



HAL
open science

Modèle distribué et asynchrone pour la simulation interactive en mécanique des fluides : vers une plateforme de jeu sérieux pour l'aide à la prise de décision

Mengchen Wang

► To cite this version:

Mengchen Wang. Modèle distribué et asynchrone pour la simulation interactive en mécanique des fluides : vers une plateforme de jeu sérieux pour l'aide à la prise de décision. Modélisation et simulation. Université Paris-Saclay, 2021. Français. NNT : 2021UPASG113 . tel-03609841

HAL Id: tel-03609841

<https://theses.hal.science/tel-03609841>

Submitted on 16 Mar 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Modèle distribué et asynchrone pour la simulation
interactive en mécanique des fluides :
*vers une plateforme de jeu sérieux pour l'aide à la
prise de décision*

*Distributed and asynchronous approach for interactive simulations in fluid
mechanics: towards a serious game platform to support decision making*

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 580, Sciences et technologies de l'information et de la
communication (STIC)

Spécialité de doctorat : Informatique

Unité de recherche : Université Paris-Saclay, CentraleSupélec, Mathématiques et
Informatique pour la Complexité et les Systèmes, 91190, Gif-sur-Yvette, France

Graduate School : Informatique et sciences du numérique

Référent : Faculté des sciences d'Orsay

**Thèse présentée et soutenue à Paris-Saclay,
le 20 décembre 2021, par**

Mengchen WANG

Composition du jury

Samir OTMANE

Professeur, IBISC, Université d'Évry Paris-Saclay

Pierre BOULANGER

Professeur, University of Alberta

Florian DE VUYST

Professeur, Université de Technologie de Compiègne

Bob-Antoine J. MENELAS

Maitre de conférences, Université du Québec à Chicoutimi

Mathieu MURATET

Maitre de conférences, INSHEA, LIP6, Sorbonne Université

Président

Rapporteur & Examineur

Rapporteur & Examineur

Examineur

Examineur

Direction de la thèse

Patrick BOURDOT

Directeur de Recherche, LISN, CNRS, Université Paris-Saclay

Frédéric MAGOULÈS

Professeur, MICS, CentraleSupélec, Université Paris-Saclay

Nicolas FÉREY

Maitre de conférences, LISN, CNRS, Université Paris-Saclay

Directeur de thèse

Co-Directeur de thèse

Co-Encadrant

Titre : Modèle distribué et asynchrone pour la simulation interactive en mécanique des fluides : *vers une plateforme de jeu sérieux pour l'aide à la prise de décision*

Mots clés : Simulation interactive, Jeu sérieux, Simulation de fluide, Méthode de Boltzmann sur réseau, Calcul asynchrone, Réalité Virtuelle

Résumé : C'est en grande partie grâce au jeu sérieux *Fold'It*, que les approches par simulation interactive et par jeu sérieux ont pu convaincre de leur pertinence dans le domaine de la biologie moléculaire. Ce jeu sérieux a été conçu pour adresser de manière ludique et collaborative la problématique du repliement de protéine. Au-delà de l'énorme potentiel pédagogique, un groupe de joueurs de *Fold'It* a réussi le challenge de trouver la structure d'une protéine du virus du VIH, démontrant l'utilité et l'efficacité d'une telle méthodologie centrée utilisateur.

Inspiré par cette avancée méthodologique en biologie moléculaire en matière de simulation interactive, et convaincu de son potentiel pour de nombreuses applications en mécanique des fluides, ce travail de thèse fut consacré à étudier dans quelles mesures et dans quelles conditions une approche par simulation interactive par jeu sérieux pouvait être appliquée au domaine de la mécanique des fluides. La progression des méthodes numériques en mécanique des fluides alliée à la puissance de calcul toujours plus importante permettent en effet d'envisager un transfert méthodologique par la construction d'une plateforme de la simulation des fluides interactives incluant des fonctionnalités de conception de modèles avec conditions initiales, de visualisation et de contrôle interactif du fluide et de son environnement pendant une simulation en cours.

Ce travail de thèse a donc eu pour objectif de concevoir et d'évaluer une approche par simulation interactive et par jeux sérieux consacrés à la mécanique des fluides. La première étape de ce travail fut donc d'implémenter une plateforme dédiée, conjointement à une étude bibliographique continue sur l'usage des simulations interactives et des jeux sérieux en général et en mécanique des fluides. Au-delà de l'aspect technologique et du

challenge sous-jacent, différentes pistes ont été explorées pour aboutir à une architecture matérielle et logicielle très contraintes.

