'aimerais aussi remercier l'entreprise Mokko Studio, le Consortium en innovation numérique du Québec (CINQ) et le Conseil de recherche en sciences naturelles et en génie au Canada (CRSNG), sans qui le projet de collaboration entre l'ÉTS et l'entreprise Mokko Studio n'aurait pas été possible. Cette collaboration industrielle m'a permis d'être admis au sein d'une équipe d'artistes compétents du domaine des effets visuels en cinéma. Plus particulièrement, je tiens à remercier les membres de l'équipe de recherche et développement inclus depuis le début de ce projet de collaboration, Jonathan Gagnon et François Dagenais, pour l'intérêt qu'ils ont porté au projet. L'abondance de leurs idées et leur expérience m'ont permis de progresser avec constance dans mes travaux de recherche. De plus, j'aimerais remercier *Side Effects* pour nous avoir fourni des licences de recherche académique. J'aimerais aussi remercier Quentin Avril, chercheur postdoctoral au Laboratoire Multimédia de l'ÉTS, pour l'intérêt qu'il a démontré et ses commentaires pertinents qu'il m'a amenés vers la fin de mon projet de maîtrise. Je tiens aussi à dire merci aux membres de l'équipe de techniciens et de chargés de l'application technologique et informatique du département de génie logiciel et des TI de l'ÉTS, Samuel Docquier, Olivier Rufiange, Patrice Dion et Alain Lemay, pour leur support et leur efficacité. Enfin, j'aimerais remercier tous mes amis pour leur présence durant mes années de recherche à la maîtrise. Parfois même involontairement, leurs visions extérieures sur mon projet m'ont permis de l'analyser sous différents angles. Finalement, j'aimerais spécialement remercier mon épouse Isabelle, pour son support et sa patience pendant ces dernières années. Son amour, son écoute et son enthousiasme m'ont permis d'être productif et positif tout au long de ce projet.

APPRENTISSAGE PAR DICTIONNAIRE POUR LA SYNTHÈSE DES DÉTAILS DE SURFACE D'UNE SIMULATION DE FLUIDE PAR PARTICULES

Bruno ROY

RÉSUMÉ

En infographie, la modélisation de phénomènes naturels, tels que les fluides, est une tâche plutôt complexe et nécessite des temps de calcul importants. De plus, la mise en valeur des détails fins de simulations de fluide, telles que les forces de turbulence en surface, est un sujet actif de recherche dans le domaine des effets visuels. L'intérêt de ces recherches est notamment de conserver ces détails, particulièrement à la surface du fluide.

Il peut être parfois difficile d'obtenir le résultat désiré avec de telles simulations. D'ailleurs, le comportement d'une simulation est fortement dépendant du nombre de particules. Certains éléments importants peuvent ressortir simplement en augmentant le nombre de particules d'une simulation. L'énergie cinétique et les forces de turbulence sont aussi des facteurs importants qui peuvent influencer le comportement d'un fluide. L'approche présentée dans ce mémoire a pour but de recréer l'apparence en surface d'une simulation en haute résolution à partir d'une simulation en basse résolution, et ce, à l'aide d'un apprentissage par dictionnaire. L'apprentissage est fait à partir d'un dictionnaire couplé, c'est-à-dire généré à partir de la concaténation de trois dictionnaires. Les dictionnaires permettent d'établir une correspondance entre la géométrie de la surface des exemples appris et certaines propriétés physiques. Les exemples appris sont représentés par des parcelles en surface du fluide. Notre méthode considère ces parcelles comme des champs de hauteurs. Le spectre d'énergie cinétique et les forces de vorticité sont aussi évalués par parcelle afin de conserver certaines propriétés physiques du fluide. La modélisation des nouvelles parcelles est formulée comme un problème d'optimisation sous contraintes géométriques et physiques. Le résultat de la minimisation permet d'obtenir une combinaison linéaire pondérée des atomes du dictionnaire. Ce vecteur creux pondéré est utilisé pour obtenir une représentation à haute résolution de chaque parcelle à basse résolution en entrée. Enfin, un terme de cohérence spatiale doit être ajouté pour forcer les bordures d'une parcelle à être alignées avec celles de ses voisins.

Cette approche permet de modéliser l'aspect général de la surface d'un fluide en haute résolution, et ce, uniquement à partir de quelques milliers de particules. De plus, ce traitement est fait en une fraction du temps nécessaire pour générer la simulation en haute résolution. Malgré le fait que les objectifs ne soient pas les mêmes qu'avec des méthodes de synthèse de détail par vortex, notre approche peut être utilisée en complément avec ces dernières.

Mots clés: Infographie, simulation de fluide, modélisation procédurale, synthèse de détails, turbulence, apprentissage automatique, dictionnaire

DICTIONARY LEARNING FOR DETAIL SYNTHESIS ON FREE SURFACE OF PARTICLE-BASED FLUIDS

Bruno ROY

ABSTRACT

In computer graphics, modeling natural phenomena, such as fluids, is a complex task and requires significant computation time. Moreover, adding small details in fluid simulations, such as surface turbulence, is an active research topic in the field of visual effects. Our interest in this research is to preserve those small details, especially at the free surface of the fluid.

The discretization of these phenomena requires millions of particles in order to achieve a result similar to the exact solution. However, simulating such a large number of particles is a time consuming process. Besides, the behaviour of a fluid simulation is highly dependent on the number of particles. In addition, it is sometimes difficult to obtain the desired result with such simulations. Some important features may become visible by simply increasing the number of particles of the simulation. The kinetic energy and turbulence forces are also important factors that can influence the behaviour of the fluid. The proposed approach in this master's thesis aims to recreate the surface appearance of a high-resolution fluid based on low- and high-resolution fluid examples from a dictionary learning method. The learning process is made with a coupled dictionary which is generated from concatenated dictionaries. All dictionaries are used to establish a correspondence between the surface geometric features and physical fluid properties from learned examples. These examples are represented by patches of the fluid surface. Our method considers these geometric patches as height fields. The energy spectrum and vorticity forces are also evaluated per patch in order to preserve the observed physical properties of the fluid. Reconstructing new patches is formulated as an optimization problem with geometric and physical constraints. The minimization result is a linear combination of the dictionary atoms. This weighted sparse vector is used to generate a high-resolution representation of each patch of the input low-resolution fluid. In some cases, a spatial term must be added to force the patch borders to be aligned with their neighbors.

This approach allows to model the overall appearance of the high-resolution fluid surface with only a few thousand particles. Moreover, this process is done in a fraction of the computation time required to generate the high-resolution fluid. Despite the fact that the objectives are not the same as with vortex particle methods, our approach can be used together with these to generate more realistic and low-cost high-resolution fluid.

Keywords: Computer graphics, fluid simulation, procedural modeling, detail synthesis, turbulence, machine learning, dictionary

TABLE DES MATIÈRES

	Page
INTRODUCTION	1
CHAPITRE 1 ÉTAT DE L'ART	5
1.1 Simulation de fluide	5
1.1.1 Discrétisation et dissipation numérique	6
1.1.2 Méthodes d'advection d'ordre supérieur	7
1.1.3 Méthodes avec sous-grilles	8
1.2 Modélisation de turbulence et de détails fins	9
1.2.1 Méthodes lagrangiennes par vortex	10
1.2.2 Méthodes avec synthèse de turbulence et modèle d'énergie	11
1.2.3 Méthodes par simulation à petite échelle	14
1.3 Apprentissage automatique et traitement d'image	15
1.3.1 Séparateur à vaste marge	16
1.3.2 Dictionnaire et représentation creuse	18
1.3.2.1 Codage parcimonieux	18
1.3.3 Restauration d'image et reconstruction de surface	21
1.3.3.1 Débruitage et peinture rapportée	21
1.3.3.2 Super-résolution	24
1.3.3.3 Reconstruction de maillage polygonal	24
1.4 Contributions	25
CHAPITRE 2 APPRENTISSAGE PAR DICTIONNAIRES COUPLÉS	27
2.1 Préparation des données d'entraînement	28
2.1.1 Identification et extraction des particules de surface	28
2.1.2 Sélection des parcelles de fluide	32
2.1.3 Normalisation des parcelles	35
2.1.4 Transfert des propriétés	35
2.1.4.1 Propriétés géométriques de la surface	37
2.1.4.2 Propriétés physiques du fluide	39
2.1.5 Modélisation des parcelles	43
2.2 Apprentissage couplé	45
2.2.1 Dictionnaire multivariable	47
2.2.2 Analyse en composantes principales	48
CHAPITRE 3 SYNTHÈSE DE DÉTAILS EN HAUTE RÉSOLUTION	51
3.1 Simulation en basse résolution	52
3.2 Parcelles en basse résolution	52
3.2.1 Sélection des parcelles	52
3.2.2 Modélisation et suréchantillonnage des parcelles	53
3.3 Reconstruction par optimisation	57

3.3.1	Formulation par minimisation	58
3.3.2	Cohérence spatiale et temporelle	61
CHAPITRE 4	IMPLÉMENTATION	65
4.1	Simulateur par particules	65
4.2	Environnement de développement	66
4.3	Préparation des exemples d'entraînement	68
4.3.1	Identification des particules de surface	68
4.3.2	Analyse en composantes principales	68
4.4	Apprentissage des parcelles	69
4.4.1	Stockage des données du dictionnaire	69
4.4.2	Affichage des atomes	70
4.5	Reconstruction de la surface	71
4.6	Paramétrisation et temps de calcul de l'approche	72
CHAPITRE 5	RÉSULTATS	77
CHAPITRE 6	DISCUSSION	85
6.1	Avantages de l'approche	85
6.2	Limitations	86
CHAPITRE 7	TRAVAUX FUTURS	89
CONCLUSION	91	
BIBLIOGRAPHIE	93	

LISTE DES TABLEAUX

	Page	
Tableau 2.1	Covariances entre des propriétés observées des particules.	49
Tableau 4.1	Valeurs des paramètres de simulation.....	73
Tableau 4.2	Paramètres utilisés pour identifier les particules de surface.	73
Tableau 4.3	Valeurs des paramètres d'apprentissage automatique.....	74
Tableau 4.4	Temps de calcul des étapes de la phase d'apprentissage.....	75
Tableau 4.5	Temps de calcul par trame des étapes de reconstruction.....	75
Tableau 4.6	Comparaison des temps de simulation avec l'approche proposée.....	76

LISTE DES FIGURES

	Page	
Figure 1.1	Cellule d'une grille MAC en trois dimensions.....	6
Figure 1.2	Simulation FLIP générée avec <i>Houdini</i>	7
Figure 1.3	Balle qui génère des remous turbulents par induction.....	12
Figure 1.4	Simulation de vaguelettes avec la méthode <i>iWave</i>	14
Figure 1.5	Maximisation de la marge à partir des vecteurs de support.....	17
Figure 1.6	Parcelle par combinaison linéaire.	19
Figure 2.1	Survol des étapes nécessaires pour générer un dictionnaire.....	27
Figure 2.2	Variantes des exemples d'entraînement.	28
Figure 2.3	Identification des particules de surface.....	31
Figure 2.4	Volume englobant pour identification des particules de surface.	32
Figure 2.5	Problème de correspondance de l'énergie cinétique.....	34
Figure 2.6	Correspondance de l'énergie cinétique entre deux simulations.	34
Figure 2.7	Corrélation entre les propriétés d'une particule.	36
Figure 2.8	Comparaison avec le spectre d'énergie théorique.	42
Figure 2.9	Division des parcelles à l'aide de la grille spatiale.	44
Figure 2.10	Marqueurs d'échantillonnage uniforme des particules de surface.	45
Figure 3.1	Survol des étapes de la synthèse de détail par minimisation.	51
Figure 3.2	Évaluation et sélection des parcelles de fluide à reconstruire.	53
Figure 3.3	Artéfacts indésirables de reconstruction aux frontières.	54
Figure 3.4	Seuil erroné avec la méthode du <i>color field</i>	55
Figure 3.5	Échantillonnage par disque de Poisson en deux dimensions.	56
Figure 3.6	Suréchantillonnage par disques de Poisson d'une parcelle.....	56

Figure 3.7	Reconstruction d'une parcelle à partir des atomes du dictionnaire.	57
Figure 3.8	Compromis de reconstruction basé sur les propriétés.	58
Figure 3.9	Cohérence spatiale sur la surface générée.....	61
Figure 4.1	Représentation nodale dans <i>Houdini</i> des étapes de l'approche.	67
Figure 4.2	Représentation polygonale des atomes dans <i>Houdini</i>	71
Figure 4.3	Convergence des points de surface par minimisation.	72
Figure 5.1	Scénario d'une goutte tombant dans un réservoir de liquide.....	77
Figure 5.2	Scénario d'un changement périodique des forces gravitationnelles.	79
Figure 5.3	Scénario d'une rupture de barrage.....	80
Figure 5.4	Problème associé à la séparation du fluide.	81
Figure 5.5	Rendu d'une simulation de fluide avec nuanceur.	81
Figure 5.6	Comparaison de la courbure entre deux simulations.	82
Figure 5.7	Scène problématique avec l'approche proposée.	82

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

ÉTS	École technologie supérieure
ACM	<i>Association for computing machinery</i>
Correction va-et-vient	<i>Back and forth correction</i>
Profile d'interpolation sous contraintes	<i>Constrained interpolation profile</i>
Ligne de niveau	<i>Level set</i>
Peinture rapportée	<i>Inpainting</i>
Codage parcimonieux	<i>Sparse coding</i>
PBF	<i>Position-based fluids</i>
SPH	<i>Smoothed-particle hydrodynamics</i>
SCF	<i>Smoothed-color field</i>
FLIP	<i>Fluid-implicit particle</i>
MAC	<i>Marker-in-cell</i>
LARS	<i>Least angle regression</i>
PCA	<i>Principal component analysis</i>
Pas de temps	Laps de temps simulé lors d'une itération d'un simulateur (<i>time step</i>)
HDK	<i>Houdini development kit</i>
SPAMS	<i>Sparse modeling software</i>
ASCII	<i>American Standard Code for Information Interchange</i>
GPU	<i>Graphics processing unit</i>
CFD	<i>Computational fluid dynamics</i>
Apprentissage automatique	<i>Machine learning</i>
PDS	<i>Poisson disk sampling</i>

API	<i>Application program interface</i>
SOP	<i>Surface operations</i>

LISTE DES SYMBOLES ET UNITÉS DE MESURE

\vec{v}_i	Vitesse de la particule i
\vec{a}_i	Accélération de la particule i
$\vec{\omega}_i$	Vorticité de la particule i
\vec{g}	Force gravitationnelle de la simulation
α	Codage parcimonieux des atomes du dictionnaire
λ	Paramètre de régularisation
c_i	<i>Color field</i> de la particule i
\vec{x}_i	Position de la particule i
\vec{n}_i	Vecteur normal à la surface de la particule i
ε_x	Rayon du cylindre englobant
ε_y	Hauteur du cylindre englobant
W	Matrice de Weingarten
K	Courbure gausienne
H	Courbure moyenne
CV	<i>Curvedness</i>
SI	<i>Shape index</i>
SR	<i>Sphericity ratio</i>
\vec{f}_i	Forces externes de la particule i
p_i	Pression de la particule i
ρ_i	Masse volumique de la particule i
m_i	Masse de la particule i
\vec{x}_{ij}	$\vec{x}_i - \vec{x}_j$
W_{ij}	Noyau $W(\vec{x}_{ij}, h)$

h	Rayon d'influence d'un noyau
E_k	Énergie cinétique totale d'une simulation
e_k^i	Énergie cinétique de la particule i
\hat{u}^k	Décomposition de la vitesse de l'octave k
D	Dictionnaire constitué d'un ensemble d'atomes
d	Atome d'un dictionnaire
C	Matrice de covariance
$cov(p_i, p_j)$	Covariance entre les propriétés i et j d'une particule
s_c	Seuil de surface du <i>color field</i>
s_f	Seuil de frontière des particules de surface

INTRODUCTION

Contexte

Depuis déjà deux décennies, les simulations de fluide ont attiré l'attention de plusieurs chercheurs et artistes du domaine des effets visuels. Les applications des simulations de fluide sont multiples et variées, de l'ingénierie au cinéma. Par opposition à l'ingénierie mécanique où l'objectif principal est d'obtenir des modèles physiques précis, les buts en infographie sont le réalisme visuel, la contrôlabilité, l'efficacité et la stabilité. En postproduction cinématographique, la composition nécessite de travailler avec un fluide plausible et avec une résolution suffisamment élevée pour capturer tous les détails importants. De plus, dans le but d'obtenir le résultat désiré, ces phénomènes numériques doivent offrir un niveau de contrôle adéquat aux animateurs.

D'autre part, les comportements chaotiques d'un fluide sont associés à leur nature turbulente. Un fluide peu visqueux à grande vitesse risque rapidement de devenir instable et par le fait même, passer d'un état laminaire à un état hautement turbulent. Le simple fait qu'une petite instabilité peut se traduire en changement important de l'aspect à grande échelle du fluide est une preuve de la difficulté d'interprétation et de contrôle des écoulements turbulents. D'ailleurs, puisque ces petits détails ont une influence sur l'ensemble du fluide, ils ne doivent pas être négligés par le modèle de simulation. Une résolution suffisamment élevée permet un meilleur suivi de ces détails fins.

Problématique et hypothèse

Malgré le fait que certaines scènes des films d'aujourd'hui présentent des simulations de fluide convaincantes, certains des problèmes associés à ces phénomènes sont toujours présents. Dans un premier temps, obtenir des résultats aussi convaincants nécessite un grand nombre de manipulations de l'artiste. L'artiste doit souvent effectuer des tâches manuelles exigeantes (*e.g.* assembler plusieurs simulations). De plus, les effets visuels concernant les fluides sont souvent améliorés à l'aide de nombreuses couches de composition. Pour en masquer les défauts,

ces artistes des fluides doivent fréquemment agencer plusieurs simulations superposées afin d'atteindre le résultat désiré.

Ensuite, le comportement du fluide est fortement dépendant de sa résolution. En ce qui concerne les simulations par particules, plus il y a de particules et plus les détails fins de la simulation vont ressortir à la surface libre du fluide, et ce, à plusieurs échelles. Un nombre impressionnant de travaux ont été menés pour améliorer ces détails à petites échelles par l'ajout d'un bruit turbulent dans le champ de vitesses, l'intégration d'un modèle d'énergie et de turbulence ou même la simulation de vaguelettes dans les hautes fréquences de la simulation. Ces approches permettent d'augmenter le niveau de détail des simulations en haute résolution. La surface résultante est une extrapolation faite à partir des particules de la simulation en haute résolution. Cela signifie que l'apparence à grande échelle de la simulation n'est pas nécessairement similaire à celle d'une simulation dont la résolution est plus élevée. L'ajout de détails par méthode procédurale ne permet pas de retrouver cette apparence. Conséquemment, le problème associé aux différences d'apparence causées par la résolution d'une simulation n'est toujours pas résolu.

Avec l'approche proposée, nous introduisons une méthode permettant de retrouver une apparence similaire à une simulation de résolution supérieure. Tout d'abord, nous faisons la démonstration qu'il existe une corrélation simple entre l'apparence de la surface du fluide et certaines propriétés physiques des particules de celui-ci. Cette relation nous donne la possibilité d'apprendre un modèle permettant de retrouver cette apparence du fluide à partir d'échantillons de résolution inférieure aux exemples d'entraînement. De plus, nous faisons l'hypothèse que cette nouvelle apparence peut être posée sous une formulation de minimisation. Cette minimisation sous contraintes doit permettre de conserver les différences entre deux résolutions distinctes de simulations similaires.

Organisation du document

Le chapitre 1 présente une revue de littérature sur les simulations de fluide en infographie, les méthodes de modélisation de la turbulence et d'ajout procédural de détails fins ainsi qu'une

brève introduction à l'apprentissage automatique. Au chapitre 2, la méthode d'apprentissage par dictionnaire utilisée par l'approche est détaillée. Ce chapitre inclut la préparation des données d'entraînement et l'apprentissage des exemples de parcelles de fluide par dictionnaire couplé. Par la suite, l'approche développée pour déterminer une solution de la nouvelle surface par minimisation est expliquée au chapitre 3. Certaines précisions concernant l'implémentation de l'approche sont présentées au chapitre 4. Le chapitre 5 présente plusieurs résultats obtenus à l'aide de cette approche. Une discussion portant sur les expérimentations et sur les résultats est exposée au chapitre 6. Finalement, des travaux futurs sont introduits au chapitre 7.

CHAPITRE 1

ÉTAT DE L'ART

1.1 Simulation de fluide

Avant de s'exposer à une revue de littérature détaillée sur l'amélioration des modèles de synthèse de détail en simulation de fluide, une brève introduction des équations fondamentales de la mécanique des fluides nous paraît essentielle. Les simulations de fluide sont gouvernées par les équations incompressibles de Navier-Stokes (Navier, 1823). Ces équations sont formulées en deux parties et définissent le mouvement d'un fluide :

$$\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} + \frac{1}{\rho} \nabla p = \vec{g} + \nu \cdot \nabla \vec{u} \quad (1.1)$$

$$\nabla \cdot \vec{u} = 0 \quad (1.2)$$

La vitesse et la force de pression sont respectivement représentées par \vec{u} et p . La première équation différentielle (1.1) est l'équation de la quantité de mouvement (*momentum equation*). L'équation 1.1 contrôle l'accélération du fluide par rapport aux forces internes et externes qui agissent sur celui-ci. En isolant $\frac{\partial \vec{u}}{\partial t}$ des autres termes de l'équation, on obtient la variation de vitesse par rapport au temps :

$$\frac{\partial \vec{u}}{\partial t} = \nu \cdot \nabla \vec{u} - \vec{u} \cdot \nabla \vec{u} - \frac{1}{\rho} \nabla p + \vec{f} \quad (1.3)$$

Dans cette formulation, la variation de la vitesse dans le temps est définie respectivement par un terme de viscosité, un terme d'advection, un terme de pression et des forces externes. La formulation et le développement de l'accélération et des forces externes \vec{f} (force gravitationnelle \vec{g} , turbulence $\vec{\omega}$, etc.) dans un contexte avec les *smoothed particle hydrodynamics* (SPH) sont détaillés à la section 2.1.4.2 de ce document. La deuxième équation (1.2) est appelée la condition d'incompressibilité. Cette équation assure la conservation du volume de fluide dans le temps. De plus, un fluide dont le champ vectoriel de vitesses est dit sans divergence est né-

cessairement incompressible. Puisque les particules subissent des forces d'attraction et de répulsion, il n'est pas possible d'avoir un fluide entièrement incompressible. Toutefois, les forces de pression (section 2.1.4.2) permettent d'obtenir un fluide à l'état presque incompressible.

1.1.1 Discréétisation et dissipation numérique

En infographie, la simulation de fluide est devenue un sujet actif de recherche suite à la publication des travaux de Stam (1999) qui a introduit l'advection semi-lagrangienne à l'aide de la projection de la pression en utilisant une méthode par différences finies, c'est-à-dire la discréétisation *Marker-In-Cell* (Harlow *et al.*, 1965). Cette discréétisation des équation de Stokes est faite au centre pour la pression et sur les faces d'une cellule de la grille échelonnée tel qu'illustré à la figure 1.1. Ce type de solveur offre une stabilité inconditionnelle, mais n'est

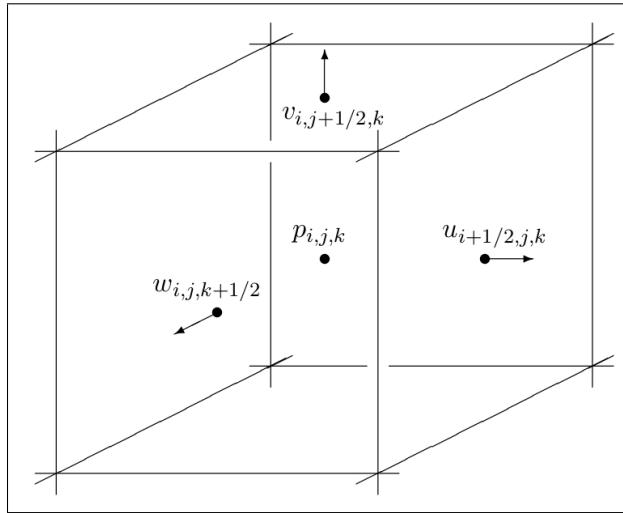


Figure 1.1 Cellule d'une grille MAC en trois dimensions.
Tirée de Bridson (2008).

pas à l'abri d'importantes dissipations numériques. Par la suite, plusieurs travaux se sont basés sur ce solveur de premier ordre. Par exemple, la simulation de liquide (Enright *et al.*, 2002), l'émission de bulles (Hong et Kim, 2003), les fluides viscoélastiques (Goktekin *et al.*, 2004) et même l'interaction entre un fluide et des corps rigides (Carlson *et al.*, 2004) ont tous tiré avantage de ce type d'approche. Nécessairement, dû à la nature dissipative du solveur utilisé, ces

approches n'ont pas été en mesure de bien gérer les détails fins présents dans une simulation de fluide. Ce problème est d'autant plus apparent dans une simulation en haute résolution. Des solutions à ce problème ont été abordées par les approches présentées à la section suivante.

1.1.2 Méthodes d'advection d'ordre supérieur

Ces détails fins perdus par la dissipation numérique sont en fait la plupart du temps les effets de la turbulence. Plusieurs des approches qui ont été proposées tentent de réduire au minimum la dissipation numérique causée par la discréétisation. Une méthode populaire pour résoudre ce problème est d'utiliser une advection d'ordre supérieur. De cette façon, il est possible d'augmenter la précision de l'advection semi-lagrangienne avec une approche de second et de troisième ordre. Quelques approches ont été proposées au fil des ans, par exemple avec la correction du mouvement va-et-vient (Kim *et al.*, 2005), avec l'advection de MacCormack (Selle *et al.*, 2008), ou bien avec la méthode du par profile d'interpolation sous contraintes (Kim *et al.*, 2008a). Par la suite, une approche PIC (Zhu et Bridson, 2005), originalement tirée des

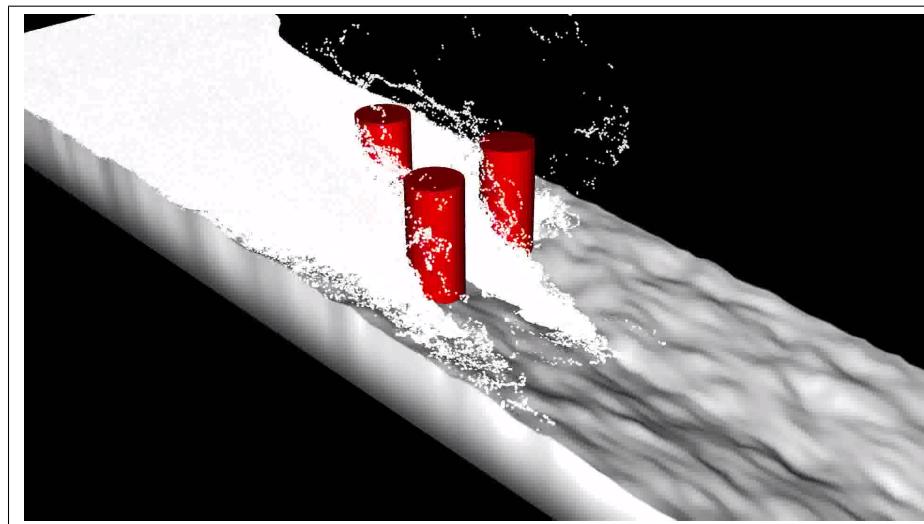


Figure 1.2 Simulation de 5 millions de particules FLIP simulée avec le logiciel *Houdini*.

travaux de Harlow (1962) et amélioré par Brackbill et Ruppel (1986) en dynamique compu-

tationnelle des fluides (CFD), permettant l'utilisation d'un ensemble additionnel de particules servant de marqueurs à l'intérieur d'une grille échelonnée (MAC). La figure 1.2 présente un exemple d'une simulation *Fluid-implicit particle* (FLIP). Ces méthodes tirent avantage d'une grille pour améliorer les temps de calcul, mais conservent un niveau de détail élevé à l'aide de la représentation du fluide par des particules. De plus, Mullen *et al.* (2009) ont proposé une méthode d'intégration et de conservation d'énergie basée sur la vitesse discrétisée à l'aide d'un maillage tétraédrique.

Parallèlement, Fedkiw *et al.* (2001) se sont concentrés sur l'amplification de la vorticité présente dans les flux en basse résolution. Jang *et al.* (2010) ont utilisé un principe similaire permettant de gérer le confinement de ces flux, et ce, à plusieurs échelles. Il est aussi possible de semer des particules de vorticité dans les emplacements où une énergie de turbulence peut être détectée afin de remédier à la dissipation numérique (Selle *et al.*, 2005). La faiblesse de toutes les méthodes présentées dans cette section est nécessairement la dépendance de celles-ci à la résolution de la grille utilisée.

1.1.3 Méthodes avec sous-grilles

D'autres approches ont réduit les effets de la dissipation numérique en raffinant au besoin certaines cellules de la grille de simulation. Par exemple, Losasso *et al.* (2004) ont su tirer avantage d'une structure par arbre octaire afin de discrétiser adaptativement la grille de simulation. Par ailleurs, l'analyse par couche bi-dimensionnelle combinée avec la simulation en trois dimensions a donné des résultats plutôt convaincant (Irving *et al.*, 2006; Chentanez et Müller, 2011). Néanmoins, ces approches sont excessivement coûteuses en temps de calcul et se sont révélées efficaces uniquement dans certains cas spécifiques où les régions d'intérêt sont en surface du fluide.

Dans un autre ordre d'idée, l'utilisation de marqueurs en observation lagrangienne permet aussi de suivre la progression des quantités physiques d'un fluide et par le fait même, d'atténuer l'effet de dissipation. Par exemple, pour un liquide, il est possible d'utiliser des marqueurs afin

d'accroître la résolution des lignes de niveau (Enright *et al.*, 2002). Une variante à cette dernière méthode est l'utilisation d'un maillage triangulaire à l'interface entre l'air et le liquide par Bargteil *et al.* (2006) et Wojtan *et al.* (2010). D'autres ont aussi utilisé un diagramme de Voronoï afin de subdiviser la grille en surface d'un liquide (Brochu *et al.*, 2010). Ce type d'approche a aussi été utilisé pour suivre et représenter l'interface entre l'air et un gaz (Brochu et Bridson, 2009). Thürey *et al.* (2010) ont préféré la formulation lagrangienne leur permettant de placer des tenseurs de surface sur les particules d'interface. Malgré le fait qu'une formulation lagrangienne permet de faire ressortir des détails fins que la grille n'est pas apte d'évaluer, la dynamique reste limitée à la résolution de la grille du champ de vitesses.

Les approches utilisant une grille adaptative sont particulièrement efficaces lorsqu'il est question d'évaluer l'interaction entre un fluide et un corps rigide. Ces interactions sont notamment les effets d'induction entre un corps rigide et les forces de turbulence. Certains travaux ont mis l'emphase sur l'effet entre la phase discrète et le continuum (et vice versa), c'est-à-dire le couplage bidirectionnel entre un fluide et un corps rigide (Guendelman *et al.*, 2003; Carlson *et al.*, 2004; Klingner *et al.*, 2006). Finalement, Batty *et al.* (2007) ainsi que Robinson-Mosher *et al.* (2008) ont amélioré la précision des interactions avec des corps rigides en reformulant la traditionnelle projection de pression par une minimisation à partir de l'énergie cinétique.

1.2 Modélisation de turbulence et de détails fins

La modélisation procédurale de détails fins, tel que la turbulence en surface, est une méthode permettant de coupler une approche de synthèse de détail avec une simulation de fluide générée préalablement. Les approches de synthèse de détails peuvent être basées sur l'ajout d'un bruit, sur une fonction de turbulence avec un modèle de transport d'énergie ou sur l'ensemencement de particules de vorticité. Dans certains cas, le champ de vitesses de la simulation de fluide est utilisé pour générer des détails cohérents à petite échelle.

1.2.1 Méthodes lagrangiennes par vortex

Tout d'abord, il est important de noter que les équations du mouvement d'un fluide définies par Navier-Stokes peuvent être résolues sous une formulation de vorticité (équation 1.5), et ce, par opposition à la formulation traditionnelle utilisant la vitesse (équation 1.4).

$$\vec{u}_t + (\vec{u} \cdot \nabla) \vec{u} + \nabla \frac{p}{\rho} = \mu \nabla^2 \vec{u} + \vec{f} \quad (1.4)$$

$$\vec{\omega}_t + (\vec{u} \cdot \nabla) \vec{\omega} - (\vec{\omega} \cdot \nabla) \vec{u} = \mu \nabla^2 \vec{\omega} + \nabla \times \vec{f} \quad (1.5)$$

La vorticité décrit la rotation du flux d'un fluide. Cette force rotationnelle peut être déterminée par l'équation ci-dessous :

$$\vec{\omega} = \nabla \times \vec{u} \quad (1.6)$$

Toutefois, les conditions aux frontières, l'interaction avec les obstacles et les surfaces libres sont mieux évaluées par une représentation par la vitesse. Les méthodes par vortex sont la plupart du temps utilisées sous une formulation lagrangienne. Néanmoins, un bon nombre d'entre elles utilisent une grille, ou bien un équivalent d'une représentation eulérienne, en complément. Cette grille est utilisée pour estimer les vitesses qui seront transférées aux particules de vorticité. De cette façon, les temps de calcul sont réduits pour les particules ajoutées à la simulation.

Une des approches les plus intuitives et utilisées est la représentation de la vorticité par les particules « vorton » sous formulation lagrangienne (Novikov, 1983). Les premiers essais utilisant ces éléments de vorticité en trois dimensions ont été introduits par Beale et Majda (1982). Par la suite, Marshall et Grant (1996) ont ajouté une grille à l'approche afin de stabiliser le terme d'étirement des tourbillons au centre de la vorticité. En infographie, les méthodes par vortex ont été principalement utilisées pour ajouter des détails localement et à petite échelle. L'objectif est d'augmenter l'effet de la turbulence déjà présente dans un fluide, par exemple avec la semence de « vortons » (Rosenhead, 1931). D'ailleurs, Selle *et al.* (2005) utilisent des

« vortons » pour améliorer et faire ressortir l'effet de turbulence d'une simulation eulérienne en basse résolution.

D'autres approches ont plutôt mis l'emphase sur le contrôle de la vorticité produite dans un flux (Angelidis et Neyret, 2005; Angelidis *et al.*, 2006). Les nappes de tourbillon ont ensuite été introduites par Kim *et al.* (2009) afin de représenter la turbulence dans certains scénarios plus complexes (*e.g.* des éclaboussures). L'utilisation d'une grille afin de synthétiser la vorticité confinée est encore nécessaire et représente une grande limitation pour ce type de méthode.

1.2.2 Méthodes avec synthèse de turbulence et modèle d'énergie

En infographie, les premières méthodes de modélisation de la turbulence ont utilisé des techniques de synthèse afin d'augmenter ou de remplacer une simulation de fluide. Stam et Fiume (1993) ont introduit une méthode basée sur le spectre d'énergie de Kolmogorov dans le but de produire un champ de turbulence sans divergence. Ce type de modèle a aussi été utilisé pour simuler des explosions et de la fumée à grande échelle (Lamorlette et Foster, 2002; Rasmussen *et al.*, 2003). Une approche procédurale utilisant un champ vectoriel de bruit rotationnel a été proposée par Bridson *et al.* (2007) afin de produire un champ de vitesses sans divergence et intégrant certaines propriétés de turbulence. Cette approche assez intuitive a permis d'obtenir des résultats convaincants dans un contexte où un fluide à l'état gazeux subit des forces de turbulence induite. Néanmoins, ces méthodes ne considèrent pas la distribution spatiale et temporelle de l'énergie produite et dissipée par modèle de turbulence. Par conséquent, cette lacune peut produire des résultats invraisemblables (niveau faible de réalisme) avec des scénarios complexes. Certaines méthodes plus récentes ont proposé d'estimer les écoulements à l'aide de méthodes statistiques afin d'extraire plus efficacement les caractéristiques de la turbulence. Par exemple, la décomposition par ondelette (Kim *et al.*, 2008b) permet de déterminer l'emplacement de la turbulence et par le fait même, évaluer certaines caractéristiques d'énergie manquantes dans les hautes fréquences en surface d'un fluide. Par la suite, il est possible de synthétiser une force de turbulence à l'aide d'un bruit rotationnel. Bien entendu, ce type de méthode fait l'hypothèse que la turbulence a été conservée par le solveur du simulateur, ce

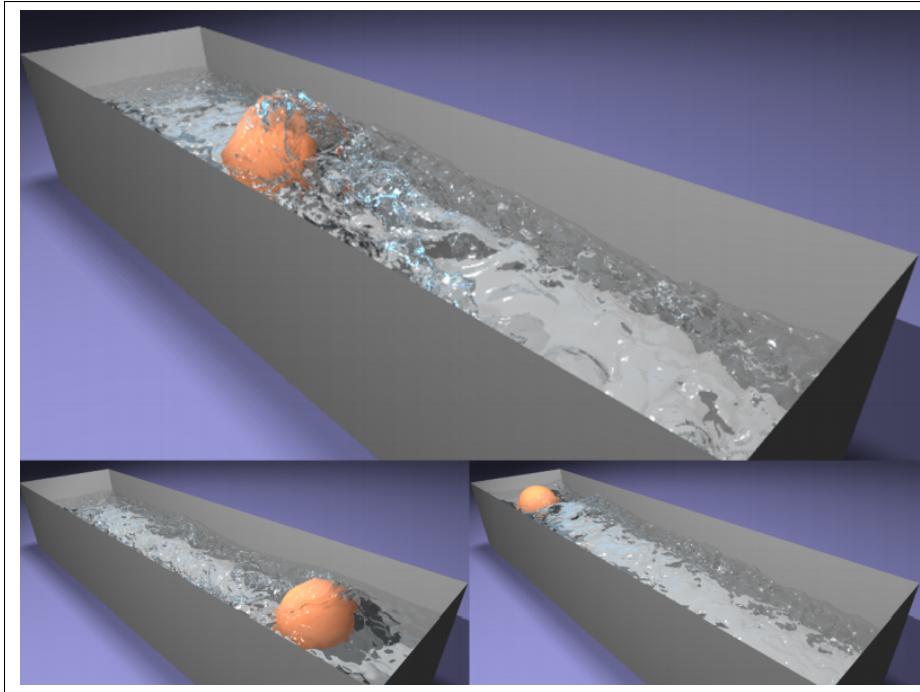


Figure 1.3 Balle qui génère des remous turbulents par induction.
Tirée de Narain *et al.* (2008).

qui n'est pas toujours le cas. Certains scénarios complexes à basse résolution ne permettront pas de conserver un niveau de détail et de précision adéquat à la production de turbulence. Par exemple, avec une simulation lagrangienne, un nombre élevé de particules permet de reproduire fidèlement les tourbillons générés à petite échelle par les forces de turbulence. Pour ces situations, l'utilisation d'un modèle de transport de la turbulence permet de conserver et de suivre l'énergie de turbulence à grande et petite échelle. Narain *et al.* (2008) ont proposé de combiner le modèle d'énergie de Kolmogorov et des éléments de sous-grille pour conserver ce niveau de précision (figure 1.3). En parallèle, Schechter et Bridson (2008) a simplifié un modèle de transport k-epsilon afin d'instancier un champ de vitesse à l'aide des différentes octaves d'énergie. Plus récemment, Pfaff *et al.* (2009) ont proposé une approche combinant un modèle d'énergie complet et physiquement plausible avec une méthode de synthèse.

D'autres approches utilisent une méthode de synthèse pour générer des détails en surface du fluide à l'aide d'observations faites sur les propriétés du fluide à grande et petite échelle. No-

tamment, les approches permettant de synthétiser la turbulence sont au centre de ces avancées depuis quelques années. Chen *et al.* (2011) présentent une approche hybride de modélisation de la turbulence qui permet de synthétiser ces forces directement sur le système de particules. Cette synthèse est faite à l'aide d'une approche par fonction de densité de probabilité. Un système sous forme d'équation différentielle stochastique peut ensuite être résolu. Toutefois, il est à noter que les vitesses résultantes de cette méthode ne sont pas sans divergence. D'autres auteurs ont préféré utiliser un champ de forces précalculé afin de prévenir cette divergence (Zhao *et al.*, 2010). Ce champ est fidèle au spectre d'énergie par cascade entre les forces à grande échelle et celles à petite échelle. Néanmoins, la cohérence temporelle des forces de turbulence synthétisées par cette méthode n'est pas conservée si la grille utilisée n'est pas suffisamment fine.

Yoon *et al.* (2009) ont poursuivi les travaux de Selle *et al.* (2005) en basant leur approche sur les particules de vorticité. Dans leur approche, les particules de vorticité sont ensemencées dans une grille suréchantillonnée de résolution supérieure (*up-scaled*). Cette méthode peut aussi être utilisée pour gérer les régions susceptibles aux forces de flottabilité (Kim *et al.*, 2012). Les particules de vorticité sont donc ajoutées où il y a possibilité d'émergence à la surface de portions de fluide. Weißmann et Pinkall (2010) ont proposé de simuler un fluide en utilisant des filaments de vorticité. De plus, leur formulation comprend un terme qui permet de gérer l'induction générée par des obstacles statiques. Cette approche n'est pas particulièrement efficace avec des scénarios où la turbulence est forte. La représentation par filament limite la méthode aux fluides à l'état gazeux.

La synthèse de détail ne permet pas à elle seule de reconstituer l'apparence d'une simulation en haute résolution à partir d'une simulation à basse résolution en entrée. Toutefois, les observations faites sur les propriétés du fluide à petite échelle dans ce type d'approche peuvent être réutilisées en complément des connaissances assimilées à l'aide d'un processus d'apprentissage.

L'idée des filaments a été reprise par Barnat et Pollard (2012) afin de fusionner des anneaux de vorticité à l'aide d'une structure en arbre. Finalement, l'utilisation des feuilles de vorticité a été proposée afin d'accélérer l'intégration de la vitesse par la méthode Biot-Savard (Brochu *et al.*, 2012). Cela permet aisément de traiter la vorticité à plusieurs échelles, et ce, sans l'utilisation d'une grille (donc pas nécessairement sous formulation eulérienne). Pfaff *et al.* (2012) ont aussi présenté une approche similaire.

1.2.3 Méthodes par simulation à petite échelle



Figure 1.4 Simulation de vaguelettes avec la méthode *iWave*.
Tirée de Kim *et al.* (2013).

Tout récemment, certains auteurs se sont intéressés à simuler des vaguelettes localement dans les hautes fréquences en surface d'une simulation en basse résolution. Ce type d'approche ne permet pas nécessairement de reproduire le comportement d'un fluide en haute résolution, mais permet d'augmenter considérablement le niveau de détails à petite échelle en surface de celui-ci. Pan *et al.* (2012) proposent d'utiliser une méthode discrète par vortex en deux dimensions afin de représenter et d'améliorer le mouvement des vagues en bordure d'un obstacle. Néanmoins, il est à noter qu'avec une telle approche les particules de vorticité ne sont ensemencées qu'aux emplacements où des vagues sont générées. Kim *et al.* (2013) proposent un modèle afin de découpler la dynamique du fluide en haute résolution de celle en basse résolution. Ils utilisent la technique du *iWave* (Tessendorf *et al.*, 2001) pour synthétiser des remous sous forme d'ondulation en haute résolution. La figure 1.4 illustre l'effet du *iWave* sur les hautes

fréquences à la surface du liquide. Tout récemment, Raveendran *et al.* (2014) ont proposé une méthode d’interpolation entre deux liquides à l’aide de marqueurs de contrôle. Leur approche de morphose est posée sous contrainte de conservation de volume.

Ces derniers travaux sont beaucoup plus proches des objectifs visés par notre approche. Particulièrement dans le sens où nous tentons d’obtenir une nouvelle simulation en haute résolution à partir d’exemples qui permettent d’établir une correspondance entre la simulation en basse résolution et celle en haute résolution. De plus, ces méthodes offrent des techniques complémentaires à notre approche. La combinaison de la surface résultante de notre approche avec des simulations de vaguelettes à petite échelle (ou avec la technique du *iWave*) permettrait d’obtenir un fluide détaillé beaucoup plus plausible à grande échelle qu’avec les approches antérieures.

1.3 Apprentissage automatique et traitement d’image

L’apprentissage automatique est de plus en plus utilisé en infographie. L’intérêt et l’enthousiasme sont suscités par plusieurs facteurs touchant ce domaine. Toutefois, la difficulté ou l’impossibilité d’être en mesure de modéliser des données est sans aucun doute le plus important de tous ces facteurs. De plus, la tentative de retrouver des données perdues ou tout simplement manquantes est aussi dans le haut de la liste. Plus particulièrement en infographie, ce type de méthode est utilisé en illumination, animation et avec des modèles complexes en trois dimensions. Une simulation de fluide est naturellement représentée par un modèle géométrique complexe (les particules et la surface du fluide). Néanmoins, la dynamique de son mouvement peut aussi faire l’objet d’analyses détaillées. D’ailleurs, les séparateurs à vaste marge ont été largement utilisés en animation pour étudier le mouvement des humains, animaux et objets du monde réel. Nous avons fait quelques expérimentations avec les SVM. Malheureusement, cette technique ne permet pas de conserver les hautes fréquences. Un bref survol des SVM est présenté à la section 1.3.1. Toutefois, l’apprentissage par dictionnaire est devenu de plus en plus utilisé pour reconstituer des modèles simples ou complexes à partir des propriétés d’un ensemble d’exemples. Cette dernière méthode offre une flexibilité intéressante aux objectifs

de notre approche. Quoi qu'il en soit, ces différentes méthodes d'apprentissage automatique seront brièvement présentées dans les sections qui suivent.

1.3.1 Séparateur à vaste marge

Les séparateurs à vaste marge, aussi appelés les machines à vecteur de support (Cortes et Vapnik, 1995), représentent un modèle d'apprentissage automatique qui utilise une méthode d'analyse des données et de détection de patrons. Cette méthode est utilisée pour la classification et l'analyse par régression. Cela permet de modéliser des frontières de décision. Les vecteurs de support sont en fait les valeurs proches de cette frontière. La distance entre un point i (ou une valeur) et l'hyperplan séparateur est exprimée comme ceci :

$$\frac{|\vec{w}^\top \vec{x}_i + w_0|}{\|\vec{w}\|} \quad (1.7)$$

Dans l'équation 1.7, \vec{w} est le vecteur unitaire et orthogonal à l'hyperplan (w_0 est la norme de ce vecteur). Cette séparation linéaire (équation 1.9) représente la fonction de décision du classifieur.

$$g(x|\vec{w}, w_0) = \vec{w}^\top \vec{x} + w_0 \quad (1.8)$$

$$= \sum_{j=1}^d w_j x_j + w_0. \quad (1.9)$$

Supposons que S est un ensemble $\{(x_1, u_1), \dots, (x_N, u_N)\}$ où $u_i \in \{-1, 1\}$. Donc la décision revient à faire cette assignation :

$$\vec{u}_i = \begin{cases} +1 & \text{si } g(x_i|\vec{w}, w_0) > 0 \\ -1 & \text{sinon} \end{cases} \quad (1.10)$$

Afin d'optimiser la séparation entre les données, une marge doit être maximisée (figure 1.5). Dans la figure 1.5, x_1 et x_2 sont les deux axes arbitraires de l'espace dans lequel sont les données. Les vecteurs de support permettent de définir la direction idéale du séparateur. Il s'agit

d'une minimisation de $\frac{\|\vec{w}\|^2}{2}$ sous contrainte où $\vec{u}_i(\vec{w}^\top \vec{x}_i + w_0)$ doit être plus grand que 1. De

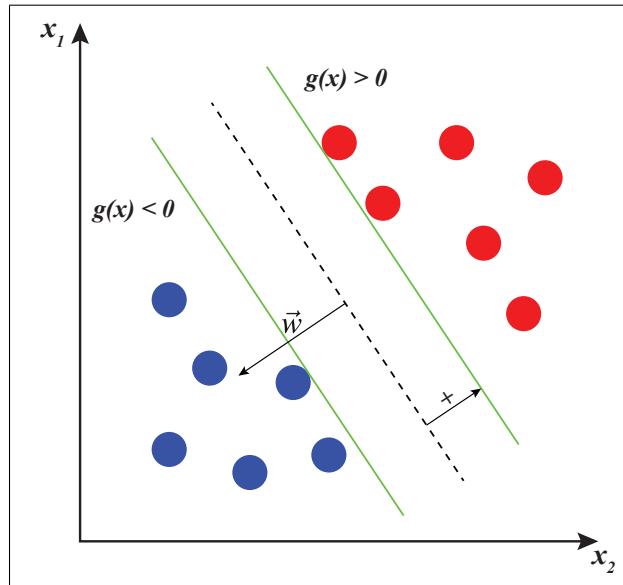


Figure 1.5 Maximisation de la marge à partir des vecteurs de support.

plus, des fonctions noyau peuvent être utilisées afin de transformer l'espace des données vers un espace où ces données sont linéairement séparables. Il existe plusieurs types de noyaux tel que polynomial, radial et sigmoïdal. Il est aussi possible d'utiliser des séparateurs à plusieurs classes. Ce type de séparateur est utilisé lorsque l'espace des données à un grand nombre de dimensions ou lorsque plusieurs de ces dimensions sont linéairement non séparables. L'analyse par composantes principales (Duda *et al.*, 2012) peut être aussi utilisée pour réduire la dimensionnalité d'un élément observé (par exemple, réduire le nombre d'attributs d'un élément) et par le fait même, réduire considérablement la complexité du modèle de régression utilisé par la méthode de séparateur à vaste marge. Nous utilisons l'analyse en composantes principales (PCA) à cette fin dans notre approche (section 2.2.2).

Finalement, la paramétrisation est une étape importante dans l'utilisation des SVM. L'optimisation est faite sur l'erreur (coût de la fonction) et la variance du noyau utilisé. Une recherche par grille est faite afin de couvrir l'espace des possibilités de permutation des deux paramètres

en question. Cette étape est appelée la validation croisée. Ces essais sont faits à partir de l'ensemble des données d'entraînement divisé en plusieurs sous-ensembles de taille égale. Une procédure détaillée permettant d'optimiser ces paramètres est disponible en ligne par les auteurs de la librairie LIBSVM (Chang et Lin, 2011).

1.3.2 Dictionnaire et représentation creuse

L'apprentissage par dictionnaire est une approche qui vise à apprendre un ensemble d'atomes. Par la suite, une combinaison linéaire de quelques-uns de ces atomes permet d'approximer un signal donné. L'apprentissage par dictionnaire n'est pas conçu pour la classification, par opposition aux SVM, mais plus particulièrement pour la reconstitution d'un signal. Ce type d'apprentissage est plus adapté à l'approche proposée dans ce document puisque notre objectif est de reconstituer une surface à partir d'exemples de simulations en haute résolution. Il est important de mentionner à ce stade qu'un signal peut être représenté par une structure de donnée complexe (*e.g.* une particule de fluide).

1.3.2.1 Codage parcimonieux

La parcimonie des données signifie que celles-ci comprennent un grand nombre de zéros sous forme vectorielle ou matricielle (matrice creuse). Le codage parcimonieux (*sparse coding*) est représenté par la modélisation des données sous forme d'un vecteur et basé sur des éléments simples provenant d'une combinaison linéaire parcimonieuse. Par exemple, un signal reconstitué à l'aide de cette approche, soit $x \approx D\alpha$, est une combinaison des atomes de D où α représente la pondération des éléments sélectionnés (figure 1.6). Cette notation est une approximation de la solution puisque celle-ci contient habituellement un bruit résiduel.

Dans la figure 1.6, les éléments d_i sont les atomes du dictionnaire et x est la parcelle reconstruite à l'aide de la combinaison linéaire des atomes d_2 , d_4 et d_5 . Les poids de cette combinaison sont

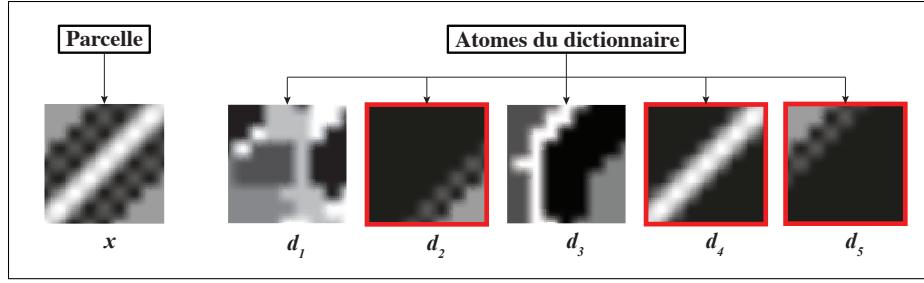


Figure 1.6 Représentation d'une parcelle par la combinaison de certains atomes du dictionnaire.

obtenus avec la minimisation de la fonction objective :

$$\frac{1}{2} \|\mathbf{x} - \mathbf{D}\alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (1.11)$$

Le premier terme de la fonction objective représente la similarité entre \mathbf{x} et $\mathbf{D}\alpha$. Cette similarité est estimée à partir de la norme l_2 . Naturellement, les atomes qui ne contribue pas à la reconstruction de \mathbf{x} ont un poids de valeur nulle (*e.g.* d_1 et d_3 de la figure 1.6). Ce type de codage permet d'apprendre certaines propriétés simples des données (dérivé de premier ordre et de second ordre) qui pourront être utilisées et ajoutées à titre d'élément du dictionnaire. Ces propriétés sont les descripteurs des atomes du dictionnaire. La fonction objective (équation 1.11) est utilisée pour extraire ces informations pour un nombre fixe d'échantillons dans les données traitées. Un terme de régularisation $\lambda \|\alpha\|_1$ est ajouté pour s'assurer que la représentation de chaque élément est parcimonieuse. Il s'agit d'un terme de pénalité basé sur la norme l_1 qui force la représentation parcimonieuse α à utiliser le moins d'atomes possible.

Cette fonction objective à pour but de minimiser la différence résiduelle entre l'élément à l'entrée et le résultat issu de la pondération des atomes du dictionnaire. Dans l'équation 1.11, \mathbf{x} représente le signal incomplet (image, maillage, etc.), \mathbf{D} est le dictionnaire (tous les atomes) et α le vecteur issu de la décomposition creuse des atomes du dictionnaire. L'appellation « signal incomplet » signifie que de l'information à ce signal est manquante. Enfin, le deuxième terme de la fonction objective $\lambda \|\alpha\|_1$ force la solution à être parcimonieuse. En d'autres mots, la solution par combinaison linéaire doit utiliser le moins d'atomes possible pour la reconstitution.

Pour résumer, l'objectif du codage parcimonieux est de représenter les vecteurs en entrée (*e.g.* pixels d'une image) par une combinaison linéaire pondérée d'une petite quantité de vecteurs simples qui représentent les caractéristiques de base des données.

Les éléments du dictionnaire sont communément appelés des atomes. L'apprentissage du dictionnaire peut être représenté sous forme d'une factorisation matricielle. La décomposition linéaire à l'aide de quelques atomes basés sur les ondelettes (Mallat, 1999) provenant d'un dictionnaire appris a permis d'obtenir d'excellents résultats en récupération d'information, par exemple, dans une texture (Peyré, 2009). De plus, malgré le fait que les SVM soient plus aptes à cette application, l'utilisation d'un dictionnaire a aussi permis d'effectuer des tâches complexes de classification (Bradley et Bagnell, 2008; Yang *et al.*, 2009).

Bien d'autres expérimentations ont été faites et ont prouvé que la décomposition linéaire est énormément adaptée aux applications de reconstruction de signaux (Chen *et al.*, 1998). L'utilisation d'un dictionnaire appris par comparaison à une base de données fixe a fait ses preuves dans la reconstruction d'un signal (Elad et Aharon, 2006). Certaines méthodes d'apprentissage par dictionnaire utilisent une approche itérative par lot (Aharon *et al.*, 2006; Lee *et al.*, 2006). Toutefois, ces approches peuvent s'avérer excessivement gourmandes en temps de calcul sur le CPU, car le traitement est fait à partir d'une fonction de coût sous contraintes qui doit prendre en compte l'ensemble des données d'entraînement, et ce, à chaque itération. Néanmoins, ces dernières approches ont prouvé obtenir de meilleurs temps de calcul qu'avec une descente de gradient du premier ordre (Aharon et Elad, 2008). Cependant, aucune de ces approches ne permet de traiter des ensembles d'entraînement de grande taille (Bousquet et Bottou, 2008). Dans ce cas, une approximation stochastique basée sur une descente du gradient de premier ordre est parfois utilisée pour réduire les coûts par itération (Kushner et Yin, 2003). Mairal *et al.* (2009a, 2010) proposent une approche d'apprentissage par dictionnaire formulée sous optimisation d'une fonction objective non convexe dans le but de minimiser le coût associé aux traitements d'un ensemble d'entraînement de grande taille. Le problème de minimisation est résolu à chaque itération à partir d'une fonction quadratique substitut sur un ensemble de contraintes.

1.3.3 Restauration d'image et reconstruction de surface

Dans cette section, nous introduisons un bref survol de l'état de l'art portant sur la restauration d'image et sur la reconstruction de surface par apprentissage automatique. La restauration d'image se réfère au problème d'estimer une image de référence à partir d'une version endommagée (information manquante), bruitée ou flouée. Ces applications couvrent un large éventail incluant notamment l'imagerie médicale et la recherche en aérospatiale. Il est bien connu que la restauration d'image est un problème inverse mal posé. En d'autres termes, il n'y a aucune garantie qu'une solution existe. Toutefois, la résolution d'un tel problème peut permettre d'obtenir de l'information sur l'image de référence. Ces informations a priori peuvent être utilisées pour reconstruire une version similaire à l'image de référence. Ce type d'approche est applicable à l'ensemble des méthodes de restauration de signal et est la plupart du temps assisté d'un processus d'apprentissage automatique. Tout récemment, des recherches ont été menées dans le but d'appliquer ces concepts de restauration à un maillage polygonal, et ce, à partir de connaissances acquises par apprentissage automatique avec un dictionnaire.

1.3.3.1 Débruitage et peinture rapportée

L'objectif des méthodes de débruitage (*image denoising*) est de récupérer l'image originale à partir de l'analyse de cette même image bruitée. L'analyse de cette image bruitée permet de mesurer un bruit ajouté sur l'image originale :

$$I_b = I_o + N \quad (1.12)$$

Dans l'équation ci-dessus, I_b est l'image bruitée observée, I_o est l'image originale et N est le bruit observé. On peut aussi dire que pour un pixel i de I_b , N_i est la mesure de perturbation faite sur celui-ci. Plusieurs méthodes ont été proposées pour éliminer cette perturbation (bruit) et retrouver l'image originale I_o . Malgré le fait que ces méthodes offrent des différences significatives, le débruitage d'une image est fait essentiellement par moyennage. Ce moyennage peut être fait localement par un modèle de lissage gaussien (Lindenbaum *et al.*, 1994), par un

filtre anisotrope (Catté *et al.*, 1992; Perona et Malik, 1990) ou par un filtre basé sur le voisinage (Yaroslavsky, 1985). D'autres travaux se sont concentrés sur le calcul et l'analyse des variations. Cette analyse peut notamment porter sur la minimisation de cette variation sur l'ensemble de l'image (Rudin *et al.*, 1992) ainsi que dans le domaine fréquentiel avec un filtre de Wiener (Yaroslavsky, 1985) ou par méthode de seuillage par ondelettes (Donoho, 1995).

D'une part, lorsqu'une image est fortement bruitée, les méthodes par moyennage ne sont pas adéquates. Dans ces situations, l'utilisation d'une méthode d'apprentissage automatique est nécessaire pour retrouver l'information qui a été perdue de l'image originale. Cet apprentissage permet d'extraire certaines propriétés des images utilisées à l'entraînement. Elad et Aharon (2006) ainsi que Mairal *et al.* (2009b) ont proposé d'utiliser une représentation creuse pour entraîner (à partir d'images non bruitées) un dictionnaire. Dans le même ordre d'idée, des images non bruitées peuvent servir à entraîner un champ aléatoire de Markov (Roth et Black, 2005). Zoran et Weiss (2011) ont présenté une méthode permettant d'entraîner un modèle fini par mélange gaussien. Finalement, Burger *et al.* (2012) ont étudié la relation entre des parcelles d'image bruitée et celles des exemples d'image non bruitée utilisés avec un perceptron multicouche.

D'une autre part, la méthode par peinture rapportée (*inpainting*) permet de reconstruire des portions d'image endommagée. Cette méthode est utilisée entre autres pour récupérer de l'information sur une image endommagée par la transmission en différé (*streaming*) et par l'occlusion causée par des couches supplémentaires à l'image originale. Hirani et Totsuka (1996) ont combiné le domaine spatial et fréquentiel pour remplir les régions endommagées de l'image. Plusieurs autres approches ont utilisé la synthèse de texture (Heeger et Bergen, 1995; Simoncelli et Portilla, 1998; Efros et Leung, 1999) pour remplir ces régions où l'information est manquante. Masnou et Morel (1998) ont proposé une alternative à ces approches leur permettant de joindre les points des isotopes (lignes de pixels à valeurs égales par canal) aux frontières des régions à retoucher à l'aide des courbes géodésiques.

Un nombre important d'approches ont été développées sur le concept de modélisation des parcelles cibles (inconnues) par des parcelles sources (connues) de l'image. Intuitivement, les régions à retoucher d'une image peuvent être basées sur les portions similaires à celle-ci (Jia et Tang, 2003). La composition peut aussi être utilisée pour générer un mélange de ces régions fragmentées dans le but de reconstruire l'image (Drori *et al.*, 2003). Bertalmio *et al.* (2003) ont aussi développé une technique qui combine une approche par diffusion (Bertalmio *et al.*, 2000) à la synthèse de texture (Efros et Leung, 1999). Par la suite, Wong et Orchard (2008) ont proposé une approche utilisant une fonction de similarité pondérée. Cette approche est très semblable à celle de Hung *et al.* (2008). Komodakis et Tziritas (2007) de leur côté ont utilisé une formulation de minimisation à l'aide d'une fonction objective permettant de déterminer le mélange optimal des sources pour reconstruire itérativement une région donnée. Plus récemment, Xu et Sun (2010) ont proposé une nouvelle approche basée sur la parcimonie de la représentation d'une parcelle de l'image à retoucher.

En somme, les méthodes de débruitage ne sont pas adéquates aux fins de notre approche. Les filtres utilisés dans ces méthodes ne font que dissiper les hautes fréquentes contenues dans la structure de données bruitée. Ces hautes fréquences doivent être conservées par notre approche afin de reproduire fidèlement les différences entre la surface de la simulation en haute résolution et de la surface de la simulation en basse résolution. Néanmoins, les méthodes par peinture rapportée ont un objectif en commun avec l'approche proposée dans ce document. Cet objectif est la reconstruction d'un signal. Ce signal, pouvant être aussi complexe qu'une simulation de fluide par particules, est incomplet et doit être reconstruit à partir d'exemples. Ces exemples représentent la base de connaissances en haute résolution qui rendent possible la reconstruction d'une surface détaillée. D'autant plus, les techniques récentes de l'état de l'art en peinture rapportée sont fréquemment utilisées conjointement avec une méthode d'apprentissage automatique.

1.3.3.2 Super-résolution

Dans la dernière décennie, plusieurs travaux portant sur la *super-résolution* ont été publiés. Notamment, Yang *et al.* (2010) ont proposé une telle méthode avec une formulation basée sur le codage parcimonieux.

La recherche statistique sur les images propose qu'une portion (parcelle) d'une image peut être représentée par une combinaison linéaire des atomes (vecteurs de base) d'un dictionnaire. La méthode par *super-résolution* s'inspire de cette observation et tente de déterminer une représentation parcimonieuse basée sur des exemples en haute résolution pour chaque parcelle d'une image en basse résolution. Les coefficients obtenus lors de la minimisation sont utilisés pour retrouver la parcelle en haute résolution à partir du dictionnaire associé. D'ailleurs, un apprentissage de deux dictionnaires (basse et haute résolution) en simultanée permet d'établir une correspondance de la représentation parcimonieuse entre les résolutions distinctes.

L'approche présentée dans ce document a été inspirée de la méthode par Yang *et al.* (2010). Néanmoins, par opposition à générer un codage parcimonieux similaire pour deux résolutions, nous utilisons cette relation pour établir une correspondance entre des propriétés géométriques et physiques d'une simulation de fluide.

1.3.3.3 Reconstruction de maillage polygonal

Tout récemment, l'utilisation d'un dictionnaire généré par apprentissage automatique a été effectuée afin d'optimiser la reconstruction d'un maillage polygonal (Xiong *et al.*, 2014). Dans cette approche la géométrie et la connectivité de reconstruction sont formulées dans un même problème d'optimisation. L'apprentissage par dictionnaire est posé sous contrainte afin de déterminer la position d'un sommet et la triangulation permettant de minimiser une fonction d'énergie composée de propriétés géométriques et d'un terme de régularisation de parcimonie de la solution. Les propriétés géométriques observées sont principalement la distance entre les sommets, la conservation des caractéristiques pointues et de l'estimation de la normale.

Enfin, cette méthode permet d'optimiser la géométrie et la connectivité d'un maillage en une seule phase. Ce traitement à l'apprentissage des différentes propriétés en simultanée par la fonction objective est intéressant. Cette formulation du problème d'optimisation est à considérer pour des travaux futurs. Ce travail de recherche a été réalisé par Xiong *et al.* parallèlement à notre projet.

1.4 Contributions

L'objectif de ce projet de recherche est de reproduire l'apparence de la surface en haute résolution à partir d'une simulation de résolution inférieure, et ce, en réduisant considérablement le temps de calcul nécessaire pour y arriver. Les contributions de ce projet de recherche présentées dans ce document sont les suivantes :

- Découplage de l'apparence à grande échelle de la surface du fluide avec la résolution de celui-ci.
- Procédure de préparation des simulations de fluide pour l'apprentissage par dictionnaire.
- Formulation des points de surface du fluide par un problème de minimisation sous contraintes géométriques et physiques.
- Réduction des temps de calcul nécessaires pour simuler un fluide en haute résolution.

CHAPITRE 2

APPRENTISSAGE PAR DICTIONNAIRES COUPLÉS

Dans ce chapitre, l'approche utilisée pour effectuer l'apprentissage des exemples d'entraînement dans le cadre de ce projet de recherche est détaillée. Il est important de noter que cette étape de l'approche est divisée en deux parties. Tout d'abord, la section 2.1 décrit les manipulations faites sur les données pour préparer la simulation de fluide en vue de l'apprentissage. Par la suite, la section 2.2 décrit la méthode utilisée pour faire l'apprentissage d'un dictionnaire à partir des propriétés géométriques et physiques d'un fluide.

La préparation des données d'entraînement est une phase importante de l'approche. La négligence de certaines des étapes de cette préparation peut inférer un taux d'erreur important dans les étapes subséquentes. Particulièrement dans ce contexte, la sélection des exemples d'entraînement a beaucoup d'influence sur la qualité du dictionnaire résultant. L'approche présentée dans ce mémoire se base sur une simulation de type SPH. Bien que notre choix s'est arrêté sur ce type de simulation par particules, d'autres types de simulation lagrangienne, tels que *position-based fluid* ou FLIP, auraient pu être tout autant adéquats. La figure ci-dessous présente un bref survol des étapes nécessaires à la génération d'un dictionnaire pour une propriété donnée.

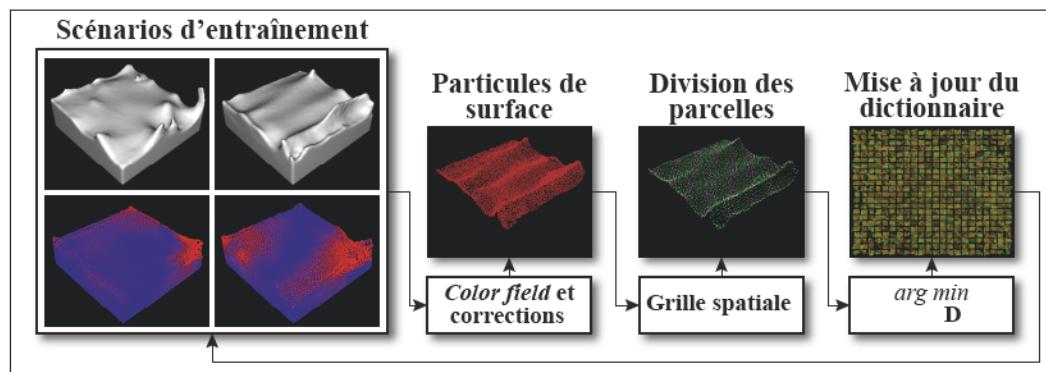


Figure 2.1 Survol des étapes nécessaires pour générer le dictionnaire d'une propriété.

2.1 Préparation des données d'entraînement

Il est important de mentionner à ce stade que toutes simulations utilisées pour générer un dictionnaire sont des variantes d'un même scénario (figure 2.2). Par exemple, afin de reconstruire adéquatement la simulation d'une goutte tombant dans un réservoir, les exemples d'entraînement doivent provenir de scénarios similaires avec variances. Une variance peut être simplement de modifier aléatoirement la position initiale de la goutte avant sa descente.

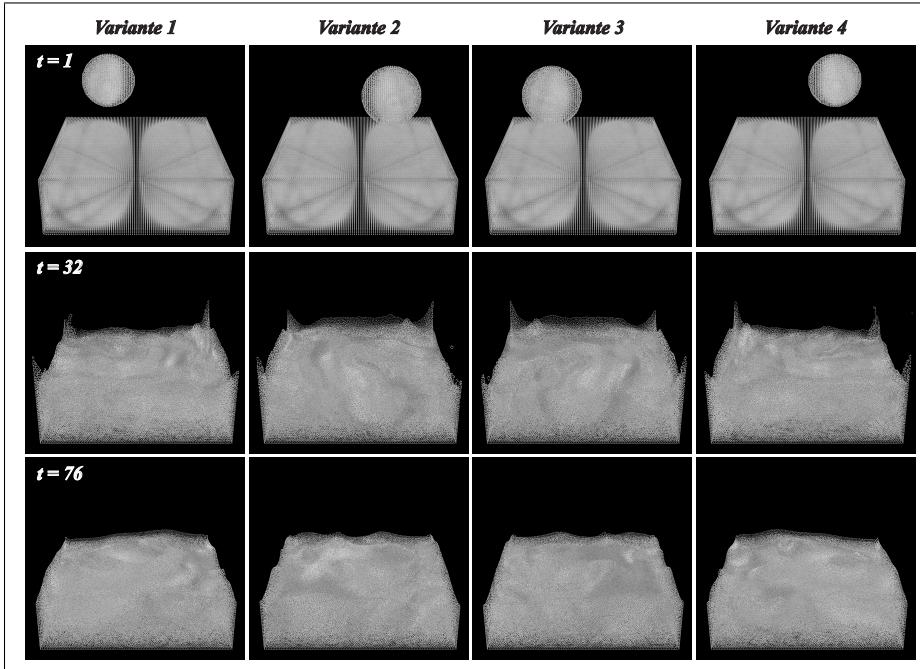


Figure 2.2 Exemple de variantes utilisées à l'entraînement du scénario de la goutte qui tombe dans un réservoir.

2.1.1 Identification et extraction des particules de surface

Notre approche ne s'intéresse qu'au niveau de détail en surface d'un fluide. En infographie, dans la majorité des cas avec ce type d'approche, l'intérêt est d'améliorer l'apparence de l'aspect visible de la simulation, c'est-à-dire la surface. De plus, il est beaucoup plus simple d'analyser une parcelle en surface d'un fluide qu'un volume de fluide.

Méthode avec *color field*

La modélisation des forces de tension de la surface peut être faite à l'aide de la méthode de Morris (2000). À partir de cette formulation, il est possible de suivre l'interface (entre l'air et le liquide) d'une simulation par SPH en simulant l'advection à l'aide d'une fonction de convolution avec un noyau d'ordre élevé. D'ailleurs, les particules d'un fluide sont influencées par des forces d'attraction avec leurs voisins. Sous la surface, ces forces d'attraction sont égales dans toutes les directions. Toutefois, ces forces ne sont pas balancées pour les particules à la surface libre du fluide. La force résultante de ces particules de surface tend dans la direction de la normale du fluide. De plus, la tension de la surface dépend des fluides qui forment la surface. Un coefficient θ est défini pour chaque fluide.

Un champ additionnel est utilisé pour identifier les particules de surface. Ce champ est appelé *color field* dans la littérature. La formulation utilisée avec les SPH est :

$$c_i \approx \sum_j \frac{m_j}{\rho_j} W_j(\vec{r}_{ij}) \quad (2.1)$$

Le *smoothed color field (SCF)* est l'appellation de champ lorsqu'il est utilisé avec les SPH (Müller *et al.*, 2003). Dans l'équation 2.1, $W_j(\vec{r}_{ij})$ est une fonction noyau où \vec{r}_{ij} est le vecteur entre la position de la particule i est celle de la particule j . Le gradient de ce champ permet d'obtenir une estimation de la normale à la surface. Le vecteur normal résultant pointe à l'intérieur du fluide.

$$\vec{n} \approx \nabla c_i \approx \sum_j \frac{m_j}{\rho_j} \nabla W_j(\vec{r}_{ij}) \quad (2.2)$$

Ensuite, les particules de surface p_i^S sont identifiées lorsque la magnitude de ∇c_i est supérieure à un certain seuil s_c . Ce seuil est déterminé en fonction de la distance de séparation des particules (résolution) du fluide de la simulation. La magnitude de la normale \vec{n} des particules à la surface libre sera nécessairement plus grande comparativement aux particules submergées qui subissent d'importantes forces d'attraction. De plus, un autre seuil s_f est utilisé pour éliminer les particules de surface qui sont aux frontières de la simulation. Le seuil s_f est calculé à partir

d'une frontière f du domaine de simulation et d'une constante normalisée (e.g. $s_{f_x} = \pm 0.9 f_x$).

Ce seuil est évalué sur toutes les composantes de la position des particules. Par exemple, si une particule est proche d'une frontière du domaine de simulation, celle-ci ne sera pas conservée. De cette façon, uniquement les particules à la surface libre du fluide sont conservées.

D'ailleurs, les particules de surface p^S est un sous-ensemble des particules p .

$$p_i^S = \begin{cases} 1 & \text{si } |\nabla c_i| \geq s_c \\ 0 & \text{sinon} \end{cases} \quad (2.3)$$

Cette méthode d'identification des particules de surface n'est pas idéale avec les simulations à basse résolution. Si l'espacement entre les particules est trop grand, l'estimation de la normale de la surface risque de prendre en considération des particules qui sont à l'intérieur du fluide. De plus, lorsque le seuil s_c est poussé pour contourner ce problème, il est possible de perdre certaines des particules de surface telle qu'illustrée à la figure 2.3. Un nombre plus élevé de particules permet de partiellement résoudre ce problème.

L'erreur est notamment due au fait que la normale est estimée à partir du gradient. Il est possible de corriger cette erreur en calculant le vecteur normal à l'aide d'une sommation pondérée du gradient de ce champ des voisins j à l'interface :

$$\vec{n}_i = \frac{\sum_j \nabla c_j A_j}{|\sum_j \nabla c_j A_j|} \quad (2.4)$$

Dans l'équation 2.4, A_j est une fonction surface-delta. Cette fonction détermine une pondération entre la distance de la particule voisine avec celle courante et le rayon d'influence de la particule courante. Tout récemment, Orthmann *et al.* (2013) ont aussi proposé une solution semblable à Adami *et al.* (2010) présentée à l'équation 2.4 afin de mieux gérer les fines feuilles de liquide. Nous n'avons pas intégré ces derniers ajustements à notre implémentation. Néanmoins, pour améliorer l'efficacité du *color field*, nous avons utilisé une grille spatiale fixe afin d'identifier les cellules de surface au fluide. La taille et la résolution de la grille sont déterminées à partir de la résolution de la simulation.

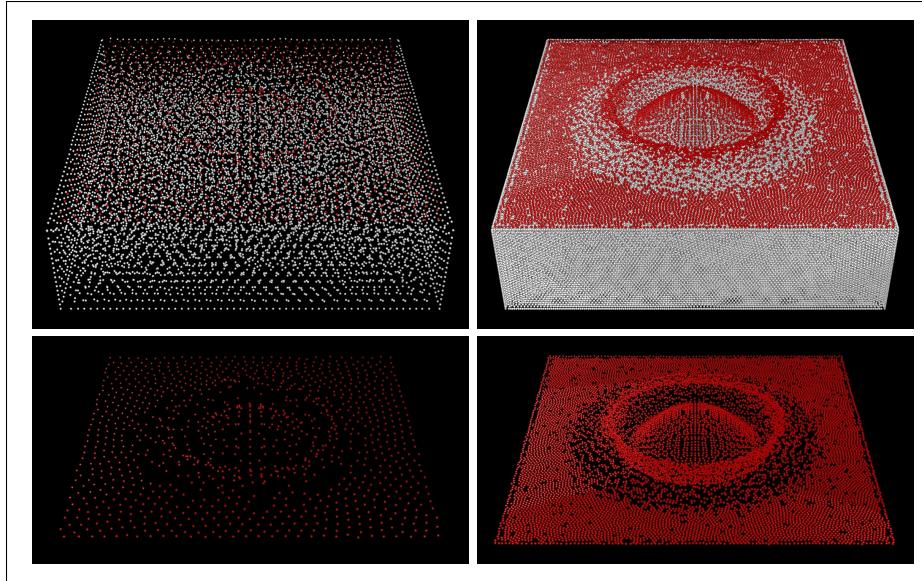


Figure 2.3 Identification des particules de surface (en rouge) à l'aide de la méthode du *color field*. La simulation à gauche contient environ 10 000 particules et celle droite plus de 300 000 particules.

Pour éviter une mauvaise évaluation des particules de surface, nous avons préféré utiliser des volumes englobants pour chaque particule identifiée par la méthode du *SCF*. Le volume englobant utilisé est un cylindre. Pour déterminer si une particule est à l'interface du fluide, on doit déterminer s'il existe une autre particule au-dessus de celle-ci. Pour ce faire, on doit valider l'existence de particules j (voisines) à l'intérieur d'un volume placé au-dessus d'une particule courante i et orienté sur le vecteur normal à la particule.

La normale utilisée est celle de l'approximation obtenue par la méthode du *SCF*. Si une particule j existe dans ce volume, la particule i n'est pas une particule de surface. Le plan normal à la surface est nécessairement au sommet de la particule (ligne pointillée de la figure 2.4). Cette méthode est complémentaire à l'approche du *SCF*. Les équations d'inégalité 2.6, 2.7 et 2.7

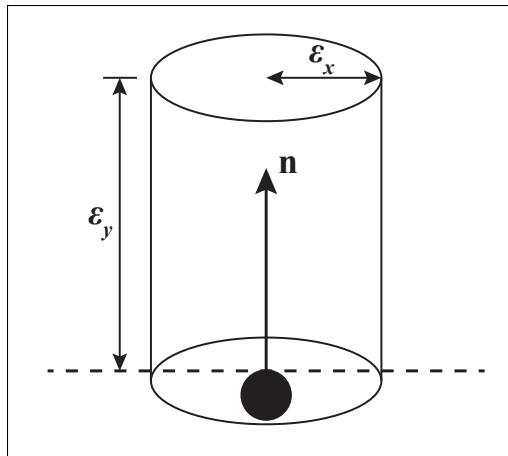


Figure 2.4 Volume englobant (cylindre) utilisé pour identifier une particule de surface.

présentent les bornes imposées par le volume englobant :

$$(p_{i_x} - \epsilon_x) < p_{j_x} < (p_{i_x} + \epsilon_x) \quad (2.5)$$

$$(p_{i_z} - \epsilon_z) < p_{j_z} < (p_{i_z} + \epsilon_z) \quad (2.6)$$

$$p_{j_y} > p_{i_y} \quad (2.7)$$

Les inégalités décrites aux équations ci-dessus doivent toutes être fausses sans exception, et ce, pour toutes les particules i , afin qu'une particule j puisse être considérée comme une particule de surface. Les paramètres ϵ_y et ϵ_x représentent respectivement la hauteur maximum de validation entre deux particules et la marge d'espacement maximum entre celles-ci. La valeur de ϵ_x est toujours égal à la valeur de ϵ_z puisqu'il s'agit du rayon d'un cylindre.

2.1.2 Sélection des parcelles de fluide

La sélection des parcelles de fluide pour l'entraînement est une étape importante. L'idée est de ne pas mettre à jour le dictionnaire avec des parcelles qui ne présentent aucune propriété intéressante de la simulation.

Tout d'abord, les scénarios utilisés pour l'entraînement doivent être sélectionnés manuellement. Le choix des scénarios est basé sur la complexité de comportement du fluide. D'ailleurs, l'approche proposée ne permet pas de traiter la surface des scénarios qui comportent des éclaboussures et des détachements d'un volume de fluide (*e.g.* des gouttelettes). La figure 5.7 du chapitre 5 présente un exemple de scénario actuellement impossible à reconstruire avec notre approche. De plus, notre but n'est pas de généraliser un dictionnaire. Dans notre contexte, une généralisation serait de tenter d'apprendre tous les scénarios possibles à partir d'un seul dictionnaire. Un surentraînement aurait pour effet de diluer les exemples du dictionnaire et ainsi perdre de l'information sur les hautes fréquences de ceux-ci. En limitant l'apprentissage à une sélection donnée, nous garantissons la qualité des éléments appris.

Lors du traitement d'un scénario, les causes de rejet d'une parcelle sont notamment en lien avec l'absence de courbure et l'impossibilité de correspondance entre les propriétés physiques des deux résolutions distinctes. Le détail des calculs de ces propriétés est présenté aux sections 2.1.4.1 et 2.1.4.2.

Afin d'accélérer le processus de sélection, une moyenne par parcelle de fluide est calculée pour chaque propriété. Par exemple, pour un nombre N d'échantillons par parcelle, une moyenne de la propriété p est calculée et évaluée au moment du processus de sélection :

$$\bar{p} = \frac{\sum_i^N p_i}{N} \quad (2.8)$$

Dans certains cas, l'estimation par la moyenne n'est pas représentative de la situation. Si la résolution de la simulation de fluide est trop faible (peu de particules), le nombre de particules par parcelle sera insuffisant pour obtenir une estimation adéquate par propriété. Par exemple, cette estimation permet de ne pas considérer les parcelles qui ont un trop grand écart d'énergie entre les différentes résolutions. La vorticité est aussi utilisée comme discriminant en complément de l'énergie cinétique.

Nous avons choisi d'évaluer l'énergie cinétique par parcelle puisque celle-ci offre la possibilité de retrouver l'information en basse résolution à partir d'exemples en haute résolution

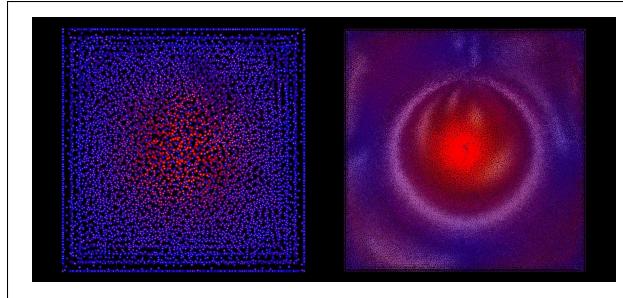


Figure 2.5 Instant dans la simulation où les valeurs d'énergie cinétique ne sont pas équivalentes entre les différentes résolutions.

(figure 2.5). Il est également possible d'effectuer cette relation avec la vorticité puisque celle-ci est aussi calculée à partir de la vitesse. Toutefois, l'inverse n'est pas toujours réalisable. Certaines simulations en basse résolution ne conservent pas assez d'information sur l'énergie pour établir une relation avec l'énergie en haute résolution (figure 2.6).

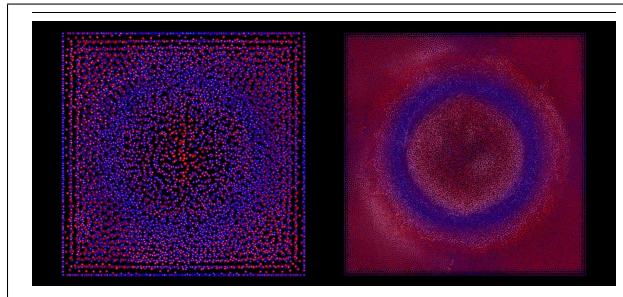


Figure 2.6 Instant dans la simulation où l'énergie cinétique est similaire entre les différentes résolutions.

La sélection tient aussi le rôle de modérateur sur la taille du dictionnaire. Puisque le processus de sélection restreint le nombre de parcelles retenues, le dictionnaire peut être limité à une certaine taille prédéterminée. De plus, le nombre de parcelles retenues réduit à mesure que le nombre d'itérations de mise à jour du dictionnaire (par trame) augmente. Il est aussi important de rappeler qu'un dictionnaire ne couvre pas l'ensemble des scénarios possibles. Dans nos expérimentations, un dictionnaire est généré par type de scénario (*e.g.* l'effet d'une goutte qui tombe dans un réservoir).

2.1.3 Normalisation des parcelles

La normalisation (intervalle entre 0 et 1) des données doit être effectuée sur l'ensemble des hauteurs (composante y de la position d'une particule). De plus, afin de conserver une cohérence des données, cette normalisation est faite sur l'ensemble des trames, et ce, pour les deux résolutions utilisées. Les trames sélectionnées peuvent provenir de plusieurs scénarios. La normalisation y_n est faite à l'aide d'une valeur minimum y_{min} et d'une valeur maximum y_{max} :

$$y_n = \frac{y - y_{min}}{y_{max} - y_{min}} \quad (2.9)$$

La valeur minimum et la valeur maximum sont calculées lors de la phase d'identification des particules de surface. De cette façon, la performance de l'approche n'est pas affectée par cette étape. Ces valeurs sont conservées avec le dictionnaire afin d'être réutilisées pour récupérer les valeurs originales lors de l'étape de la reconstruction :

$$y = y_n(y_{max} - y_{min}) + y_{min} \quad (2.10)$$

2.1.4 Transfert des propriétés

Ce transfert des propriétés se fait en deux temps. Tout d'abord, la relation que nous proposons dans ce document est basée sur la théorie de corrélation entre la courbure et la vorticité en surface d'un fluide (Longuet-Higgins, 1998). Cette relation est faite à partir de la théorie des flux turbulents en deux dimensions (Longuet-Higgins, 1953) :

$$\omega = -2kq \quad (2.11)$$

Cette simple relation entre la vorticité de surface ω , la vitesse tangentielle q et la courbure principale k , fait l'hypothèse qu'il n'y a pas de séparation du fluide (*e.g.* pas d'éclaboussures). En d'autres mots, on assume que le fluide est stable. Longuet-Higgins a étendu l'approche aux écoulements en trois dimensions (Longuet-Higgins, 1998). Par la suite, des termes de vorticité

ω_{\parallel} et ω_{\perp} peuvent être calculés. La preuve de cette relation est très bien détaillée dans l'article de Longuet-Higgins. À partir de particules échantillonées d'une simulation turbulente (points rouges de la figure 2.7), il nous a été possible de mettre en évidence la relation entre la courbure de la surface et la vorticité des particules. Notre estimation de la tendance a été représentée à partir d'un polynôme d'ordre supérieur basé sur 500 échantillons sélectionnés à la surface du fluide.

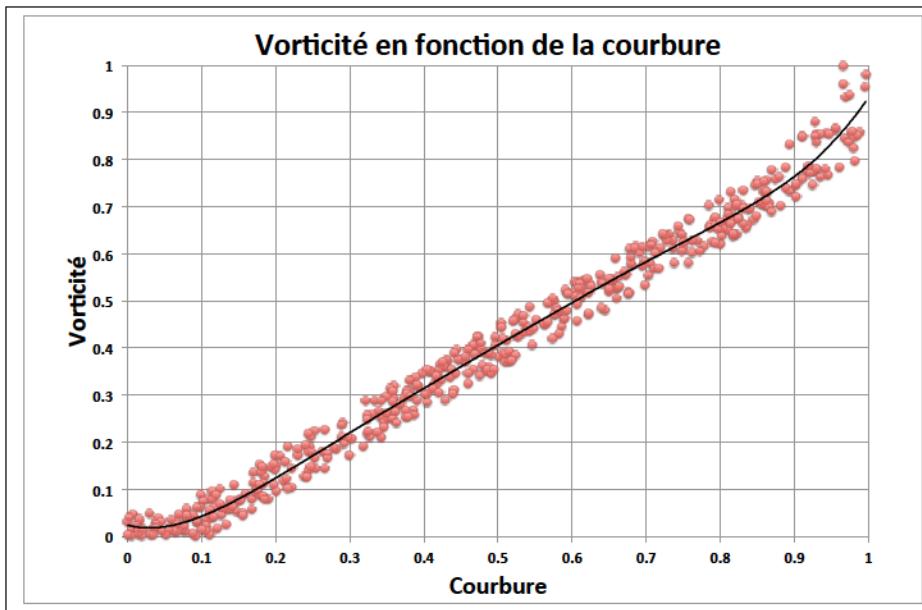


Figure 2.7 Corrélation entre la courbure de la surface et la vorticité d'une particule.

Dans un second temps, l'association des propriétés géométriques de surface aux particules est faite à partir d'une approche simple basée sur la distance la plus courte. Il s'agit de la distance entre une particule et un polygone. Pour chaque particule, une distance est calculée entre celle-ci et tous les polygones d'une surface approximative. La distance la plus courte détermine le polygone associé avec la particule. Les propriétés de surface peuvent maintenant être transférées à la particule.

Les propriétés de surface transférées sont les suivantes :

- Normale
- Intensité de courbure (*curvedness*)
- Directions principales de courbure
- Type de courbure (*shape index*)
- Rapport de sphéricité (*sphericity ratio*)

Le calcul de ces propriétés est détaillé à la section suivante.

2.1.4.1 Propriétés géométriques de la surface

Les propriétés géométriques de la surface sont principalement associées à la courbure. La courbure principale calculée à cette étape est basée sur les travaux de Theisel *et al.* (2004). Dans un premier temps, nous devons calculer la matrice de Weingarten. Pour construire cette matrice, nous devons tout d'abord calculer les dérivées partielles \vec{x}_u , \vec{x}_v , \vec{n}_u , et \vec{n}_v obtenues à partir de la position \mathbf{X} et de la normale \mathbf{N} des sommets formant le triangle. Ces sommets sont en fait trois particules de surface avoisinantes et les normales proviennent de l'estimation faite à l'étape d'identification des particules de surface.

Posons

$$\mathbf{X} = \{\vec{x}_0, \vec{x}_1, \vec{x}_2\} \quad (2.12)$$

$$\mathbf{N} = \{\vec{n}_0, \vec{n}_1, \vec{n}_2\}. \quad (2.13)$$

Les dérivés partielles sont calculées à partir des vecteurs \vec{x}_i et \vec{n}_i .

$$\vec{x}_u = x_1 - x_0 \quad (2.14)$$

$$\vec{x}_v = x_2 - x_0 \quad (2.15)$$

$$\vec{n}_u = n_1 - n_0 \quad (2.16)$$

$$\vec{n}_v = n_2 - n_0 \quad (2.17)$$

Avec ces dérivées partielles, nous pouvons maintenant calculer les termes qui serviront à construire la matrice Weingarten.

$$\vec{E} = \vec{x}_u^2 \quad (2.18)$$

$$\vec{F} = \vec{x}_u \vec{x}_v \quad (2.19)$$

$$\vec{G} = \vec{x}_v^2 \quad (2.20)$$

$$\vec{Q} = -\vec{n}_u \vec{x}_u \quad (2.21)$$

$$\vec{L} = -\vec{n}_u \vec{x}_u \quad (2.22)$$

$$\vec{M}_1 = -\vec{n}_v \vec{x}_v \quad (2.23)$$

$$\vec{M}_2 = -\vec{n}_v \vec{x}_v \quad (2.24)$$

Les vecteurs $\vec{E}, \vec{F}, \vec{G}, \vec{Q}, \vec{L}, \vec{M}_1$ et \vec{M}_2 sont utilisés pour construire la matrice Weingarten :

$$\mathbf{W} = \begin{bmatrix} \vec{L} \cdot \vec{G} - \vec{M}_1 \cdot \vec{F} & \vec{M}_2 \cdot \vec{G} - \vec{Q} \cdot \vec{F} \\ \vec{E} \cdot \vec{G} - \vec{F}^2 & \vec{E} \cdot \vec{G} - \vec{F}^2 \\ \vec{M}_1 \cdot \vec{E} - \vec{L} \cdot \vec{F} & \vec{Q} \cdot \vec{E} - \vec{M}_2 \cdot \vec{F} \\ \vec{E} \cdot \vec{G} - \vec{F}^2 & \vec{E} \cdot \vec{G} - \vec{F}^2 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{21} \\ w_{12} & w_{22} \end{bmatrix} \quad (2.25)$$

w_{11} et w_{22} sont respectivement les valeurs propres k_1 et k_2 et leurs vecteurs propres correspondants sont :

$$\vec{w}_1 = \begin{bmatrix} w_{11} \\ w_{12} \end{bmatrix} \quad (2.26)$$

$$\vec{w}_2 = \begin{bmatrix} w_{21} \\ w_{22} \end{bmatrix} \quad (2.27)$$

Finalement, ces termes nous permettent de définir la courbure gaussienne K , la courbure moyenne H et les deux directions principales de courbure \vec{k}_1 et \vec{k}_2 .

$$K = k_1 k_2 \quad (2.28)$$

$$H = \frac{1}{2}(k_1 + k_2) \quad (2.29)$$

$$\vec{k}_1 = w_{11}\vec{x}_u + w_{12}\vec{x}_v \quad (2.30)$$

$$\vec{k}_2 = w_{21}\vec{x}_u + w_{22}\vec{x}_v \quad (2.31)$$

De plus, à partir de ces valeurs de courbure (k_1 et k_2), il est possible de calculer d'autres métriques pour définir la surface, et ce, localement. Ces autres métriques sont l'intensité de courbure (**CV**), le type de courbure (**SI**) et le rapport de sphéricité (**SR**). Ce type de métrique est utilisé notamment en imagerie médicale (Chen *et al.*, 2007) pour définir la morphologie des artères et des intestins. Voici comment ces métriques sont calculées à partir des valeurs de courbure principale.

$$\mathbf{CV} = \sqrt{\left(\frac{k_1^2 + k_2^2}{2}\right)} \quad (2.32)$$

$$\mathbf{SI} = \frac{1}{2} - \frac{1}{\pi} \arctan\left(\frac{k_1 + k_2}{k_1 - k_2}\right) \quad (2.33)$$

$$\mathbf{SR} = \left| \frac{k_1 - k_2}{H} \right| \quad (2.34)$$

Le terme **SI** représente la famille de courbure. D'ailleurs, il existe cinq formes géométriques simples qui peuvent caractériser une surface. La valeur de **SI** est normalisée (entre 0 et 1) et l'intervalle est divisé également en cinq (tranche de 0.2). Finalement, chaque tranche représente une des cinq familles.

2.1.4.2 Propriétés physiques du fluide

Les propriétés physiques des particules utilisées dans notre approche proviennent de la simulation en haute résolution. Ces propriétés sont calculées pendant les itérations du simulateur

SPH. Actuellement, les propriétés physiques qui ont été considérées sont la vitesse, l'énergie cinétique et la vorticité. La vitesse est calculée à partir des équations de mouvement du fluide. Les calculs de l'énergie cinétique et de la vorticité ont été ajoutés comme étape supplémentaire à la simulation. Le détail des calculs pour obtenir ces forces est présenté dans cette section.

Tout d'abord, la vitesse doit être calculée puisque les autres propriétés en dépendent. La vitesse d'une particule est obtenue à partir de l'équation d'accélération. Cette équation est composée des forces qui agissent sur les particules de la simulation. L'accélération pour une particule i est calculée comme ceci :

$$\vec{a}_i = \rho_i \frac{\delta \vec{v}}{\delta t} = \vec{f}_i^p + \vec{f}_i^v + \vec{f}_i^e \quad (2.35)$$

\vec{f}_i^p , \vec{f}_i^v et \vec{f}_i^e sont respectivement les forces de pression (Müller *et al.*, 2003), de viscosité (Müller *et al.*, 2005) et externes. Dans nos simulations, la force externe est exclusivement la force gravitationnelle. Avec une simulation SPH, ces forces sont calculées comme suit :

$$\vec{f}_i^p = - \sum_j m_j \frac{p_i + p_j}{2\rho_j} \nabla W_{ij} \quad (2.36)$$

$$\vec{f}_i^v = \sum_j \frac{\mu_i + \mu_j}{2} \frac{m_j}{\rho_j} (\vec{v}_j - \vec{v}_i) \nabla^2 W_{ij} \quad (2.37)$$

$$\vec{f}_i^e = \rho_i \vec{g} \quad (2.38)$$

où

$$p_i = k_p(\rho_i - \rho_0) \quad (2.39)$$

$$\rho_i = \sum_j m_j W_{ij} \quad (2.40)$$

Dans l'équation 2.40, ρ_i est la masse volumique de la particule i (équation 2.40) et ρ_0 sa masse volumique au repos. De plus, k_p est une constante définie par l'utilisateur qui permet de contrôler l'influence de la force de pression sur la simulation. Plus cette valeur est élevée et moins le fluide est compressible.

Ensuite, à partir d'une étape supplémentaire dans le simulateur, il est possible d'estimer l'énergie cinétique du fluide.

$$E_k = \frac{1}{2}mv^2 \quad (2.41)$$

Et voici comment l'énergie cinétique d'une particule i est calculée sous la formulation SPH :

$$e_k^i = \frac{1}{2} \sum_j m_j v_i^2 W_{ij} \quad (2.42)$$

Nous pouvons aussi calculer le spectre de cette énergie par parcelle en surface du fluide. D'ailleurs, en décomposant les hautes fréquences d'énergie, nous pouvons échelonner l'énergie et ainsi suivre la théorie des deux tiers de Kolmogorov (Pope, 2000; Davidson, 2004) :

$$E(k) \sim \varepsilon_0^{\frac{2}{3}} k^{-\frac{5}{3}}. \quad (2.43)$$

Dans l'équation 2.43, la dissipation de l'énergie par cascade sur les différentes échelles sont respectivement représentées par ε_0 et k (échelles inertielles par longueur d'onde). En d'autres termes, la dissipation de l'énergie dans les échelles inférieures du fluide. Toutefois, pour la décomposition nous avons utilisé la méthode discrète de la transformée en ondelettes (Kim *et al.*, 2008b) par comparaison à la transformée de Fourier utilisé par Pope (2000). Afin de déterminer les différentes octaves d'énergie, une décomposition du champ de vitesse de la simulation doit être faite :

$$\hat{u}_i^k = \frac{1}{k\sqrt{s}\Psi_{sum}} \sum_j u_j \Psi \quad (2.44)$$

Dans l'équation 2.44, Ψ est la fonction ondelette (fonction mère) et Ψ_{sum} est la sommation $\sum_j \Psi$ pour toutes les particules j d'une parcelle de la particule i . Le terme u_j représente la vitesse d'une particule avoisinante j . La fonction ondelette utilisée est celle de Ryan (1994), aussi appelée la fonction du chapeau mexicain. La constante k est le niveau de l'échelle du spectre d'énergie. La sommation des échelles de ce spectre d'énergie est déterminée comme suit :

$$\hat{u}_i = \sum_k \hat{u}_i^k \quad (2.45)$$

La solution obtenue par l'équation ci-dessus nous permet d'avoir une énergie cinétique par octave suivant la pente $(-5/3)$ d'énergie par échelle déterminée par Kolmogorov. Tel que présenté à la figure 2.8, le spectre d'énergie observé (ligne rouge semi-transparente) dans nos simulations est similaire au spectre théorique (ligne verte) de la cascade d'énergie de Kolmogorov (Pope, 2000). De plus, nous obtenons une pente dont l'estimation est proche de la valeur théorique $-5/3$.

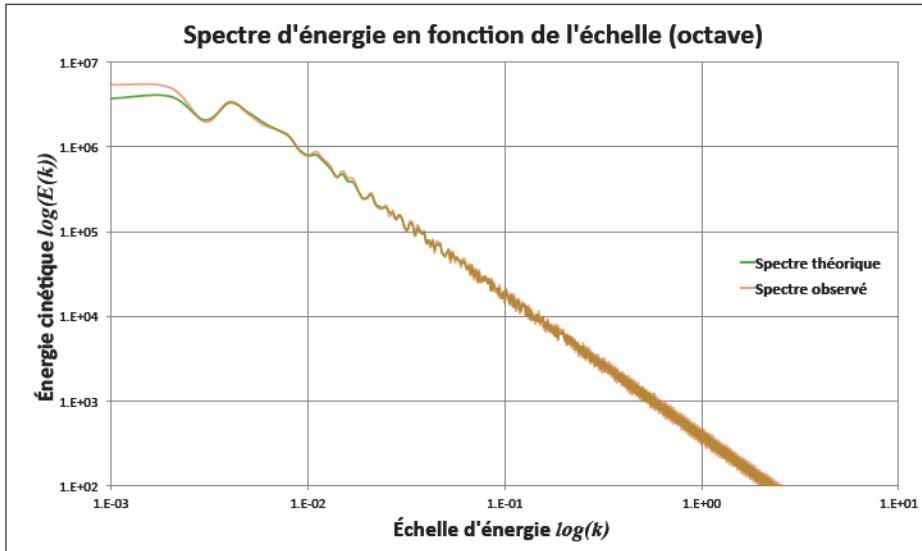


Figure 2.8 Comparaison du spectre d'énergie théorique avec le spectre d'énergie observé dans l'une des simulations utilisées à la phase d'apprentissage.

Finalement, la dernière propriété à calculer est la force de vorticité confinée. Cette force est la principale cause des remous en surface du fluide. Tout d'abord, le calcul de la vorticité pour une particule i à partir de la vitesse de celle-ci est décrit comme suit :

$$\omega_i = \nabla \times \vec{u}_i = \sum_j m_j \vec{u}_{ij} \times \nabla W_{ij} \quad (2.46)$$

La force confinée de la vorticité est donc exprimée comme ceci :

$$f_i^\omega = \varepsilon (\vec{N} \times \omega_i) \quad (2.47)$$

où ϵ est une constante définie par l'utilisateur pour contrôler le degré de vorticité de la simulation. \vec{N} est un vecteur normalisé pointant dans la direction du centre de rotation.

$$\vec{N} = \frac{\eta}{|\eta|} \quad (2.48)$$

et où η sous formulation SPH est égale à :

$$\eta = \sum_j \frac{m_j}{\rho_j} |\omega_j| \nabla W_{ij}. \quad (2.49)$$

Nos calculs de la force de vorticité sont inspirés de l'approche utilisée avec les *Position-based particles* (Macklin et Müller, 2013).

2.1.5 Modélisation des parcelles

La modélisation des parcelles est l'étape où le domaine de la simulation est divisé en portions égales. Ces portions de fluide sont en fait un ensemble de particules à la surface de celui-ci. La division est ce qui dicte le niveau de détail des parcelles. Plus le domaine de simulation est subdivisé et moins une parcelle contiendra de particules. D'ailleurs, il s'agit d'une limitation concernant la résolution de la simulation. La résolution d'une parcelle est fortement dépendante du nombre de particules en surface du fluide. La qualité d'extraction des particules de surface peut aussi être en cause.

La séparation du domaine de simulation à la surface est effectuée à l'aide d'une grille spatiale. La grille est divisée seulement sur deux axes, c'est-à-dire l'axe des x et l'axe des z (y étant l'axe vers le haut). En d'autres termes, la division est faite en deux dimensions sur le plan xz .

Outre la méthode d'extraction des particules de surface, cette technique de séparation du domaine nous limite à l'apprentissage de simulations stables avec un niveau de complexité faible à modéré. De plus, afin de ne pas inférer d'erreur à l'apprentissage, les parcelles de bordure ne sont pas considérées. L'explication est que la plupart des anomalies dans une simulation de fluide surviennent aux frontières de celle-ci.

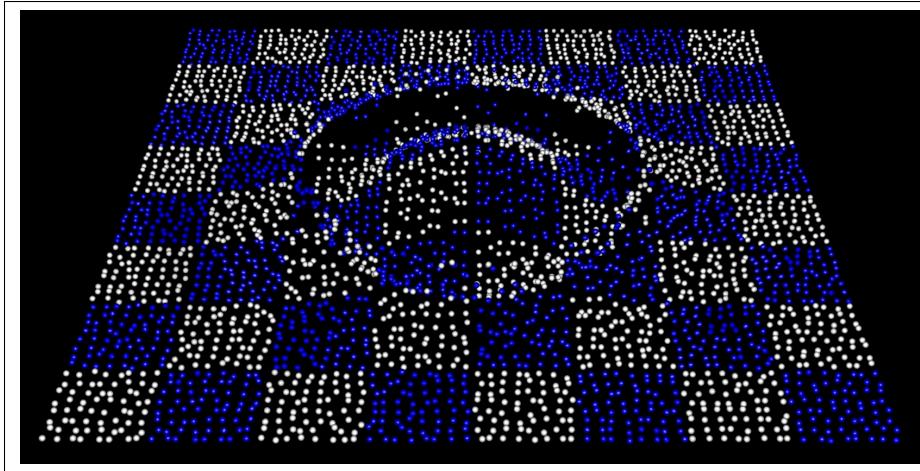


Figure 2.9 Division des parcelles de particules de surface à partir de la grille spatiale.

Une fois la division faite, des particules par cellule doivent être sélectionnées. La distribution des particules sélectionnées par parcelle est importante. Si la répartition des particules d'une parcelle n'est pas uniforme, les caractéristiques géométriques et physiques de cette dernière ne seront pas conservées. Tel que présenté aux sections 2.1.4.1 et 2.1.4.2, trois propriétés sont utilisées pour modéliser les parcelles : la composante y de la position, l'énergie cinétique et la vorticité. L'uniformité de la distribution des particules d'une parcelle nous permet de définir un champ de hauteur représentatif de la géométrie locale. De plus, cette uniformité améliore grandement l'estimation de l'énergie cinétique par parcelle.

La méthode utilisée pour assurer une uniformité de la distribution des particules par parcelle est la semence de marqueurs (figure 2.10). Ces marqueurs sont semés à distance égale l'un de l'autre à l'intérieur des frontières de la parcelle.

Pour chaque marqueur, la particule la plus proche parmi les particules de surface de la cellule est identifiée. Cette particule devient alors un échantillon de la parcelle. Naturellement, une particule identifiée comme échantillon est retirée de la liste pour les prochains marqueurs. S'il n'y a pas suffisamment de particules pour modéliser une parcelle, la cellule (division) est automatiquement rejetée.

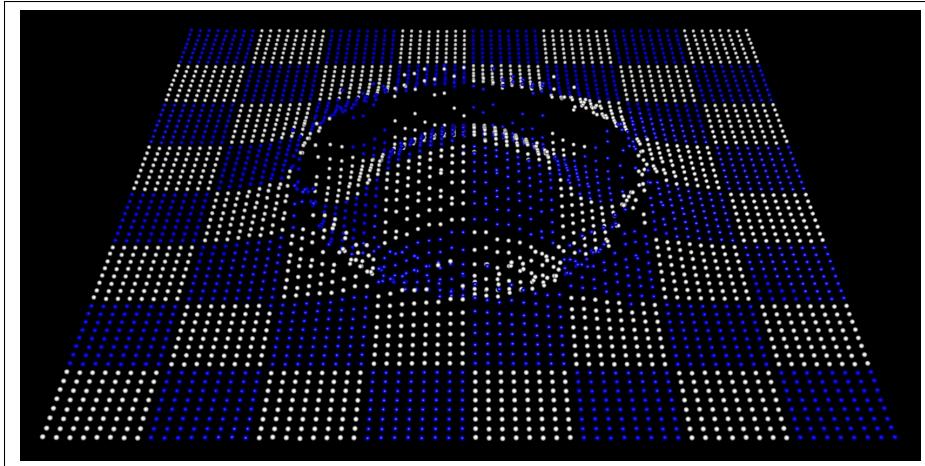


Figure 2.10 Marqueurs utilisés pour échantillonner uniformément les particules de surface par parcelle.

2.2 Apprentissage couplé

L'étape d'apprentissage est un processus plutôt long. Toutes les étapes de préparation des données (section 2.1) sont un prérequis de cette phase d'entraînement. L'apprentissage couplé signifie que celui-ci est fait afin de conserver une relation entre plusieurs propriétés. À titre comparatif, voici le problème traditionnel d'optimisation pour générer un dictionnaire basé sur des exemples :

$$\mathbf{D}_t = \arg \min_{\mathbf{D}} \frac{1}{2} \|\mathbf{x} - \mathbf{D}\alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (2.50)$$

Le premier terme est exprimé en norme l_2 et le second terme en norme l_1 . L'ensemble des atomes du dictionnaire est exprimé par \mathbf{D} . Cette minimisation permet, à l'aide du codage parcimonieux, de déterminer une décomposition α de \mathbf{x} sur \mathbf{D} . Dans cette équation, \mathbf{x} est un vecteur représentant l'ensemble des données à apprendre (image, signal, etc.). La constante λ est le paramètre de régularisation. Ce paramètre permet de balancer la parcimonie de la solution et la fidélité de l'approximation de x . De plus, le codage parcimonieux α est déterminé à l'aide d'un solveur de type LARS. Ce solveur est basé sur une méthode de régression des moindres angles. Il s'agit d'un algorithme de régression efficace pour retrouver une combinaison linéaire sur un ensemble à multiples dimensions. Le vecteur parcimonieux α de la trame t est déterminé par

la fonction objective :

$$\alpha_t = \arg \min_{\alpha} \|\mathbf{x}_t - \mathbf{D}_t \alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (2.51)$$

Le vecteur creux α_t est minimisé par rapport aux atomes du dictionnaire de la trame courante t . Puisque notre problème de minimisation est fait sur l'ensemble des exemples utilisés pour l'entraînement. Le problème peut être écrit comme ceci :

$$\mathbf{D}_t = \arg \min_{\{\mathbf{D}, \alpha\}} \frac{1}{t} \sum_{i=1}^T \left(\frac{1}{2} \|\mathbf{x}_i - \mathbf{D}\alpha_i\|_2^2 + \lambda \|\alpha_i\|_1 \right) \quad (2.52)$$

où t est la trame courante et T le nombre total de trames utilisées à l'entraînement du dictionnaire. La première itération de l'entraînement est faite afin de générer un dictionnaire initial \mathbf{D}_0 . Le dictionnaire initial est basé uniquement sur la trame courante. La génération de \mathbf{D}_0 est faite à partir des parcelles brutes (sans transformation) formées par les particules de surface. Par la suite, le codage parcimonieux trouvé à l'itération précédente est utilisé pour mettre à jour l'itération courante et ainsi de suite. Ce processus de mise à jour est fait sur l'ensemble des trames sélectionnées.

Par ailleurs, avec notre approche, nous devons utiliser trois dictionnaires pour faire l'acquisition des exemples basés sur les trois propriétés : la hauteur des particules, leur énergie cinétique et leur force de vorticité. Ces dictionnaires sont respectivement nommés \mathbf{D}^y , \mathbf{D}^e et \mathbf{D}^ω .

$$\mathbf{D}^y = \arg \min_{\mathbf{D}} \frac{1}{2} \|\mathbf{y} - \mathbf{D}\alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (2.53)$$

$$\mathbf{D}^e = \arg \min_{\mathbf{D}} \frac{1}{2} \|\mathbf{e} - \mathbf{D}\alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (2.54)$$

$$\mathbf{D}^\omega = \arg \min_{\mathbf{D}} \frac{1}{2} \|\omega - \mathbf{D}\alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (2.55)$$

Dans les équations 2.54, 2.55 et 2.55, \mathbf{y} représente l'ensemble des échantillons de hauteur, \mathbf{e} l'ensemble des échantillons d'énergie cinétique et ω l'ensemble des échantillons de vorticité. L'entraînement des trois dictionnaires est fait en simultanée. Cela permet d'évaluer et de sélectionner les parcelles de fluide, et ce, pour les deux résolutions. D'ailleurs, il est important de mentionner que si une parcelle est sélectionnée dans une résolution, elle doit nécessaire-

ment l'être dans l'autre. Toutefois, les propriétés et les dictionnaires peuvent être concaténés pour simplifier la formulation et ainsi minimiser un seul α . La minimisation d'un α commun aux trois dictionnaires est essentielle pour établir une correspondance adéquate à l'étape de reconstruction lors de la synthèse des détails en haute résolution.

$$\mathbf{D} = \arg \min_{\{\mathbf{D}, \alpha\}} \frac{1}{2} \|\mathbf{p} - \mathbf{D}\alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (2.56)$$

où

$$\mathbf{p} = \begin{bmatrix} y_i \\ e_i \\ \omega_i \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} D^y \\ D^e \\ D^\omega \end{bmatrix}. \quad (2.57)$$

2.2.1 Dictionnaire multivariable

Nous parlons d'un dictionnaire multivariable puisque celui renferme plusieurs propriétés des exemples d'entraînement. Les trois dictionnaires, \mathbf{D}^y , \mathbf{D}^e et \mathbf{D}^ω , peuvent être concaténés pour n'en générer qu'un seul.

Le dictionnaire \mathbf{D}^y qui renferme l'information relative à la géométrie des parcelles est construit à partir des exemples de simulation en haute résolution. L'important est d'établir une correspondance entre l'apparence en haute résolution du fluide et les propriétés physiques en basse résolution de celui-ci (tel que présenté à la figure 2.1). Les composantes y de chaque parcelle représentent un champ de hauteurs approximant l'apparence de cette région en haute résolution.

Quant au dictionnaire \mathbf{D}^e , uniquement l'information associée à l'énergie cinétique des parcelles en basse résolution y est conservée. Toutefois, la validité de cette information d'énergie doit absolument être confirmée lors de l'étape de sélection de parcelles. Cette validité est vraie si l'énergie cinétique de la parcelle en haute résolution permet de retrouver l'énergie cinétique de son équivalent en basse résolution. Si cette condition est satisfaite, l'énergie cinétique des

particules de la parcelle en basse résolution est retenue. Il en est de même pour le dictionnaire \mathbf{D}^ω contenant l'information sur la vorticité des parcelles.

2.2.2 Analyse en composantes principales

L'analyse en composantes principales est une méthode qui nous permet d'identifier des patrons dans les données. Cela permet notamment de mettre en évidence les différences et les similitudes de celles-ci. L'avantage considérable de la PCA est qu'une fois les dépendances identifiées, il est possible de réduire la dimensionnalité des données sans perdre l'essentiel de l'information.

Corrélation entre les propriétés

Nous avons deux objectifs à remplir avec la PCA. Le premier objectif est de faire ressortir les corrélations existantes entre les différentes propriétés des particules d'une simulation. Ces corrélations sont notamment la relation entre la courbure et la vorticité. Cette relation peut être mise en évidence par la covariance entre les différentes propriétés des particules. Naturellement, cette analyse doit être faite trame par trame. La première étape est de calculer la matrice de covariance C :

$$C = \begin{bmatrix} cov(p_0, p_0) & cov(p_0, p_1) & \dots & cov(p_0, p_m) \\ cov(p_1, p_0) & cov(p_1, p_1) & \dots & cov(p_1, p_m) \\ \vdots & \vdots & \ddots & \vdots \\ cov(p_m, p_0) & cov(p_m, p_1) & \dots & cov(p_m, p_m) \end{bmatrix} \quad (2.58)$$

Cette matrice est de dimensions $m \times m$. Le terme m est égal au nombre de propriétés utilisées pour construire la matrice de covariance C . Le tableau ci-dessous permet de faire ressortir la relation entre les propriétés géométriques de courbure, l'énergie cinétique et la vorticité. Ces données ont été obtenues à partir d'une simulation à basse résolution (environ 15 000 particules).

Tableau 2.1 Covariances entre des propriétés observées des particules.

	\vec{x}	$\vec{\mu}$	e_k	$\vec{\omega}$	CV	SI	SR
\vec{x}	-	0.001	0.005	0.002	0	0	0
$\vec{\mu}$	0.001	-	0.724	0.312	0.143	0.045	0.056
e_k	0.005	0.724	-	0.757	0.112	0.045	0.441
$\vec{\omega}$	0.002	0.312	0.757	-	0.641	0.566	0.328
CV	0	0.143	0.112	0.641	-	0.173	0.141
SI	0	0.045	0.424	0.566	0.173	-	0.147
SR	0	0.056	0.441	0.328	0.141	0.147	-

La covariance entre les deux propriétés (*e.g.* p_u et p_v) est calculée à l'aide de cette équation :

$$\text{cov}(p_u, p_v) = \frac{\sum_i^N (p_{u_i} - \bar{p}_u)(p_{v_i} - \bar{p}_v)}{N} \quad (2.59)$$

Dans l'équation 2.59, N est le nombre d'échantillons (*e.g.* particules). Plus la valeur de la covariance est grande et plus la dépendance entre ces deux propriétés est forte. D'ailleurs, basés sur le fait que les axes principaux de variance dans les données correspondent aux vecteurs propres de la matrice de covariance C , nous pouvons maintenant déterminer la valeur propre associée. Ce système peut être résolu à partir de cette relation :

$$C = VDV^\top \quad (2.60)$$

Dans cette expression, chaque colonne de la matrice orthogonale V correspond à un vecteur propre de C . Quant à la matrice diagonale D , elle contient les valeurs propres de C . Les deux matrices sont de dimensions $m \times m$.

$$D = \begin{bmatrix} d_0 & 0 & \dots & 0 \\ 0 & d_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & d_m \end{bmatrix} \quad (2.61)$$

Les valeurs propres de la matrice D doivent être triées en ordre décroissant. Plus cette valeur est grande, et plus l'influence de cette propriété sur la variance des données est importante.

$$d_0 \geq d_1 \geq \dots \geq d_m. \quad (2.62)$$

Réduction dimensionnelle de la structure des atomes

Le deuxième objectif est de réduire la taille de l'information stockée par les atomes du dictionnaire. Avec la PCA, il est possible de réduire la dimensionnalité sans pour autant perdre d'information sur les données analysées. À l'aide des valeurs propres, nous pouvons déterminer un certain ratio γ permettant de fixer un taux maximum de perte d'information.

$$\frac{d_0 + d_1 + \dots + d_p}{d_0 + d_1 + \dots + d_m} \geq \gamma \quad (2.63)$$

Dans l'équation 2.63, p premières valeurs propres.

Finalement, les composantes principales sont calculées comme suit :

$$P_{PCA} = \hat{v}(p - \bar{p}). \quad (2.64)$$

Dans l'équation 2.64, \hat{v} représente une matrice des vecteurs propres avec les plus grandes valeurs propres et \bar{p} la valeur moyenne de la propriété. Finalement, voici l'équation afin retrouver les valeurs originales des propriétés à partir des composantes principales :

$$P_{original} = (\hat{v}^\top P_{PCA}) + \bar{p}. \quad (2.65)$$

Le traitement de la PCA est effectué sur l'ensemble des propriétés sélectionnées des particules, c'est-à-dire la hauteur, l'énergie cinétique et la vorticité. Les composantes principales sont évaluées avant l'étape d'apprentissage et la récupération des données est faite à la fin du processus de reconstruction par synthèse de détails présentée au chapitre suivant.

CHAPITRE 3

SYNTHÈSE DE DÉTAILS EN HAUTE RÉSOLUTION

La synthèse de détail en simulation de fluide est le processus qui permet de modéliser des détails à grande échelle à partir de détails à petite échelle. En d'autres termes, il s'agit de regrouper diverses propriétés concernant le fluide à reconstruire afin d'améliorer sa représentation en un tout cohérent. D'ailleurs, l'apprentissage d'un dictionnaire nous permet de rassembler ces éléments qui définissent la surface du fluide. C'est à partir de ces éléments du dictionnaire, aussi appelés atomes, que nous pouvons déterminer une nouvelle surface améliorée, et ce, basée sur la simulation de fluide en basse résolution.

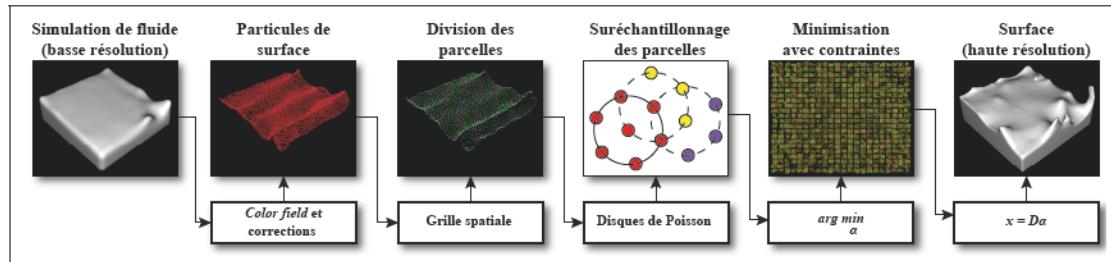


Figure 3.1 Survol des étapes nécessaires pour générer les points de surface à partir d'un dictionnaire couplé et formulé par minimisation sous contraintes.

Dans ce chapitre, l'approche de synthèse de détail utilisée dans le cadre de ce projet de recherche est détaillée. Tout d'abord, la section 3.1 décrit les données de simulation utilisées en entrée à la technique. Ensuite, la section 3.2 présente l'approche choisie afin de modéliser une parcelle sélectionnée de la surface du fluide en basse résolution. Notamment, la méthode utilisée pour suréchantillonner les parcelles y est présentée. Finalement, la section 3.3 explique la reconstruction par minimisation de la surface du fluide.

3.1 Simulation en basse résolution

Les données utilisées en entrée de la technique sont essentiellement les particules provenant d'une simulation de fluide en basse résolution. Naturellement, un dictionnaire doit au préalable avoir été entraîné à partir d'exemples de simulations similaires à celle à reconstruire (figure 2.2 de la section 2.1).

Les simulations utilisées pour la reconstruction ne contiennent pas plus de 100 000 particules. L'approche proposée ne nécessite que quelques milliers de particules pour reproduire l'apparence du fluide en haute résolution.

3.2 Parcelles en basse résolution

Afin de comparer les parcelles de fluide en basse résolution avec celles des exemples d'entraînement en haute résolution, nous devons répéter les étapes décrites aux sections 2.1.1, 2.1.2 et 2.1.5. En résumé, les particules de surface doivent être identifiées, le domaine de simulation du fluide doit être divisé et finalement, les parcelles présentant des caractéristiques géométriques et physiques intéressantes doivent être sélectionnées.

3.2.1 Sélection des parcelles

La seule variante de cette partie de l'approche par rapport à celle utilisée à l'étape d'apprentissage (section 2.1.2) est l'objectif de la sélection des parcelles. À l'apprentissage, la sélection des parcelles est faite dans le but de faire l'acquisition des caractéristiques importantes de la surface de la simulation. Pour la synthèse, cette étape est un processus de sélection qui permet de déterminer les parcelles qui seront à reconstruire. Dans tous les cas (reconstruit ou non), l'ensemble des points de la surface générée par cette approche sera lissé spatialement et temporellement à l'aide d'une matrice laplacienne (section 3.3.2). Néanmoins, il est préférable de reconstruire la surface en entier afin d'éviter l'apparition d'artéfacts générés par l'incohérence temporelle ou spatiale du fluide. Les parcelles à reconstruire sont sélectionnées à partir des discriminants basés sur une valeur moyenne des propriétés physiques (énergie cinétique et vor-

ticité) des particules. Nous avons déterminé les seuils s_e (énergie cinétique) et s_ω (vorticité) par essais-erreurs. Les seuils utilisés pour s_e et s_ω sont respectivement 0.4 et 0.5. Conséquemment, les parcelles qui présentent des valeurs inférieures à ces seuils ne sont pas reconstruites.

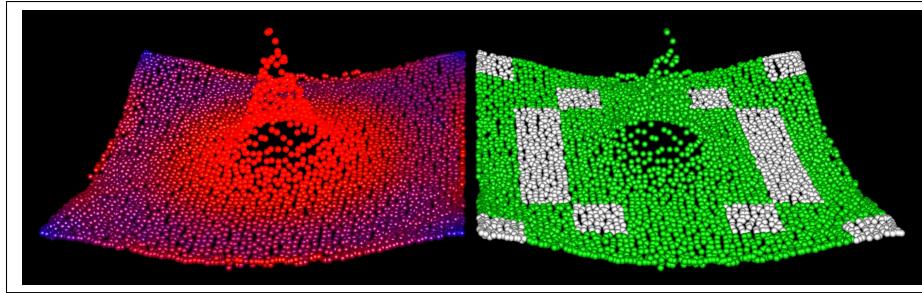


Figure 3.2 Évaluation et sélection des parcelles de fluide à reconstruire basé sur la similarité du spectre. L'évaluation de l'énergie cinétique est effectuée (gauche) afin de connaître quelles parcelles reconstruire (en vert dans l'image de droite).

La figure 3.2 présente une situation où il est préférable de reconstruire toute la surface. Intuitivement, toutes les parcelles sont générées lorsque le nombre de parcelles à reconstruire dépasse la moitié du nombre total de parcelles en surface du fluide. Par ailleurs, lorsque la simulation est stable, ce processus de reconstruction sélectif permet de réduire considérablement les temps de calcul.

3.2.2 Modélisation et suréchantillonnage des parcelles

La division du domaine de simulation, telle que présentée à la section 2.1.5, est utilisée avec le fluide en basse résolution afin de comparer les parcelles à reconstruire avec les éléments du dictionnaire. Actuellement, la taille des parcelles est égale à celle des atomes (parcelles en haute résolution) du dictionnaire. Néanmoins, certaines parcelles de la surface du fluide en basse résolution n'ont pas suffisamment de particules pour satisfaire cette exigence. Ce problème survient particulièrement aux frontières de la simulation (figure 3.3). Pour résoudre ce problème, nous ne considérons pas les parcelles aux frontières lors de l'entraînement. Nous

faisons l'hypothèse que les autres parcelles sélectionnées sont suffisantes pour reconstruire les parcelles aux frontières. Si ce n'est pas le cas, il est possible de reconstruire ces parcelles basées uniquement sur les propriétés géométriques de la surface. Puisque les frontières présentent rarement des aspects intéressants du fluide, nous estimons que la perte de détail n'est pas significative.

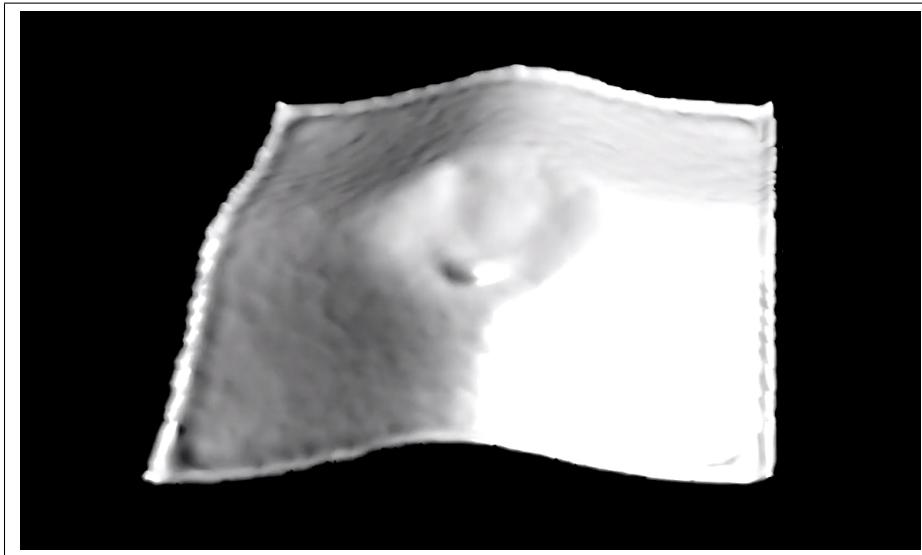


Figure 3.3 Artéfacts indésirables de reconstruction aux frontières causés par la quantité insuffisante de particules dans ces régions.

Dans certains cas plus complexes, des portions du fluide subissent de grands changements par *pas de temps*. Dans la plupart des cas, nous ne considérons pas les scénarios qui présentent ce genre de comportement. Une mauvaise paramétrisation de la méthode d'identification des particules de surface peut aussi être en cause (figure 3.4). Ce type de problème est plus courant avec les simulations en basse résolution. Ceci est principalement dû à la distance plus grande de séparation entre les particules de la simulation.

Une méthode de suréchantillonnage peut être utilisée pour corriger la distribution des particules. La méthode utilisée pour semer de nouvelles particules à l'intérieur d'une parcelle est l'échantillonnage par disques de Poisson (PDS). Dans cette approche, nous avons utilisé l'al-

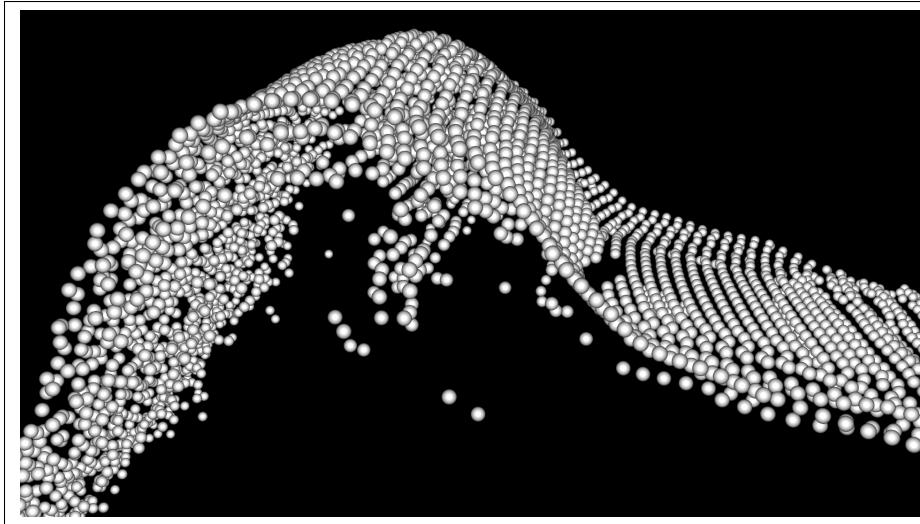


Figure 3.4 Seuil erroné avec la méthode du *color field*.

gorithme d'échantillonnage rapide par disques de Poisson (Bridson, 2007). Notre approche du PDS est utilisée en deux dimensions et demie. La seule différence avec l'approche originale de Bridson est que le disque permet d'échantillonner en deux dimensions sur un plan et ensuite interpoler la hauteur par rapport aux particules voisines. À partir de chaque particule de la parcelle, nous semons de nouvelles particules aux frontières du disque d'influence (figure 3.5). Ce disque a un rayon d'échantillonnage égal à la moitié de la distance de séparation des particules de la simulation en basse résolution. Naturellement, les particules semées sont aussi considérées par les itérations ultérieures de l'algorithme.

La condition d'arrêt de l'approche est lorsque le centre du disque d'influence dépasse les frontières de la parcelle. Ces frontières sont définies par la même grille spatiale utilisée à l'étape de modélisation des parcelles (section 2.1.5). Le rayon du disque peut dépasser afin de conserver une cohérence spatiale entre la parcelle et ses voisins.

L'avantage considérable avec cette technique d'échantillonnage est qu'elle permet de préserver une distribution également espacée des particules en surface du fluide. Cette distribution est similaire à l'effet des forces d'attraction-répulsion des particules. Les propriétés des nouvelles particules semées par la méthode *PDS* sont obtenues par une interpolation trilinéaire basée

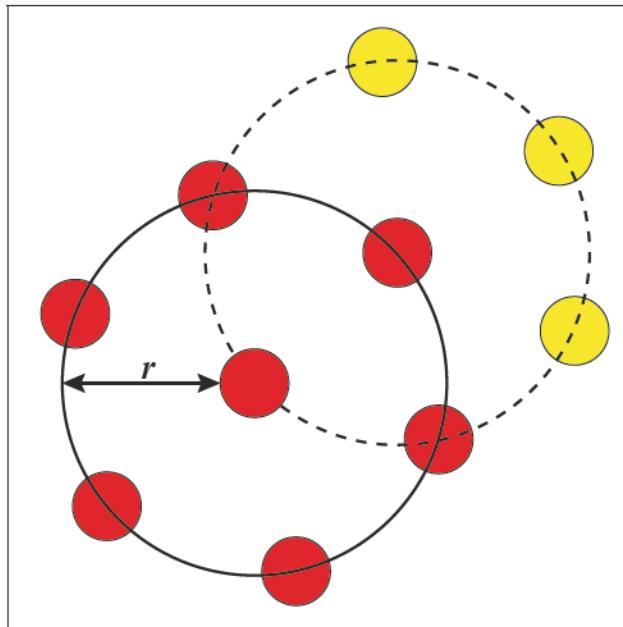


Figure 3.5 Distribution de l’approche d’échantillonnage par disque de Poisson en deux dimensions. Dans cette figure, nous présentons les deux premières itérations (rouge et jaune) de l’algorithme.

sur leurs voisins. La figure 3.6 présente une situation où le nombre de particules par parcelle est insuffisant (image de gauche). À l’image de droite de cette même figure, la parcelle et ses voisins sont suréchantillonnés.

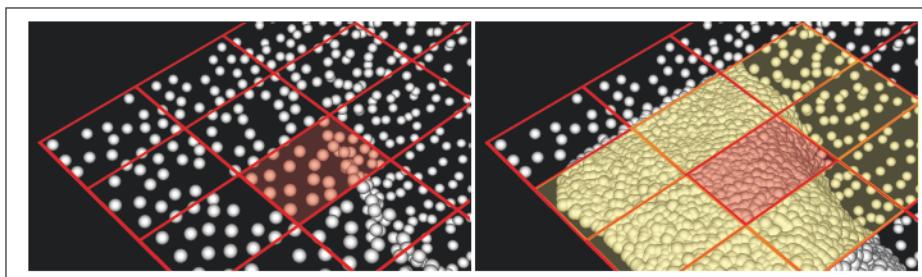


Figure 3.6 Suréchantillonnage par disques de Poisson d’une parcelle (rouge) et de ses voisins (jaune).

3.3 Reconstruction par optimisation

La reconstruction par optimisation signifie que nous tentons de trouver une parcelle en haute résolution à l'aide des correspondances établies entre les parcelles en basse résolution et les parcelles utilisées pour générer le dictionnaire en haute résolution. Cette approche permet de minimiser la différence entre la parcelle et ses propriétés en entrée et la parcelle résultante issue des atomes du dictionnaire. Une combinaison linéaire pondérée permet d'obtenir une parcelle similaire à celle en entrée (figure 3.7), mais tout en respectant les propriétés physiques de celle-ci. La correspondance en haute résolution est possible uniquement par le fait que des caractéristiques simples, telles que les hauteurs, de parcelles en haute résolution sont associées à des propriétés physiques de parcelles en basse résolution.

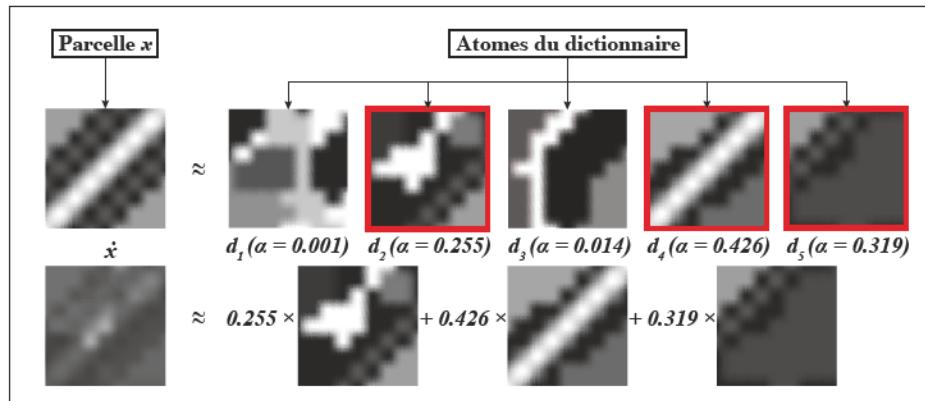


Figure 3.7 Reconstruction d'une parcelle à partir d'une combinaison linéaire obtenue par un problème de minimisation des atomes du dictionnaire.

La correspondance est faite principalement par les propriétés physiques des échantillons (particules) de la parcelle. La géométrie de la parcelle en entrée sert uniquement à aiguiller le choix des atomes du dictionnaire. L'influence de la géométrie est moindre que les autres propriétés puisque celle-ci est basée uniquement sur l'information des parcelles de la simulation en basse résolution utilisée en entrée.

3.3.1 Formulation par minimisation

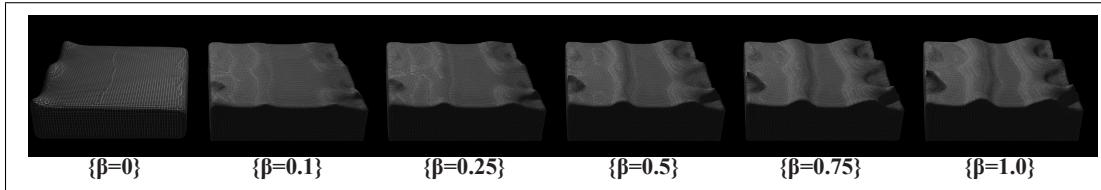


Figure 3.8 Compromis de reconstruction entre les propriétés géométriques et les propriétés physiques.

Une minimisation basée uniquement sur les points de hauteur d'une parcelle aurait pour effet de reconstruire cette même parcelle sans y ajouter de détails. Dans la figure 3.8, β représente le poids accordé à l'importance d'une similarité des parcelles au niveau de l'énergie cinétique. Tel qu'illustré par cette figure, plus la valeur de β est faible et plus la surface générée par notre approche ressemble à la simulation en basse résolution à gauche de l'image. En d'autres mots, $\beta = 0$ est équivalent à la simulation en basse résolution fournie en entrée.

Des termes additionnels de pénalité doivent être établis afin de déterminer les atomes correspondant à la situation en haute résolution. Ces contraintes additionnelles sont basées essentiellement sur l'énergie cinétique et la vorticité des échantillons par parcelle et sont détaillées à la section suivante.

La formulation de la minimisation doit tenir compte de trois termes. Ces termes sont représentés par la hauteur, l'énergie cinétique et la vorticité des particules de surface utilisées en entrée. Ces propriétés sont contenues respectivement dans les matrices \mathbf{y} , \mathbf{e} et $\boldsymbol{\omega}$. Cette minimisation est effectuée par parcelle de fluide. D'ailleurs, le résultat de cette minimisation est en fait un vecteur creux des atomes pondérés. La somme de ces pondérations est une combinaison linéaire des atomes du dictionnaire. La représentation initiale de ce problème de minimisation, basé uniquement sur les hauteurs des particules, est formulée de la façon suivante :

$$\alpha_i = \arg \min_{\alpha} \frac{1}{2} \|y_i - D\alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (3.1)$$

Le second terme $\lambda \|\alpha\|_1$ est une pénalité de parcimonie qui force la représentation finale de y_i à être creuse. Ce terme est aussi appelé un terme de régularisation dans ce type de problème d'optimisation. Basé sur les travaux et expérimentations de plusieurs auteurs (Elad et Aharon, 2006; Aharon et Elad, 2008; Mairal *et al.*, 2009a), nous avons utilisé une valeur similaire à ceux-ci (0.15) pour λ . Le vecteur creux α_i est la combinaison linéaire qui offre la meilleure solution pour reconstruire la parcelle i à partir des atomes du dictionnaire. Toutefois, cette solution n'est basée que sur les hauteurs des particules. Nous devons ajouter les deux autres termes de pénalité des propriétés des particules afin de conserver la cohérence physique de la surface du fluide :

$$\alpha_i = \arg \min_{\alpha} \frac{1}{2} \|\mathbf{y}_i - D^y \alpha\|_2^2 + \|\mathbf{e}_i - D^e \alpha\|_2^2 + \|\omega_i - D^\omega \alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (3.2)$$

De plus, des constantes sont placées devant chaque terme de pénalité en lien avec les propriétés physiques d'une parcelle afin de contrôler l'importance de ceux-ci sur le problème d'optimisation.

$$\alpha_i = \arg \min_{\alpha} \frac{1}{2} \|\mathbf{y}_i - D^y \alpha\|_2^2 + \frac{\beta}{2} \|\mathbf{e}_i - D^e \alpha\|_2^2 + \frac{\gamma}{2} \|\omega_i - D^\omega \alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (3.3)$$

Par la suite, puisque les atomes des dictionnaires peuvent être concaténés (semblable à l'équation 2.56), le problème peut être reformulé comme ceci :

$$\alpha_i = \arg \min_{\alpha} \frac{1}{2} \|\mathbf{p}_i - \mathbf{D}\alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (3.4)$$

où

$$\mathbf{p}_i = \begin{bmatrix} y_i \\ \beta e_i \\ \gamma \omega_i \end{bmatrix}, \mathbf{D} = \begin{bmatrix} D^y \\ \beta D^e \\ \gamma D^\omega \end{bmatrix}. \quad (3.5)$$

Tel que mentionné aux sections 2.1.4.1 et 2.1.4.2 , \mathbf{p}_i est une parcelle contenant toutes les propriétés sélectionnées des particules de surface. Chaque colonne du dictionnaire \mathbf{D} est un atome d_i dont la somme est la concaténation de chaque atome des dictionnaires généré à l'entraînement.

$$d_i = \begin{bmatrix} y_{i0} \\ y_{i1} \\ \vdots \\ y_{iN} \\ \beta e_{i0} \\ \beta e_{i1} \\ \vdots \\ \beta e_{iN} \\ \gamma \omega_{i0} \\ \gamma \omega_{i1} \\ \vdots \\ \gamma \omega_{iN} \end{bmatrix} \quad (3.6)$$

Afin de n'évaluer que les différences marquantes entre les particules de surface des deux résolutions, particulièrement au niveau des hautes fréquences, la moyenne des hauteurs de chaque parcelle est soustraite de celle-ci avant l'étape de minimisation (équation 3.4). Ayant soustrait cette moyenne par parcelle, la solution finale de α_i représente un déplacement vertical des particules. D'ailleurs, à ce stade, seulement l'information concernant les hauteurs demeure importante à la reconstruction des nouvelles parcelles. Les autres propriétés physiques ont servi à déterminer ces hauteurs. Puisque la hauteur est la première propriété concaténée des atomes, seulement les n premiers éléments $\hat{\alpha}_i$ sont retenus. Le vecteur creux $\hat{\alpha}_i$ est un sous-ensemble de α_i .

$$\hat{\alpha}_i \subset \alpha_i. \quad (3.7)$$

Le terme n représente le nombre d'échantillons par parcelle. Ceci dit, les hauteurs de la parcelle reconstruite $\hat{\mathbf{y}}_i$ (équation 3.8) peuvent être obtenues par la multiplication du dictionnaire D^y

avec le vecteur creux $\hat{\alpha}_i$ (équation 3.4) tel que présenté ci-dessous :

$$\hat{\mathbf{y}}_i = D^y \hat{\alpha}_i \quad (3.8)$$

Finalement, afin de retrouver les hauteurs relatives des particules, nous devons additionner les moyennes par parcelle soustraite précédemment :

$$\mathbf{y}_{i_{final}} = \hat{\mathbf{y}}_i + \bar{\mathbf{y}}_i \quad (3.9)$$

3.3.2 Cohérence spatiale et temporelle

La cohérence spatiale et temporelle est effectuée en *post-traitement*, donc après l'étape de minimisation (section 3.3.1). L'idée d'un tel traitement est de calculer une nouvelle surface y' qui minimise les discontinuités spatiales et temporelles sur l'ensemble des parcelles. De cette façon, une surface plus lisse peut être obtenue lors de la reconstruction du maillage basée sur les particules à l'interface du fluide. Dans les équations suivantes, y' est un vecteur des hauteurs des particules de surface.

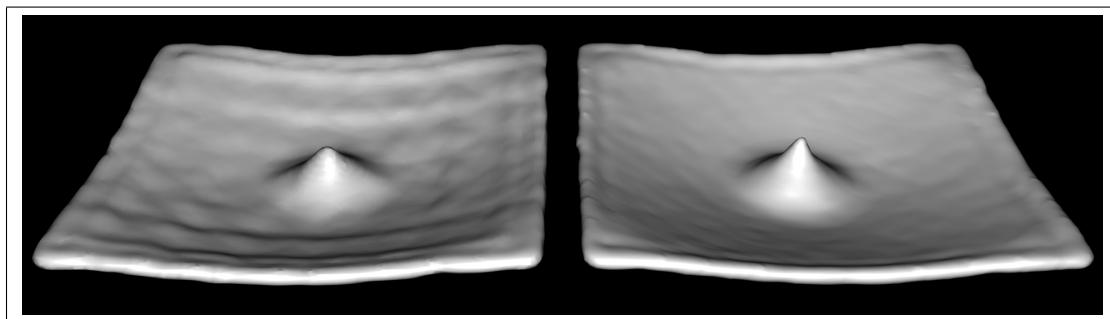


Figure 3.9 Cohérence spatiale sur la surface générée. Les artefacts générés entre les parcelles (gauche) sont atténués par les termes de cohérence spatiale et temporelle (droite)

Tout d'abord, pour s'assurer de ne pas trop s'éloigner de la solution courante, le problème de minimisation est initialement défini comme ceci :

$$\arg \min_{y'} \frac{\tau}{2} \|y' - y_t\|_2^2 \quad (3.10)$$

Dans l'équation 3.10, t est la trame courante de la simulation. Ensuite, on peut ajouter le terme de cohérence temporelle :

$$\arg \min_{y'} \frac{\tau}{2} \|y' - y_t\|_2^2 + \frac{1-\tau}{2} \|y' - y_{t-1}\|_2^2 \quad (3.11)$$

Finalement, le troisième terme permet de forcer la cohérence spatiale de la solution.

$$\arg \min_{y'} \frac{\tau}{2} \|y' - y_t\|_2^2 + \frac{1-\tau}{2} \|y' - y_{t-1}\|_2^2 + \frac{\phi}{2} (y'^\top \mathbf{L} y') \quad (3.12)$$

Dans l'équation 3.12, les deux premiers termes permettent de conserver la cohérence temporelle. Ces termes sont liés par un paramètre de contrôle utilisateur τ . Si τ est égale à 1, la solution ne prendra pas en compte la parcelle de la trame précédente. La matrice laplacienne \mathbf{L} permet d'établir une cohérence (continuité) entre les hauteurs de la parcelle courante et ses 8 voisins. Cette matrice est obtenue à partir de la matrice creuse d'adjacence \mathbf{A} et de la matrice diagonale \mathbf{D} comme suit :

$$\mathbf{L} = \mathbf{D} - \mathbf{A} \quad (3.13)$$

La matrice d'adjacence est obtenue à la construction des relations pondérées entre les voisins. Puisque notre modèle de parcelle est obtenu à partir d'un seul plan, seulement 8 voisins seront considéré. Le noyau utilisé pour construire la matrice \mathbf{A} est celui-ci :

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (3.14)$$

Finalement, la matrice diagonale ($N \times N$) est calculée à partir du résultat de la matrice d'adjacence :

$$\mathbf{D} = \begin{bmatrix} \sum_j a_{ij} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sum_j a_{ij} \end{bmatrix} \quad (3.15)$$

où N est le nombre de particules. Cette partie de l'approche permet de réduire les effets indésirables pouvant survenir spatialement et temporellement entre les parcelles à la surface d'un fluide. La figure 3.9 présente le résultat sur la cohérence spatiale de la surface générée. En fait, il est important de souligner que les termes de cohérence spatiale et temporelle sont appliqués sur les points de surface. Un maillage est ensuite généré à partir de ces points de surface.

CHAPITRE 4

IMPLÉMENTATION

Ce chapitre fournit des détails sur l'implémentation des étapes qui sont décrites dans les diverses sections de ce mémoire. Certaines précisions permettent de mieux cerner les difficultés associées à l'élaboration de l'approche. Tout d'abord, la section 4.1 présente le fonctionnement global du simulateur utilisé. Ensuite, l'environnement de développement utilisé est décrit à la section 4.2. Certaines précisions sur la préparation des données d'entraînement et l'apprentissage à base de parcelles sont discutées aux sections 4.3 et 4.4. De plus, la reconstruction de la surface est brièvement abordée à la section 4.5. Finalement, d'autres précisions concernant les paramètres utilisés et les temps de calcul sont présentées à la section 4.6.

4.1 Simulateur par particules

Pour la plupart des expérimentations faites dans le cadre de ce projet, nous avons utilisé le simulateur conçu par l'équipe de recherche et développement de Mokko Studio. Ce simulateur a été implémenté dans *Houdini* sous licence de recherche académique. Il s'agit d'un simulateur lagrangien de liquide par particules SPH. La simulation est principalement divisée en deux phases : l'initialisation et la boucle de la simulation.

La phase d'initialisation est faite à partir de la géométrie de l'état initial de la simulation. Cette géométrie initiale n'est rien d'autre qu'un nuage de points en trois dimensions. Cet ensemble de points est instancié à partir d'un algorithme de remplissage de volume disponible avec le logiciel *Houdini* de *Side Effects*. À partir de cette géométrie (points), le simulateur peut initialiser certains paramètres de simulation tels que les forces gravitationnelles, le rayon d'influence des particules et bien d'autres. Ces valeurs sont insérées à l'intérieur d'une structure définie par le simulateur. De plus, la masse de chaque particule est initialisée.

Lorsque la phase d'initialisation est complétée, la simulation peut être exécutée. L'exécution de la simulation est une séquence bouclée d'étapes répétées sur un certain nombre de trames.

Ces étapes sont les calculs des forces permettant d'établir la mécanique du mouvement des particules (section 2.1.4.2). Le fluide évolue dans le temps, par incrément de temps (pas de temps), jusqu'à la dernière trame.

Les étapes sont notamment le calcul de la masse volumique, la pression, les forces SPH et le nouveau *pas de temps* (si adaptatif). Ces calculs sont détaillés à la section 2.1.4.2. Finalement, la position et la vitesse des particules sont mises à jour. Il est important de mentionner qu'après la phase d'initialisation, une grille spatiale est utilisée pour générer la liste des voisins pour chaque particule. Cette liste est utilisée pour trouver rapidement les particules voisines et déterminer l'influence SPH des celles-ci sur une particule courante.

L'utilisation d'un simulateur maison a ses avantages et ses inconvénients. L'inconvénient est essentiellement au niveau de la stabilité du simulateur. Puisqu'il s'agit d'un produit en développement, celui-ci n'est pas à l'abri des erreurs de calcul et des temps de simulation non optimaux. L'avantage non négligeable est la flexibilité d'avoir en main une solution maison. L'ajout d'étapes supplémentaires, telles que le calcul du spectre d'énergie cinétique et des forces de vorticité, se fait avec simplicité et sans difficulté. De plus, l'analyse et le débogage peuvent s'avérer pratiques dans certaines situations complexes.

4.2 Environnement de développement

Le développement de ce projet de recherche a été fait entièrement en C++ et à l'aide de l'interface de programmation (API) *Houdini Development Kit*. Cet API permet de créer des modules d'extension à l'intérieur du logiciel *Houdini* de *Side Effects*. Ces modules d'extension permettent d'interagir avec différents aspects des opérations disponibles dans Houdini. La version utilisée d'*Houdini* est la 12.5.

Pour l'implémentation, nous avons opté pour l'utilisation du développement nodal (figure 4.1). Il s'agit d'un aspect excessivement puissant d'*Houdini*. Pour toutes les étapes de l'approche, nous avons développé des noeuds de modification et de création de géométries (*SOP*). La définition et l'implémentation d'une sous-classe de *SOP_Node* sont simples. Nous devons respec-

ter une structure afin d'implémenter certaines méthodes et fonctions virtuelles. De plus, la compilation est faite à l'aide des librairies *HDK* et d'un script *hdk-config.py* qui initialise l'environnement nécessaire au fonctionnement du module d'extension dans *Houdini*. Finalement, la version compilée (fichier .so) doit être déposée dans l'un des répertoires analysés par *Houdini*. La procédurale détaillée pour créer un module d'extension maison pour *Houdini* est disponible sur leur site web : http://www.sidefx.com/docs/hdk13.0/_h_d_k__intro__creating_plugins.html

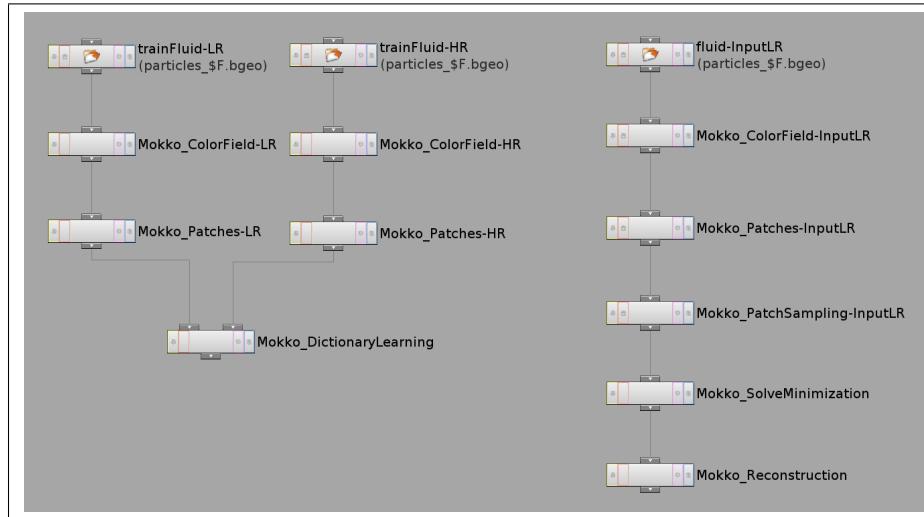


Figure 4.1 Représentation nodale dans *Houdini* des étapes de l'approche présentée dans ce document.

Nous avons décidé d'utiliser un développement nodal pour simplifier les interactions entre les différentes tâches de l'approche. Chaque noeud fonctionne avec de la géométrie en entrée et en sortie. En outre, en utilisant *Houdini*, nous avons accès à toute la puissance offerte par le moteur de rendu de celui-ci (appelé *Mantra*), et ce, sans devoir convertir nos données avec un logiciel tiers. De plus, l'interface graphique d'*Houdini* facilite la paramétrisation des étapes de l'approche.

4.3 Préparation des exemples d’entraînement

La préparation des exemples d’entraînement comprend la simulation des multiples scénarios, l’extraction des particules de surface et l’utilisation de l’analyse en composantes principales pour réduire la dimensionnalité des atomes des dictionnaires.

4.3.1 Identification des particules de surface

Une grille spatiale est utilisée à l’étape d’identification des particules de surface. Cette structure permet de réduire les temps de calcul associés à l’identification des particules de surface. Suite à l’extraction des particules de surface par la méthode du *color field*, chaque particule est associée à une cellule de la grille.

Par la suite, le traitement additionnel d’identification des particules de surface (section 2.1.1) est fait uniquement à l’intérieur des cellules qui contiennent des particules. Cette grille est représentée comme une matrice creuse où les éléments sont les cellules actives. La résolution de la grille dépend de la résolution (nombre de particules) de la simulation par particules. Plus le nombre de particules est élevé et plus la distance d’espacement entre celles-ci est petite. La résolution de la grille doit donc être ajustée en conséquence afin de capturer adéquatement la surface du fluide. Aucun des scénarios utilisés n’a nécessité une grille de dimensions supérieures à $256 \times 256 \times 256$.

4.3.2 Analyse en composantes principales

L’analyse en composantes principales est un outil qui a permis d’identifier les relations entre les différentes propriétés du fluide. Les calculs matriciels ont été faits en partie à l’aide de la librairie Eigen (Guennebaud *et al.*, 2010). Mis à part les structures matricielles de la librairie, nous avons utilisé le solveur *EigenSolver* pour calculer les valeurs propres et les vecteurs propres de la PCA à partir de la matrice de covariance.

De plus, la PCA nous permet de réduire la dimensionnalité des atomes du dictionnaire. Ce traitement n'est effectué que sur le dictionnaire final. Toutefois, cette étape n'aurait pas été nécessaire pour les scénarios utilisés dans nos expérimentations de l'approche finale. Le choix d'utiliser la PCA dans notre approche a pour but d'offrir une flexibilité d'observation des propriétés. Ceci dit, le traitement de la PCA n'est pas justifiable avec moins de 5 propriétés par particule. Toutes les expérimentations à l'étape d'apprentissage ont été faites sur des particules possédant seulement 3 propriétés (position, énergie cinétique et vorticité).

4.4 Apprentissage des parcelles

Les étapes d'apprentissage par dictionnaire et de factorisation matricielle ont été effectuées à l'aide de la librairie *SPAMS* (Mairal *et al.*, 2009a). Cette librairie est développée pour fonctionner sous Matlab ou Python. Pour des fins de performances, nous avons adapté la librairie en C++ afin de l'intégrer dans notre développement sous le *HDK d'Houdini*.

Une classe générée pour l'entraînement par codage parcimonieux est disponible. De plus, *SPAMS* offre plusieurs solveurs pour la décomposition de problème avec représentation creuse. Le solveur LARS a été utilisé pour résoudre notre structure creuse par décomposition. D'ailleurs, cette librairie offre aussi des structures pour gérer les opérations sur les vecteurs creux et les matrices creuses. À cet effet, une documentation plutôt complète est disponible sur le site web de la librairie. La documentation a été créée pour Matlab et Python. Néanmoins, à partir de ces ressources, l'intégration en C++ est possible.

4.4.1 Stockage des données du dictionnaire

Le stockage des propriétés des particules est fait à la génération du dictionnaire à l'étape d'apprentissage (section 2.2). Un fichier est créé pour chaque dictionnaire. La structure du fichier est simple et son contenu est sous format texte (donc pas binaire). L'entête de chaque fichier comprend la taille du dictionnaire. La taille du dictionnaire est définie par le nombre de particules de surface par parcelle et le nombre de parcelles sélectionnées lors de la phase d'entraî-

nement. Cette taille est utilisée pour initialiser les dimensions du dictionnaire D_0 à la lecture du fichier associé. De plus, l'entête comprend aussi les valeurs minimum et maximum enregistrées pendant les itérations de mise à jour du dictionnaire. Toutes les autres lignes du fichier sont les atomes du dictionnaire sous forme matricielle. Par conséquent, chaque ligne représente un atome du dictionnaire.

4.4.2 Affichage des atomes

L'affichage des atomes est une option (boîte à cocher) disponible par le noeud d'apprentissage développé dans *Houdini*. Un traitement supplémentaire est exigé pour créer la géométrie nécessaire pour représenter les atomes du dictionnaire. La méthode utilisée pour afficher les atomes est tirée de Mairal *et al.* (2010). Dans un premier temps, tous les échantillons des atomes du dictionnaire sont normalisés à l'aide des valeurs minimum m et maximum M . Ensuite, afin de faire ressortir les éléments du codage parcimonieux, on doit déterminer ces deux variables :

$$\varepsilon = \begin{cases} 0 & \text{si } m \geq 0 \\ \bar{D} & \text{si } m < 0 \end{cases} \quad (4.1)$$

$$\sigma = \begin{cases} \sqrt{\bar{D}\bar{D}} & \text{si } m \geq 0 \\ \sqrt{(\bar{D}-\varepsilon)^2} & \text{si } m < 0 \end{cases} \quad (4.2)$$

Dans les équations 4.1 et 4.2, \bar{D} est la valeur moyenne des atomes du dictionnaire. Ensuite, les atomes normalisés du dictionnaire sont obtenus à l'aide de cette équation :

$$D = \frac{D' - m}{M - m} \quad (4.3)$$

où D' est initialement ajusté à partir des termes ε et σ déterminés aux équations 4.1 et 4.2.

$$D' = \min(\max(D - \varepsilon, -\phi\sigma), \phi\sigma) \quad (4.4)$$

Le terme ϕ est une constante permettant d'ajuster l'intensité des caractéristiques des atomes affichés. La valeur de ϕ a été fixée à 4.5 pour respecter la même démarche utilisée par (Mairal *et al.*, 2010) et dans les exemples d'utilisation de la librairie *SPAMS*.

Finalement, ces valeurs de hauteur sont utilisées pour générer des polygones (figure 4.2) représentant les atomes du dictionnaire. Nous avons utilisé une polygonisation de Delaunay pour construire le maillage des atomes dans *Houdini*.

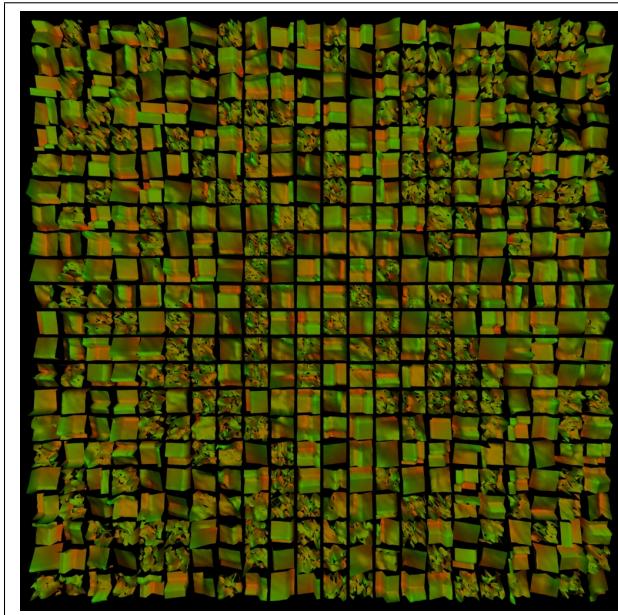


Figure 4.2 Représentation polygonale des atomes dans *Houdini*.

4.5 Reconstruction de la surface

La reconstruction de la surface se fait en deux temps. Premièrement, nous devons déterminer une solution par parcelle afin d'avoir les particules de surface résultantes de la minimisation. Ces hauteurs sont obtenues à l'aide du solveur Lasso disponible dans *SPAMS*. Ce solveur nous permet de résoudre un problème sous formulation de Lasso, c'est-à-dire la minimisation d'un moindre carré avec pénalité sous norme l_1 . Cette fonction de la librairie nous retourne un vecteur creux α pour la parcelle courante. Cette solution nous permet de calculer les points de

surface de chaque parcelle (équation 3.4). La solution retournée est en fait un simple champ de hauteurs (figure 4.3). Aux premières itérations de la figure 4.3, les parcelles sont planes puisqu'il n'y presque aucune contribution des atomes du dictionnaire et que les parcelles sont initialisées avec une hauteur moyenne.

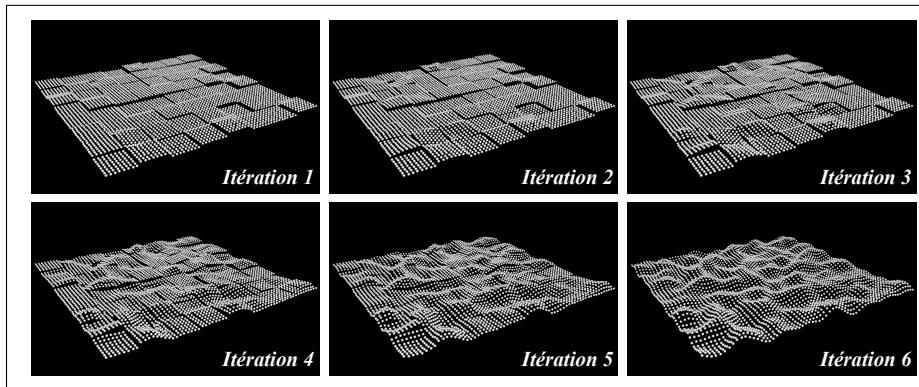


Figure 4.3 Convergence des points de surface avec l'approche par minimisation.

Après avoir ajusté ces points de surface avec les termes de cohérence spatiale et temporelle, nous pouvons reconstruire une nouvelle surface lisse à l'aide d'un maillage polygonal. La technique utilisée pour générer le maillage est basée sur l'approche de Solenthaler *et al.* (2007). Cette technique de reconstruction de surface a été développée et intégrée dans *Houdini* par l'équipe de recherche et développement de Mokko Studio.

4.6 Paramétrisation et temps de calcul de l'approche

La paramétrisation des différentes techniques utilisées dans cette approche est importante. De plus, afin de rendre ces travaux de recherche reproductibles par d'autres, les valeurs utilisées pour ces nombreux paramètres doivent être exposées. Par la suite, quelques temps de calcul sont présentés. Les temps ont été enregistrés par étape de l'approche. Ces estimations de temps ont été faites à partir d'une implémentation personnalisée basée sur la classe *Chrono* offert avec

le standard C++11. Cette version du chronomètre permet d'obtenir des temps en millisecondes avec une précision 64 bits.

Paramétrisation des étapes

Toutes les simulations de fluide utilisées dans nos expérimentations ont été générées à partir du simulateur développé par l'équipe de recherche et développement de Mokko Studio. Le tableau 4.1 présente la liste des principaux paramètres et valeurs utilisés pour simuler les scénarios de simulation de fluide :

Tableau 4.1 Paramètres utilisés pour générer les simulations de fluide par particules SPH.

Scénario	Nombre de particules	Séparation des particules	Rigidité (k)	Vorticité (ε)
Simulations en haute résolution				
Goutte	378 535	0.03	14.0	0.05
Gravité	515 605	0.02	14.0	0.02
Barrage	734 200	0.02	14.0	0.01
Simulations en basse résolution				
Goutte	28 770	0.08	14.0	0.05
Gravité	32 566	0.08	14.0	0.02
Barrage	42 291	0.08	14.0	0.01

De plus, à l'étape d'identification et d'extraction de particules de surface, les paramètres présentés au tableau 4.2 ont été utilisés. Dans le tableau 4.2, le seuil de surface est le paramètre qui

Tableau 4.2 Paramètres utilisés pour identifier les particules de surface.

Scénario	Color field			Cylindre englobant	
	Seuil de surface (s_c)	Contribution des voisins	Seuil de frontière (s_f)	Dimensions de la grille	Volume (r, h)
Goutte	10.0	25	0.06	128^3	(0.15, 1.0)
Gravité	10.5	40	0.075	192^3	(0.1, 1.0)
Barrage	11.5	50	0.08	256^3	(0.1, 1.0)

détermine l'épaisseur de la couche de surface extraite du fluide. La grandeur de ce paramètre est dépendante de la résolution de la simulation. Quant au seuil de frontière, ce paramètre permet d'éliminer les particules près des frontières de la simulation (particulièrement les côtés et le dessous). La paramétrisation optimale de ce genre d'approche d'extraction des particules de surface est pour la plupart du temps déterminée par un processus d'essais-erreurs.

Dans le même ordre d'idée, les paramètres utilisés par la méthode d'apprentissage automatique sont excessivement importants. Le tableau 4.3 résume cette paramétrisation. Les mêmes paramètres ont été utilisés pour tous les scénarios d'entraînement.

Tableau 4.3 Paramètres utilisés par la méthode d'apprentissage par dictionnaire.

Nombre d'atomes	Taille des parcelles	Nombre d'itérations	Parcimonie (λ)	Optimisation <i>mini-batch</i> (μ)
512	64 (8^2)	1 000	0.15	256

Le nombre d'atomes est déterminé à l'initialisation de D_0 . De plus, la taille des parcelles est choisie en fonction de la résolution de la simulation. Plus le nombre de particules est grand et plus la taille des parcelles peut être grande. D'autre part, le nombre d'itérations et l'optimisation *mini-batch* sont les paramètres qui déterminent la précision de la représentation des atomes dans le dictionnaire. L'optimisation *mini-batch* permet d'accélérer la convergence de la méthode de minimisation (Mairal *et al.*, 2009a). Cette méthode augmente la rapidité de l'approche par un facteur de 10 (en moyenne). Finalement, la parcimonie (*sparsity*) est le paramètre de régularisation. La même valeur de λ est utilisée lors de la reconstruction.

Temps de calcul

Les temps de calcul ont été divisés en deux sections. La première section regroupe toutes les étapes d'apprentissage et la deuxième section regroupe toutes les étapes de synthèse des détails. Les temps de calcul de la phase d'apprentissage sont composés de la préparation des données et de l'entraînement. Le tableau 4.4 expose ces temps.

Tableau 4.4 Temps de calcul des étapes de la phase d'apprentissage.

Scénario	Identification des particules de surface	Sélection et transfert des propriétés	Modélisation des parcelles	Entraînement des dictionnaires	Total
Goutte	2 127 ms	1 241 ms	620 ms	42 min 7 s	42 min 11 s
Gravité	2 843 ms	1 370 ms	711 ms	84 min 31 s	84 min 36 s
Barrage	3 261 ms	1 410 ms	804 ms	111 min 16 s	111 min 16 s

Le scénario Goutte a été entraîné sur 320 trames, donc environ 8 secondes par trame. Quant aux scénarios Gravité et Barrage, ils ont été entraînés respectivement sur 505 trames (environ 10 secondes par trame) et 612 trames (environ 11 secondes par trame).

Tableau 4.5 Temps de calcul par trame des étapes de reconstruction.

Scénario	Préparation des parcelles	Suréchantillonnage	Minimisation et points de surface	Total
Goutte	199 ms	132 ms	2 017 ms	2 348 ms
Gravité	246 ms	150 ms	2 924 ms	3 320 ms
Barrage	274 ms	201 ms	3 749 ms	4 224 ms

Pour tous les scénarios reconstruits, $\beta = 0.75$ (contrainte sur l'énergie cinétique) et $\gamma = 0.25$ (contrainte sur la vorticité) à l'étape de la minimisation par parcelle. En ce qui concerne la cohérence spatiale ϕ et temporelle τ , nous avons utilisé la valeur 0.5 afin de restreindre le lissage de la surface. De même, les temps de calcul de la phase de reconstruction sont composés de la préparation des parcelles et de la reconstruction par minimisation. Le tableau 4.5 expose ces temps. L'étape de préparation des parcelles regroupe l'identification des particules de surface, la sélection et la modélisation des parcelles. Ces temps de calcul ont été obtenus sur un processeur Intel Core i7 cadencé à 2.9 GHz. De plus, aucun traitement parallèle ou sur GPU n'a été implémenté dans ce projet.

Les temps de calcul de l'approche proposée ont été comparés avec les temps de calcul nécessaires pour générer une simulation en haute résolution. Les simulations en haute résolution ont été générées avec le simulateur SPH développé par l'équipe de recherche et développement de Mokko Studio. Une simulation FLIP générée par un simulateur plus performant tel que celui de *Houdini* permettrait d'obtenir de meilleurs temps. Cela permettrait de réduire le temps par trame par un facteur de 10. Les temps de calcul présentés dans le tableau 4.6 sont des moyennes par trame.

Tableau 4.6 Comparaison entre les temps de calcul nécessaires à la génération d'une simulation de fluide et la génération des points de surface par notre approche.

Scénario	Nombre de particules	Simulation en haute résolution	Notre approche
Goutte	378 535	30 s	0.7 s
Gravité	515 605	39 s	1.2 s
Barrage	734 200	46 s	3.8 s

CHAPITRE 5

RÉSULTATS

L'approche proposée dans ce document a été validée à l'aide de trois scénarios offrant différentes conditions de simulation. Ce chapitre présente les résultats obtenus suite aux expérimentations avec notre approche. Toutes les simulations utilisées pour l'entraînement du dictionnaire sont des variantes du scénario à reconstruire. De plus, le nombre de trames utilisées à la phase d'apprentissage pour chaque scénario présenté dans ce chapitre est précisé à la section 4.6.

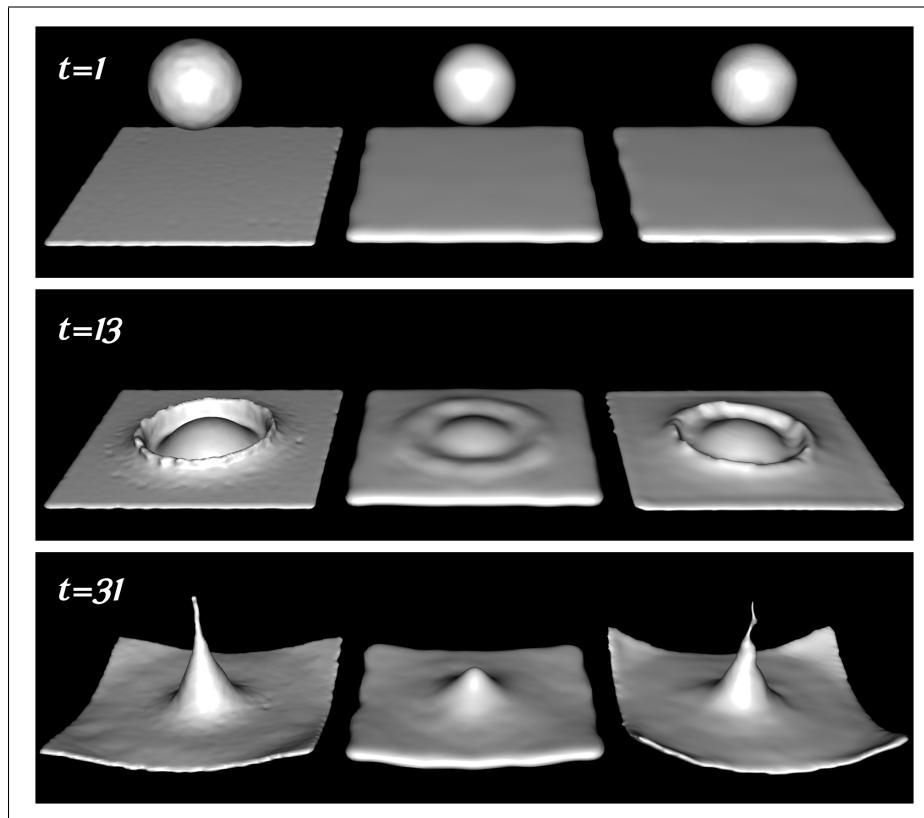


Figure 5.1 Scénario d'une goutte tombant dans un réservoir de liquide. L'objectif est de générer une surface similaire (droite) à la simulation de référence en haute résolution (gauche) à partir d'une simulation en basse résolution (milieu).

Les figures 5.1, 5.2 et 5.3 présentent respectivement de la gauche vers la droite, la surface de la simulation en haute résolution, la surface de la simulation en basse résolution et la surface de la simulation reconstruite avec notre approche. Il est à noter que la simulation en haute résolution (à gauche) est la cible que nous souhaitons atteindre avec notre approche. De plus, ce scénario de simulation n'est pas utilisé lors de l'entraînement afin de ne pas biaiser l'apprentissage du dictionnaire. En résumé, l'approche nous permet de reconstruire une goutte de fluide qui tombe dans un réservoir de fluide à partir de différents scénarios d'entraînement qui simule une goutte tombant à différents endroits.

D'ailleurs, nous tenons à souligner que toutes les simulations ont été stabilisées avant d'être utilisées pour l'entraînement. Environ 300 trames ont été nécessaires afin d'obtenir un liquide immobile. Par exemple, dans le scénario présenté à la figure 5.1, le liquide du réservoir a été stabilisé avant de laisser tomber la goutte. La figure 5.1 présente une expérimentation de l'approche à partir d'un scénario d'une goutte tombant dans un réservoir de liquide.

À la figure 5.2, nous testons l'approche sur un autre type de scénario. Dans ce scénario, le fluide est exposé à des changements périodiques de la force gravitationnelle. Ces changements périodiques sont effectués sur les deux axes du plan XY (frontière de dessous). Voici comment est déterminée la force gravitationnelle :

$$\vec{g} = \begin{cases} (x, y, 0) & \text{si } t \text{ est pair} \\ (0, y, x) & \text{si } t \text{ est impair} \end{cases} \quad (5.1)$$

où

$$x = -\frac{g_0}{3} \sin(cp) \quad (5.2)$$

$$y = g_0 \left| \sin\left(cp - \frac{\Pi}{2}\right) \right| + \frac{g_0}{3} \quad (5.3)$$

Dans l'équation 5.1, t est l'indice de la trame courante ($\{0 \dots N\}$ où N est le nombre de trame). Les termes c , p et g_0 sont respectivement la longueur du cycle (4Π), la période ($2\frac{t}{3}$) et la norme de la force gravitationnelle à l'initialisation (9.81). Le terme g_0 est en fait l'amplitude de la fonction. Ce changement de la force gravitationnelle est équivalent à pivoter périodi-

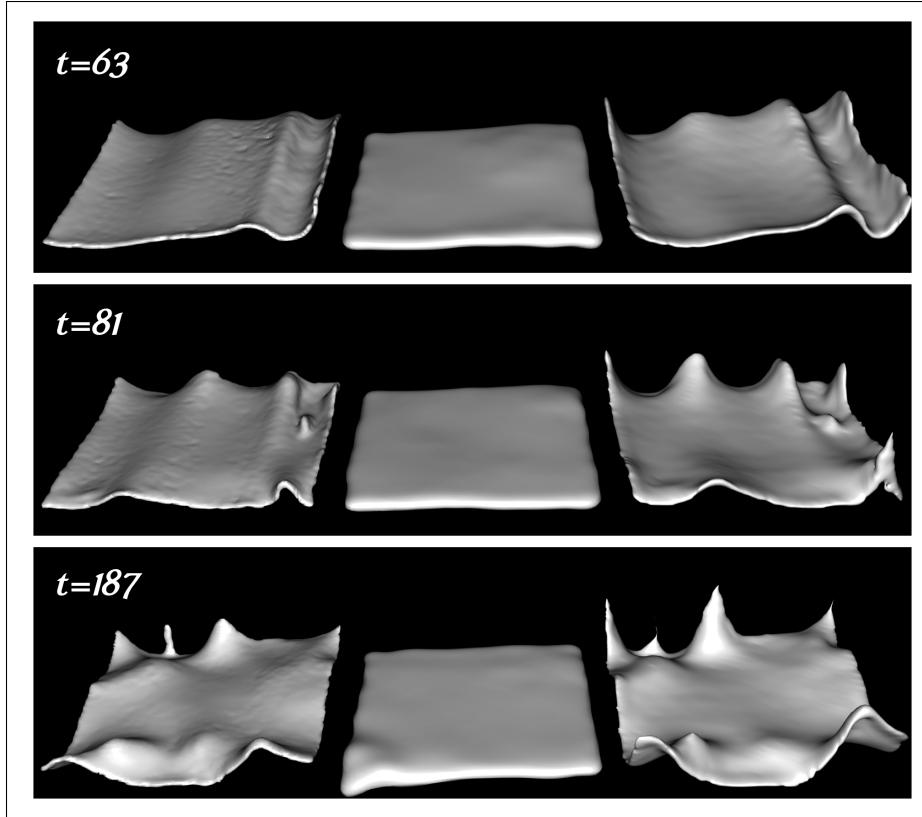


Figure 5.2 Scénario d'un changement périodique des forces gravitationnelles.

quement sur deux axes le contenant (déterminé par les frontières) du fluide. Dans un scénario moins symétrique que celui présenté à la figure 5.1, la solution finale des points de surface générés par notre approche expose quelques légères différences avec la surface de référence. Ces différences sont causées par le fait que les différences sont tellement fortes entre les deux résolutions, que la solution de la minimisation est presque entièrement basée sur les propriétés physiques par parcelle.

Cette dernière observation est aussi valable pour le scénario présenté à la figure 5.3. Toutefois, les différences entre l'apparence de la surface de la simulation en haute résolution et la surface de la simulation en basse résolution sont moins grandes qu'avec le scénario présenté à la figure 5.2.

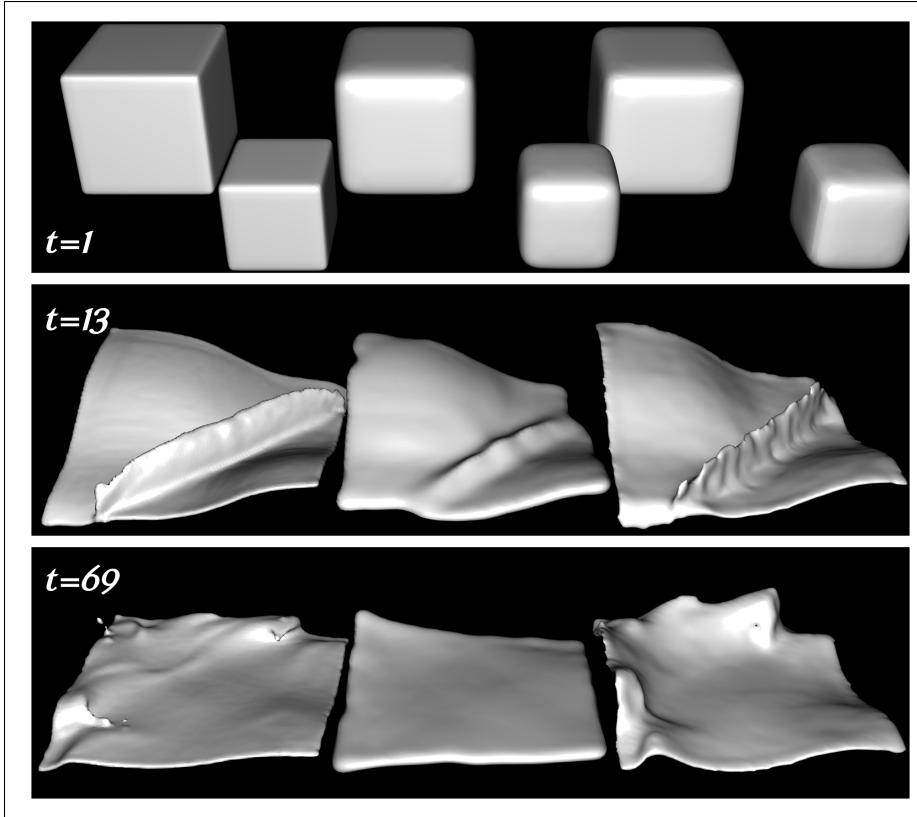


Figure 5.3 Scénario d'une rupture de barrage.

Lorsque la méthode n'arrive pas à converger vers une solution, les particules de surface de la simulation en basse résolution sont conservées. À la première trame ($t = 1$) de la figure 5.1, la surface de la simulation reconstruite par notre approche (droite) est identique à la surface de la simulation en basse résolution (milieu). Cette situation survient lorsque la simulation de fluide est divisée en plusieurs morceaux (scénario de la goutte). La même situation est à prévoir dans un cas plus agité du scénario présenté à la figure 5.3. Actuellement, notre représentation du dictionnaire ne permet pas de reconstruire dans un domaine volumique. La figure 5.4 présente le résultat lorsque l'on force la méthode à déterminer une solution pour ce genre de situation. L'approche tente de trouver des parcelles permettant de joindre les deux portions de fluide.

La figure 5.4 explique la situation introduite précédemment lorsque la solution de minimisation ne converge pas. Cette figure présente le résultat de l'approche lorsque nous forçons la minimisation à déterminer des parcelles pour les sections entre les portions de fluide détachées. Le

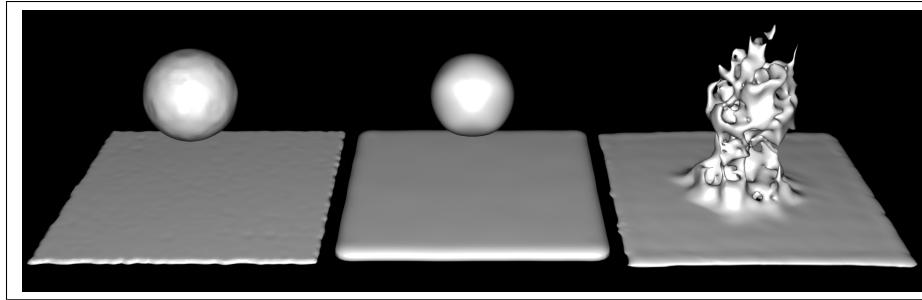


Figure 5.4 Problème de reconstruction lorsqu'une portion du fluide est détachée du reste.

résultat est plutôt artistique, mais ne présente en aucun point une cohérence avec la solution exacte (à gauche).

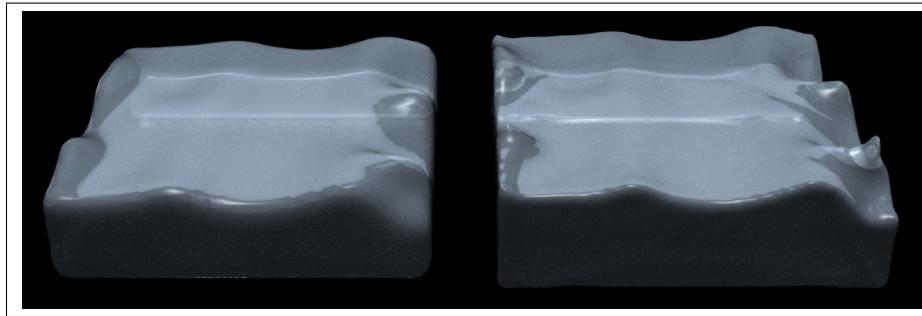


Figure 5.5 Le rendu d'une trame de simulation liquide avec nuanceur (à gauche la référence et à droite la reconstruite).

La figure 5.5 est une comparaison entre la simulation en haute résolution et la simulation en basse résolution (scénario de la gravité périodique) reconstruite par notre approche. Nous avons ajouté l'effet d'un nuanceur de liquide sur les simulations pour présenter un aperçu des résultats possibles avec l'approche proposée dans ce document. De plus, ce résultat démontre que le résultat final de la reconstruction est sans artefact apparent à l'étape de rendu. Le nuanceur utilisé ne fait qu'ajuster la transparence et la réfraction du liquide par rapport aux sources de lumière dans la scène.

L'observation de la courbure (figure 5.6) permet aussi de valider que les surfaces reconstruites (à droite) contiennent des valeurs équivalentes à celles de référence (à gauche). La différence

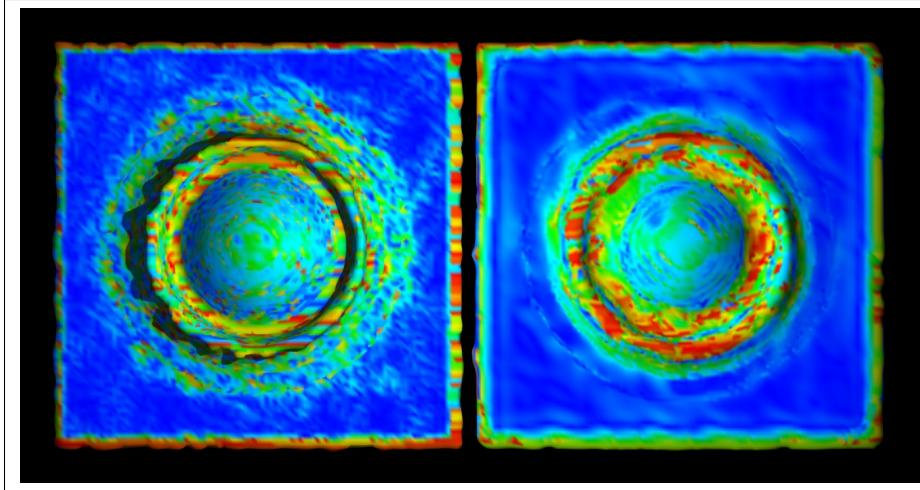


Figure 5.6 Comparaison des valeurs de courbure entre la simulation en haute résolution et celle reconstruite.

significative est plus perceptible dans les basses fréquences puisqu'un lissage a été fait (section 3.3.2) sur l'ensemble des points de la surface reconstruite.

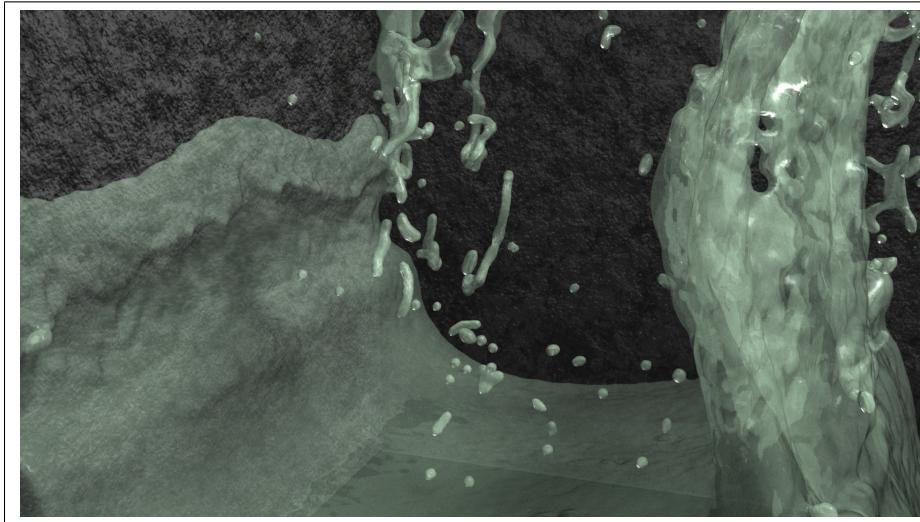


Figure 5.7 Scène actuellement impossible à reconstruire avec l'approche proposée.

Finalement, la figure 5.7 est une preuve et un rappel que l'approche peut encore subir d'importantes améliorations. Avec la formulation actuelle de l'approche proposée, ce scénario ne peut

pas être reconstruit. L'ajout d'éclaboussures et de gouttelettes est un cas spécifique qui peut faire l'objet d'un projet de recherche en soi.

Ces résultats démontrent que l'objectif principal a été atteint. La reconstruction d'une surface en haute résolution à l'aide d'un dictionnaire et basée uniquement sur les particules d'une simulation en basse résolution est possible. Bien que l'approche présente d'importantes limitations au niveau des scénarios pouvant être gérés, les résultats obtenus sont plutôt convaincants et prometteurs pour de nouveaux projets de recherche dans cette direction.

CHAPITRE 6

DISCUSSION

Ce chapitre présente une revue de l'approche proposée dans ce document. À la section 6.1, un résumé des avantages de la technique est présenté. De plus, les limitations de l'approche sont exposées à la section 6.2.

6.1 Avantages de l'approche

L'approche présentée dans le cadre de ce projet de recherche comporte plusieurs avantages. Tout d'abord, bien que l'étape d'entraînement soit la plus coûteuse de l'approche, le résultat de celle-ci n'est qu'à accomplir une seule fois pour être réutilisé sans contrainte. Le dictionnaire couplé (concaténation de dictionnaires par propriété) généré à l'apprentissage contient des atomes permettant d'établir une correspondance entre les caractéristiques de la surface en hauteur résolution et les propriétés physiques des particules d'une simulation en basse résolution. La combinaison linéaire de ces atomes rend possible la reproduction d'un grand éventail de parcelles de fluide.

Ensuite, la formulation du problème par dictionnaires de propriétés permet la modulation de l'approche. Cela signifie que la méthode est facilement adaptable aux changements de propriétés observées. De plus, l'analyse en composantes principales permet d'étendre le nombre de propriétés possiblement observable. D'ailleurs, l'approche itérative de mise à jour des dictionnaires concaténés permet aussi d'envisager une parallélisation de l'implémentation.

Un autre aspect intéressant de l'approche est le fait que l'apparence globale de la surface résultante soit similaire à celle de la simulation en haute résolution. De plus, cette nouvelle surface offre des différences (par rapport à la surface en haute résolution) qui malgré tout respectent la cohérence spatiale et la cohérence temporelle de la simulation en basse résolution utilisée en entrée à l'approche. Il s'agit d'un résultat qui ne peut pas être obtenu par une approche procédurale d'interpolation entre deux liquides (Raveendran *et al.*, 2014). Finalement, notre

méthode peut être complémentaire à plusieurs approches de synthèse de détails de l'état de l'art (Narain *et al.*, 2008; Kim *et al.*, 2013). Ces approches peuvent ajouter des détails à petites échelles localement sur la nouvelle surface en haute résolution générée par notre approche. De cette façon, nous offrons la possibilité de détailler les effets de la turbulence sur un volume de surface physiquement plausible.

6.2 Limitations

La technique décrite dans ce document comporte un certain nombre de limitations. Tout d'abord, la méthode utilisée pour identifier les particules à la surface du fluide n'est pas infaillible. Certains scénarios présentent une complexité trop importante pour efficacement extraire les particules à la surface libre du fluide. Il en est de même pour les parcelles de fluide de la simulation en basse résolution utilisée à l'entrée. De plus, l'approche de division du domaine de la simulation utilisée pour extraire les parcelles de fluide à l' entraînement est très limitative. Cette technique ne supporte pas la reconstruction de parcelles où il y a une occlusion possible. Par exemple, dans le cas d'une goutte, la portion de fluide sous la goutte ne sera pas considérée. Des expérimentations à ce sujet devront être effectuées afin de faire ressortir des métriques à respecter pour éviter les artéfacts de reconstruction causés par cette limitation.

Il existe une seconde limitation à ne pas négliger lorsque cette approche est utilisée pour reconstruire une nouvelle surface. La simulation en basse résolution dont la surface est à reconstruire doit être suffisamment similaire aux simulations utilisées pour entraîner le dictionnaire. La reconstruction d'un scénario complètement différent de ceux utilisés pour générer le dictionnaire n'offre aucune garantie que les propriétés physiques de la surface résultante seront conservées. D'ailleurs, les correspondances établies à l'apprentissage sont dépendantes des relations existantes entre les propriétés des particules de la simulation en haute résolution et les propriétés observées des particules de la simulation en basse résolution utilisée en entrée.

La dernière limitation évidente de l'approche est l'échelle de reconstruction. Actuellement, l'échelle de reconstruction des parcelles est contrainte à l'échelle des parcelles utilisées lors de

la phase d'apprentissage. Ceci dit, plus l'échelle de la simulation en basse résolution utilisée en entrée est similaire à l'échelle des atomes du dictionnaire et plus la qualité de reconstruction sera bonne. Actuellement, il nous est impossible de reconstruire des parcelles de résolution plus ou moins fine que celles encodées dans le dictionnaire. Il serait possible d'utiliser une approche similaire à la méthode proposée par Yang *et al.* (2010) et d'ajuster l'échelle des parcelles en relation avec la correspondance d'un dictionnaire en basse résolution et d'un dictionnaire en haute résolution. Par ailleurs, il pourrait être intéressant de valider les différences de variation des propriétés physiques entre les différentes échelles.

CHAPITRE 7

TRAVAUX FUTURS

Tel que mentionné à la section 6.2, l’approche proposée comporte deux importantes limitations. La première limitation concerne la qualité d’identification et de modélisation des parcelles à la surface du fluide. Puisque les parcelles sont composées à l’aide des particules de surface, cette limitation est nécessairement influencée par la précision de la technique utilisée afin d’identifier ces particules. Donc, dans un premier temps, cette partie de l’approche doit être améliorée. La méthode du *color field* est particulièrement inefficace pour les portions de fluide où le nombre de particules avoisinantes est faible. L’utilisation d’un champ de distance sur l’ensemble du volume permettrait d’obtenir de meilleurs résultats (Goswami *et al.*, 2010), et ce, particulièrement pour ces cas spécifiques. Par le fait même, notre approche pourrait s’abstraire de la grille utilisée sur plan horizontal. D’ailleurs, les temps de calcul de la génération du champ de distances scalaires sont comparables aux temps nécessaires pour calculer le gradient de la normale avec la méthode du *color field*. De plus, à l’aide de ce champ de distances, nous pourrions être en mesure d’améliorer la qualité d’extraction des parcelles en rendant possible l’identification des portions de fluide victime d’occlusion.

D’une autre part, la correspondance faite à l’étape d’apprentissage entre les propriétés géométriques et physiques du fluide est aussi une grande limitation de l’approche. Le choix des propriétés observées limite la flexibilité d’un dictionnaire sur l’étape de reconstruction. Ces observations sont basées sur des forces physiques qui pourraient être traduites en termes simplifiés afin de généraliser leurs interprétations. Par exemple, des descripteurs de premier et de deuxième ordre peuvent très bien décrire ces forces.

Dans un second temps, ce projet ouvre naturellement des portes vers d’autres travaux de recherche. Ces travaux futurs sont une suite logique et intuitive des travaux présentés dans ce projet de recherche. Tout d’abord, l’utilisation d’un dictionnaire multirésolution nous semble appropriée. L’apprentissage permettrait de conserver des détails basés sur des propriétés observées à grande échelle et à petite échelle. De plus, la reconstruction de la surface par le

problème d'optimisation des points de surface serait faite en plusieurs passes. La première passe permettrait d'obtenir une estimation de l'apparence de la surface reconstruite basée sur quelques parcelles. Chacune de ces parcelles serait ensuite divisée à nouveau et ainsi de suite jusqu'à ce que le niveau de détail désiré soit atteint.

Ensuite, le choix de l'apprentissage par dictionnaire a été fait suite à un certain nombre d'expérimentations non concluantes avec les SVM. Néanmoins, un certain nombre de travaux de recherche en dynamique computationnelle des fluides ont tiré avantage de l'apprentissage automatique par perceptron multicouche. Cette méthode d'apprentissage permet de capturer un niveau de précision impressionnant dans les écoulements turbulents par induction (Müller *et al.*, 1999). Avec une telle méthode, il serait possible de gérer les scénarios incluant des obstacles statiques et dynamiques.

Par ailleurs, d'autres pistes et idées sont à envisager. Par exemple, effectuer l'analyse et l'apprentissage des propriétés observées dans le domaine fréquentiel. Nous pourrions observer les différences entre le spectre fréquentiel (par la transformée de Fourier) d'une simulation en haute résolution et cette même simulation en basse résolution. La différence entre le spectre de la simulation en haute résolution et celui de la simulation en basse résolution permettrait d'avoir de l'information fréquentielle sur l'ajout des hautes fréquences sur une simulation avec peu de particules.

Finalement, nous avons présenté à la section 6.1 la possibilité d'utiliser l'approche proposée avec une méthode de synthèse de turbulence procédurale. Ce type d'approche peut être complémentaire au problème de minimisation des points de surface. En résumé, le concept est de générer un bruit sur la solution α_i (équation 3.4) présentée à la section 3.3.1. Ce bruit peut être généré à l'aide d'une fonction de turbulence (Bridson *et al.*, 2007) basée sur le champ de vitesse de la simulation en basse résolution utilisée en entrée.

CONCLUSION

Dans le cadre du projet de recherche présenté dans ce mémoire, une approche permettant de modéliser une surface basée sur les propriétés géométriques et physiques d'exemples de fluide en haute résolution a été générée. Cette méthode ne nécessite que quelques milliers de particules pour obtenir des résultats intéressants. Lorsque le nombre de particules est insuffisant, une technique de suréchantillonnage par disques de Poisson est utilisée. L'approche proposée utilise une méthode d'apprentissage par dictionnaire pour entraîner des atomes qui permettront de retrouver l'apparence en surface d'un fluide en haute résolution. Les atomes sont composés des propriétés observées pendant la phase d'entraînement. Chaque propriété génère un dictionnaire basé sur les échantillons sélectionnés. À la toute fin de l'étape d'apprentissage, un dictionnaire couplé est formé à partir des multiples propriétés observées.

En ce qui concerne cette approche, seulement la hauteur, l'énergie cinétique et la vorticité des particules sont observées lors de l'entraînement. Naturellement, ces propriétés sont utilisées pour sélectionner les parcelles de fluide qui seront à reconstruire. La génération des nouvelles particules de surface est formulée à l'aide d'un problème de minimisation. Les termes de contraintes de la minimisation sont essentiellement basés sur les propriétés géométriques et physiques. De plus, d'autres termes sont utilisés pour générer des points de surface qui respectent la cohérence spatiale et temporelle à partir du résultat de la minimisation.

Cette approche peut être utilisée à deux fins en production. La première utilisation est au niveau de la génération d'un fluide en haute résolution. L'approche proposée est complémentaire aux techniques procédurales qui ont été développées dans les dernières années pour ajouter des détails à la surface d'un fluide à faible résolution. Toutefois, notre méthode permet de générer ces détails fins sur une surface basée sur l'apparence globale d'un fluide en haute résolution. Par conséquent, la solution finale est plus plausible (comparativement à la simulation de référence en haute résolution). La deuxième avenue à cette approche est par la prévisualisation utilisée dans les outils de modélisation. Notre approche permet de générer rapidement une estimation de l'apparence de la surface d'un fluide en haute résolution à partir des exemples de simulations qui ont été utilisés pour générer le dictionnaire à l'apprentissage.

Enfin, les résultats obtenus démontrent que l'apparence à grande échelle peut être obtenue par l'apprentissage sur un ensemble de relations entre les caractéristiques géométriques et les propriétés physiques des particules de surface. L'hypothèse de l'approche est donc supportée par ces résultats.

BIBLIOGRAPHIE

- Adami, S., X. Hu, et N. Adams. 2010. « A conservative SPH method for surfactant dynamics ». *Journal of Computational Physics*, vol. 229, n° 5, p. 1909–1926.
- Aharon, M. et M. Elad. 2008. « Sparse and redundant modeling of image content using an image-signature-dictionary ». *SIAM Journal on Imaging Sciences*, vol. 1, n° 3, p. 228–247.
- Aharon, M., M. Elad, et A. Bruckstein. 2006. « K-SVD : An algorithm for designing overcomplete dictionaries for sparse representation ». *Signal Processing, IEEE Transactions on*, vol. 54, n° 11, p. 4311–4322.
- Angelidis, A. et F. Neyret. 2005. « Simulation of smoke based on vortex filament primitives ». In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*. p. 87–96. ACM.
- Angelidis, A., F. Neyret, K. Singh, et D. Nowrouzezahrai. 2006. « A controllable, fast and stable basis for vortex based smoke simulation ». In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*. p. 25–32. Eurographics Association.
- Bargteil, A. W., T. G. Goktekin, J. F. O'brien, et J. A. Strain. 2006. « A semi-Lagrangian contouring method for fluid simulation ». *ACM Transactions on Graphics (TOG)*, vol. 25, n° 1, p. 19–38.
- Barnat, A. et N. S. Pollard. 2012. « Smoke sheets for graph-structured vortex filaments ». In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. p. 77–86. Eurographics Association.
- Batty, C., F. Bertails, et R. Bridson. 2007. « A fast variational framework for accurate solid-fluid coupling ». *ACM Transactions on Graphics (TOG)*, vol. 26, n° 3, p. 100.
- Beale, J. T. et A. Majda. 1982. « Vortex methods. I. Convergence in three dimensions ». *Mathematics of Computation*, vol. 39, n° 159, p. 1–27.
- Bertalmio, M., G. Sapiro, V. Caselles, et C. Ballester. 2000. « Image inpainting ». In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. p. 417–424. ACM Press/Addison-Wesley Publishing Co.
- Bertalmio, M., L. Vese, G. Sapiro, et S. Osher. 2003. « Simultaneous structure and texture image inpainting ». *Image Processing, IEEE Transactions on*, vol. 12, n° 8, p. 882–889.
- Bousquet, O. et L. Bottou. 2008. « The tradeoffs of large scale learning ». In *Advances in neural information processing systems*. p. 161–168.

- Brackbill, J. et H. Ruppel. 1986. « FLIP : A method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions ». *Journal of Computational Physics*, vol. 65, n° 2, p. 314–343.
- Bradley, D. M. et J. A. Bagnell. 2008. « Differential sparse coding ».
- Bridson, R. 2007. « Fast Poisson disk sampling in arbitrary dimensions ». In *ACM SIGGRAPH*. p. 5.
- Bridson, R., 2008. *Fluid Simulation for Computer Graphics*. A K Peters Series.
- Bridson, R., J. Hourihane, et M. Nordenstam. 2007. « Curl-noise for procedural fluid flow ». In *ACM Transactions on Graphics (TOG)*. p. 46. ACM.
- Brochu, T. et R. Bridson. 2009. « Animating smoke as a surface ». *Posters and Demos*, vol. 2.
- Brochu, T., C. Batty, et R. Bridson. 2010. « Matching fluid simulation elements to surface geometry and topology ». *ACM Transactions on Graphics (TOG)*, vol. 29, n° 4, p. 47.
- Brochu, T., T. Keeler, et R. Bridson. 2012. « Linear-time smoke animation with vortex sheet meshes ». In *Proceedings of the 11th ACM SIGGRAPH/Eurographics conference on Computer Animation*. p. 87–95. Eurographics Association.
- Burger, H. C., C. J. Schuler, et S. Harmeling. 2012. « Image denoising : Can plain Neural Networks compete with BM3D ? ». In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. p. 2392–2399. IEEE.
- Carlson, M., P. J. Mucha, et G. Turk. 2004. « Rigid fluid : animating the interplay between rigid bodies and fluid ». In *ACM Transactions on Graphics (TOG)*. p. 377–384. ACM.
- Catté, F., P.-L. Lions, J.-M. Morel, et T. Coll. 1992. « Image selective smoothing and edge detection by nonlinear diffusion ». *SIAM Journal on Numerical analysis*, vol. 29, n° 1, p. 182–193.
- Chang, C.-C. et C.-J. Lin. 2011. « LIBSVM : a library for support vector machines ». *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, n° 3, p. 27.
- Chen, D., M. S. Hassouna, A. A. Farag, et R. Falk. 2007. « A new color coding scheme for easy polyp visualization in CT-based virtual colonoscopy ». In *Medical Imaging*. p. 65142I–65142I. International Society for Optics and Photonics.
- Chen, F., Y. Zhao, et Z. Yuan. 2011. « Langevin particle : A self-adaptive lagrangian primitive for flow simulation enhancement ». In *Computer Graphics Forum*. p. 435–444. Wiley Online Library.
- Chen, S. S., D. L. Donoho, et M. A. Saunders. 1998. « Atomic decomposition by basis pursuit ». *SIAM journal on scientific computing*, vol. 20, n° 1, p. 33–61.

- Chentanez, N. et M. Müller. 2011. « Real-time Eulerian water simulation using a restricted tall cell grid ». In *ACM Transactions on Graphics (TOG)*. p. 82. ACM.
- Cortes, C. et V. Vapnik. 1995. « Support-vector networks ». *Machine learning*, vol. 20, n° 3, p. 273–297.
- Davidson, P. A., 2004. *Turbulence : An Introduction for Scientists and Engineers*.
- Donoho, D. L. 1995. « De-noising by soft-thresholding ». *Information Theory, IEEE Transactions on*, vol. 41, n° 3, p. 613–627.
- Drori, I., D. Cohen-Or, et H. Yeshurun. 2003. « Fragment-based image completion ». In *ACM Transactions on Graphics (TOG)*. p. 303–312. ACM.
- Duda, R. O., P. E. Hart, et D. G. Stork, 2012. *Pattern classification*.
- Efros, A. A. et T. K. Leung. 1999. « Texture synthesis by non-parametric sampling ». In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*. p. 1033–1038. IEEE.
- Elad, M. et M. Aharon. 2006. « Image denoising via sparse and redundant representations over learned dictionaries ». *Image Processing, IEEE Transactions on*, vol. 15, n° 12, p. 3736–3745.
- Enright, D., R. Fedkiw, J. Ferziger, et I. Mitchell. 2002. « A hybrid particle level set method for improved interface capturing ». *Journal of Computational Physics*, vol. 183, n° 1, p. 83–116.
- Fedkiw, R., J. Stam, et H. W. Jensen. 2001. « Visual simulation of smoke ». In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. p. 15–22. ACM.
- Goktekin, T. G., A. W. Bargteil, et J. F. O’Brien. 2004. « A method for animating viscoelastic fluids ». In *ACM Transactions on Graphics (TOG)*. p. 463–468. ACM.
- Goswami, P., P. Schlegel, B. Solenthaler, et R. Pajarola. 2010. « Interactive SPH simulation and rendering on the GPU ». In *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. p. 55–64. Eurographics Association.
- Guendelman, E., R. Bridson, et R. Fedkiw. 2003. « Nonconvex rigid bodies with stacking ». In *ACM Transactions on Graphics (TOG)*. p. 871–878. ACM.
- Guennebaud, G., B. Jacob, et al. 2010. « Eigen v3 ». <http://eigen.tuxfamily.org>.
- Harlow, F. H. 1962. *The particle-in-cell method for numerical solution of problems in fluid dynamics*. Technical report.
- Harlow, F. H., J. E. Welch, et al. 1965. « Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface ». *Physics of fluids*, vol. 8, n° 12, p. 2182.

- Heeger, D. J. et J. R. Bergen. 1995. « Pyramid-based texture analysis/synthesis ». In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. p. 229–238. ACM.
- Hirani, A. N. et T. Totsuka. 1996. « Combining frequency and spatial domain information for fast interactive image noise removal ». In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. p. 269–276. ACM.
- Hong, J.-M. et C.-H. Kim. 2003. « Animation of bubbles in liquid ». In *Computer Graphics Forum*. p. 253–262. Wiley Online Library.
- Hung, J. C., C.-H. Hwang, Y.-C. Liao, N. C. Tang, et T.-J. Chen. 2008. « Exemplar-based image inpainting base on structure construction ». *Journal of Software*, vol. 3, n° 8, p. 57–64.
- Irving, G., E. Guendelman, F. Losasso, et R. Fedkiw. 2006. « Efficient simulation of large bodies of water by coupling two and three dimensional techniques ». *ACM Transactions on Graphics (TOG)*, vol. 25, n° 3, p. 805–811.
- Jang, T., H. Kim, J. Bae, J. Seo, et J. Noh. 2010. « Multilevel vorticity confinement for water turbulence simulation ». *The Visual Computer*, vol. 26, n° 6-8, p. 873–881.
- Jia, J. et C.-K. Tang. 2003. « Image repairing : Robust image synthesis by adaptive nd tensor voting ». In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*. p. I–643. IEEE.
- Kim, B., Y. Liu, I. Llamas, et J. Rossignac. 2005. « Flowfixer : Using bfecc for fluid simulation ». In *proceedings of the First Eurographics conference on Natural Phenomena*. p. 51–56. Eurographics Association.
- Kim, D., O.-y. Song, et H.-S. Ko. 2008a. « A semi-lagrangian CIP fluid solver without dimensional splitting ». In *Computer Graphics Forum*. p. 467–475. Wiley Online Library.
- Kim, D., O.-y. Song, et H.-S. Ko. 2009. « Stretching and wiggling liquids ». In *ACM Transactions on Graphics (TOG)*. p. 120. ACM.
- Kim, D., S. W. Lee, O.-y. Song, et H.-S. Ko. 2012. « Baroclinic turbulence with varying density and temperature ». *Visualization and Computer Graphics, IEEE Transactions on*, vol. 18, n° 9, p. 1488–1495.
- Kim, T., N. Thürey, D. James, et M. Gross. 2008b. « Wavelet turbulence for fluid simulation ». In *ACM Transactions on Graphics (TOG)*. p. 50. ACM.
- Kim, T., J. Tessendorf, et N. Thuerey. 2013. « Closest point turbulence for liquid surfaces ». *ACM Transactions on Graphics (TOG)*, vol. 32, n° 2, p. 15.
- Klingner, B. M., B. E. Feldman, N. Chentanez, et J. F. O'brien. 2006. « Fluid animation with dynamic meshes ». In *ACM Transactions on Graphics (TOG)*. p. 820–825. ACM.

- Komodakis, N. et G. Tziritas. 2007. « Image completion using efficient belief propagation via priority scheduling and dynamic pruning ». *Image Processing, IEEE Transactions on*, vol. 16, n° 11, p. 2649–2661.
- Kushner, H. J. et G. Yin, 2003. *Stochastic approximation and recursive algorithms and applications*, volume 35.
- Lamorlette, A. et N. Foster. 2002. « Structural modeling of flames for a production environment ». *ACM Transactions on Graphics (TOG)*, vol. 21, n° 3, p. 729–735.
- Lee, H., A. Battle, R. Raina, et A. Y. Ng. 2006. « Efficient sparse coding algorithms ». In *Advances in neural information processing systems*. p. 801–808.
- Lindenbaum, M., M. Fischer, et A. Bruckstein. 1994. « On Gabor's contribution to image enhancement ». *Pattern Recognition*, vol. 27, n° 1, p. 1–8.
- Longuet-Higgins, M. S. 1953. « Mass transport in water waves ». *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, vol. 245, n° 903, p. 535–581.
- Longuet-Higgins, M. S. 1998. « Vorticity and curvature at a free surface ». *Journal of Fluid Mechanics*, vol. 356, p. 149–153.
- Losasso, F., F. Gibou, et R. Fedkiw. 2004. « Simulating water and smoke with an octree data structure ». *ACM Transactions on Graphics (TOG)*, vol. 23, n° 3, p. 457–462.
- Macklin, M. et M. Müller. 2013. « Position based fluids ». *ACM Transactions on Graphics (TOG)*, vol. 32, n° 4, p. 104.
- Mairal, J., F. Bach, J. Ponce, et G. Sapiro. 2009a. « Online dictionary learning for sparse coding ». In *Proceedings of the 26th Annual International Conference on Machine Learning*. p. 689–696. ACM.
- Mairal, J., F. Bach, J. Ponce, G. Sapiro, et A. Zisserman. 2009b. « Non-local sparse models for image restoration ». In *Computer Vision, 2009 IEEE 12th International Conference on*. p. 2272–2279. IEEE.
- Mairal, J., F. Bach, J. Ponce, et G. Sapiro. 2010. « Online learning for matrix factorization and sparse coding ». *The Journal of Machine Learning Research*, vol. 11, p. 19–60.
- Mallat, S., 1999. *A wavelet tour of signal processing*.
- Marshall, J. et J. Grant. 1996. « Penetration of a blade into a vortex core : vorticity response and unsteady blade forces ». *Journal of Fluid Mechanics*, vol. 306, p. 83–109.
- Masnou, S. et J.-M. Morel. 1998. « Level lines based disocclusion ». In *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*. p. 259–263. IEEE.

- Morris, J. P. 2000. « Simulating surface tension with smoothed particle hydrodynamics ». *International Journal for Numerical Methods in Fluids*, vol. 33, n° 3, p. 333–353.
- Mullen, P., K. Crane, D. Pavlov, Y. Tong, et M. Desbrun. 2009. « Energy-preserving integrators for fluid animation ». In *ACM Transactions on Graphics (TOG)*. p. 38. ACM.
- Müller, M., D. Charypar, et M. Gross. 2003. « Particle-based fluid simulation for interactive applications ». In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*. p. 154–159. Eurographics Association.
- Müller, M., B. Solenthaler, R. Keiser, et M. Gross. 2005. « Particle-based fluid-fluid interaction ». In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*. p. 237–244. ACM.
- Müller, S., M. Milano, et P. Koumoutsakos. 1999. « Application of machine learning algorithms to flow modeling and optimization ». *Annual Research Briefs*, p. 169–178.
- Narain, R., J. Sewall, M. Carlson, et M. C. Lin. 2008. « Fast animation of turbulence using energy transport and procedural synthesis ». In *ACM Transactions on Graphics (TOG)*. p. 166. ACM.
- Navier, C.-L. 1823. « Mémoire sur les lois du mouvement des fluides ». *Mémoires de l'Academie Royale des Sciences de l'Institut de France*, vol. 6, p. 389–440.
- Novikov, E. 1983. « Generalized dynamics of three-dimensional vortex singularities/vortons ». *Zhurnal Eksperimentalnoi i Teoreticheskoi Fiziki*, vol. 84, p. 975–981.
- Orthmann, J., H. Hochstetter, J. Bader, S. Bayraktar, et A. Kolb. 2013. « Consistent surface model for SPH-based fluid transport ». In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. p. 95–103. ACM.
- Pan, Z., J. Huang, Y. Tong, et H. Bao. 2012. « Wake synthesis for shallow water equation ». In *Computer Graphics Forum*. p. 2029–2036. Wiley Online Library.
- Perona, P. et J. Malik. 1990. « Scale-space and edge detection using anisotropic diffusion ». *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 12, n° 7, p. 629–639.
- Peyré, G. 2009. « Sparse modeling of textures ». *Journal of Mathematical Imaging and Vision*, vol. 34, n° 1, p. 17–31.
- Pfaff, T., N. Thuerey, A. Selle, et M. Gross. 2009. « Synthetic turbulence using artificial boundary layers ». In *ACM Transactions on Graphics (TOG)*. p. 121. ACM.
- Pfaff, T., N. Thuerey, et M. Gross. 2012. « Lagrangian vortex sheets for animating fluids ». *ACM Transactions on Graphics (TOG)*, vol. 31, n° 4, p. 112.
- Pope, S. B., 2000. *Turbulent flows*.

- Rasmussen, N., D. Q. Nguyen, W. Geiger, et R. Fedkiw. 2003. « Smoke simulation for large scale phenomena ». *ACM Transactions on Graphics (TOG)*, vol. 22, n° 3, p. 703–707.
- Raveendran, K., C. Wojtan, N. Thuerey, et G. Turk. 2014. « Blending liquids ». *ACM Transactions on Graphics (TOG)*.
- Robinson-Mosher, A., T. Shinar, J. Gretarsson, J. Su, et R. Fedkiw. 2008. « Two-way coupling of fluids to rigid and deformable solids and shells ». In *ACM Transactions on Graphics (TOG)*. p. 46. ACM.
- Rosenhead, L. 1931. « The formation of vortices from a surface of discontinuity ». *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, p. 170–192.
- Roth, S. et M. J. Black. 2005. « Fields of experts : A framework for learning image priors ». In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. p. 860–867. IEEE.
- Rudin, L. I., S. Osher, et E. Fatemi. 1992. « Nonlinear total variation based noise removal algorithms ». *Physica D : Nonlinear Phenomena*, vol. 60, n° 1, p. 259–268.
- Ryan, H. 1994. « Ricker, Ormsby, klauder, Butterworth — a choice of wavelets ». *Canadian Society of Exploration Geophysicists (CSEG Record)*, vol. 19, n° 7, p. 8–9.
- Schechter, H. et R. Bridson. 2008. « Evolving sub-grid turbulence for smoke animation ». In *Proceedings of the 2008 ACM SIGGRAPH/Eurographics symposium on Computer animation*. p. 1–7. Eurographics Association.
- Selle, A., N. Rasmussen, et R. Fedkiw. 2005. « A vortex particle method for smoke, water and explosions ». In *ACM Transactions on Graphics (TOG)*. p. 910–914. ACM.
- Selle, A., R. Fedkiw, B. Kim, Y. Liu, et J. Rossignac. 2008. « An unconditionally stable MacCormack method ». *Journal of Scientific Computing*, vol. 35, n° 2-3, p. 350–371.
- Simoncelli, E. P. et J. Portilla. 1998. « Texture characterization via joint statistics of wavelet coefficient magnitudes ». In *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*. p. 62–66. IEEE.
- Solenthaler, B., J. Schläfli, et R. Pajarola. 2007. « A unified particle model for fluid–solid interactions ». *Computer Animation and Virtual Worlds*, vol. 18, n° 1, p. 69–82.
- Stam, J. 1999. « Stable fluids ». In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. p. 121–128. ACM Press/Addison-Wesley Publishing Co.
- Stam, J. et E. Fiume. 1993. « Turbulent wind fields for gaseous phenomena ». In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. p. 369–376. ACM.

- Tessendorf, J. et al. 2001. « Simulating ocean water ». *Simulating Nature : Realistic and Interactive Techniques. SIGGRAPH*, vol. 1.
- Theisel, H., C. Rossi, R. Zayer, et H.-P. Seidel. 2004. « Normal based estimation of the curvature tensor for triangular meshes ». In *Computer Graphics and Applications, 2004. PG 2004. Proceedings. 12th Pacific Conference on*. p. 288–297. IEEE.
- Thürey, N., C. Wojtan, M. Gross, et G. Turk. 2010. « A multiscale approach to mesh-based surface tension flows ». In *ACM Transactions on Graphics (TOG)*. p. 48. ACM.
- Weißmann, S. et U. Pinkall. 2010. « Filament-based smoke with vortex shedding and variational reconnection ». *ACM Transactions on Graphics (TOG)*, vol. 29, n° 4, p. 115.
- Wojtan, C., N. Thürey, M. Gross, et G. Turk. 2010. « Physics-inspired topology changes for thin fluid features ». *ACM Transactions on Graphics (TOG)*, vol. 29, n° 4, p. 50.
- Wong, A. et J. Orchard. 2008. « A nonlocal-means approach to exemplar-based inpainting ». In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*. p. 2600–2603. IEEE.
- Xiong, S., J. Zhang, J. Zheng, J. Cai, et L. Liu. 2014. « Robust surface reconstruction via dictionary learning ». *ACM Transactions on Graphics (TOG)*, vol. 33, n° 6, p. 201.
- Xu, Z. et J. Sun. 2010. « Image inpainting by patch propagation using patch sparsity ». *Image Processing, IEEE Transactions on*, vol. 19, n° 5, p. 1153–1165.
- Yang, J., K. Yu, Y. Gong, et T. Huang. 2009. « Linear spatial pyramid matching using sparse coding for image classification ». In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. p. 1794–1801. IEEE.
- Yang, J., J. Wright, T. S. Huang, et Y. Ma. 2010. « Image super-resolution via sparse representation ». *Image Processing, IEEE Transactions on*, vol. 19, n° 11, p. 2861–2873.
- Yaroslavsky, L. P., 1985. *Digital picture processing : An introduction*, volume 1.
- Yoon, J.-C., H. R. Kam, J.-M. Hong, S. J. Kang, et C.-H. Kim. 2009. « Procedural synthesis using vortex particle method for fluid simulation ». In *Computer Graphics Forum*. p. 1853–1859. Wiley Online Library.
- Zhao, Y., Z. Yuan, et F. Chen. 2010. « Enhancing fluid animation with adaptive, controllable and intermittent turbulence ». In *Proceedings of the 2010 ACM SIGGRAPH/Eurographics symposium on computer animation*. p. 75–84. Eurographics Association.
- Zhu, Y. et R. Bridson. 2005. « Animating sand as a fluid ». In *ACM Transactions on Graphics (TOG)*. p. 965–972. ACM.
- Zoran, D. et Y. Weiss. 2011. « From learning models of natural image patches to whole image restoration ». In *Computer Vision (ICCV), 2011 IEEE International Conference on*. p. 479–486. IEEE.