Il s'agit en effet d'offrir les performances requises par l'interactivité, de s'assurer de la conservation de la pertinence physique d'un phénomène simulé lors de son édition interactive, de respecter les usages pour que cette approche puisse susciter l'adhésion de la communauté. Ce travail a abouti à la proposition d'une architecture modulaire basée sur l'outil de conception de jeu vidéo *Unity 3D*, offrant une grande flexibilité pour la conception de fonctionnalités de visualisation et d'interaction, associé à une infrastructure permettant de coupler cet outil avec n'importe quel code de simulation de fluide *opensource*. Cette approche a été appliquée à une méthode de calcul de type *Lattice-Boltzmann*, fortement parallélisable et peu sensible aux perturbations interactives durant une simulation en cours avec l'adaptation de *Palabos*. Enfin, l'usage de nouvelles méthodologies par calcul asynchrone a été exploré, étant susceptibles de compléter le besoin de performances déjà fournies par les approches fortement parallèles.

Sur cette base ont été menées trois expérimentations pour mesurer l'intérêt de méthodologie interactives et par jeux sérieux. La première expérimentation fut de comparer la performance entre la simulation conventionnelle et la simulation interactive. Une seconde expérimentation a été menée afin de mesurer la plus-value de l'immersion en termes d'expérience utilisateur et de performance. Enfin, la troisième expérimentation a cherché à mesurer l'impact du niveau de dégradation des résultats induit par les calculs asynchrones, sur la prise de décision, pour anticiper l'usage de ce type de méthodologies pour adresser les problématiques de performance relative au contexte interactif.

Title : Distributed and asynchronous approach for interactive simulations in fluid mechanics : *towards a serious game platform to support decision making*

Keywords : Interactive simulation, Serious game, Fluid simulation, Lattice Boltzmann method, Asynchronous computation, Virtual Reality

Abstract : It is largely contributed by the serious game *Fold'it*, that the interactive simulation and serious game approaches have been able to convince the researchers in the field of molecular biology. This serious game was designed to present and solve the problem of protein folding in a fun and collaborative way. Beyond the enormous educational potential, a group of *Fold'it* players has succeeded in finding the structure of an HIV protein, demonstrating the usefulness and the efficiency of such a methodology.

Inspired by this methodological advance in molecular biology in terms of interactive simulation and convinced of its potential for many applications in fluid mechanics, this thesis work was devoted to studying at which level and under which conditions an approach of interactive simulation or serious game can be applied to the field of fluid mechanics. The progress of numerical methods in fluid mechanics allied to the great computing power makes it possible to propose a potential transfer by the conception of an interactive fluid simulation platform including features for the design of models with initial conditions, visualization and interactive control of the fluid and its condition limits during a simulation in progress.

The objective of this thesis was therefore to design and evaluate an approach with interactive simulation and serious games devoted to fluid mechanics. The first step of this work was therefore to implement a dedicated platform, together with a continuous bibliographic study on the applications of interactive simulations and serious games in general and in fluid mechanics. Beyond the tech-

nological aspect and the underlying challenge, various ways have been explored to lead to a very constrained hardware and software architecture.

It is in fact a question of how to offer the performances required by interactivity, at the same time ensuring the physical relevance of a phenomenon simulated during its interactive edition. This work resulted in a proposal of a modular architecture based on the video game design tool *Unity 3D*, offering great flexibility for the design of visualization and interaction functionalities, associated with an infrastructure allowing to couple with any *open-source* fluid simulation code. This approach was applied on the *Lattice-Boltzmann* method, which is highly parallelizable and less sensitive to interactive disturbances during a simulation in progress with the library *Palabos*. Finally, the use of new methodologies by asynchronous computation has been explored, as they are likely to furthermore optimize the performance already provided by strongly parallel approaches.

Based on the works above, three experiments were carried out to measure the benefits of interactive methodology and serious games. The first experiment was to compare the performance between conventional simulation and interactive simulation. A second experiment was carried out to measure the added value of immersion in terms of user experience and performance. Finally, the third experiment sought to measure the impact of the degradation level of decision-making results induced by asynchronous calculations, to anticipate the use of this type of methodologies to avoid performance issues relating to the interactive context.

Remerciements

Je tiens à remercier en premier lieu mes co-directeurs de thèse Patrick Bourdot et Frédéric Magoulès pour m'avoir accueilli et aiguillé durant ces quatre années. Je souhaite remercier également Nicolas Férey pour son encadrement efficace, ses conseils précieux, sa patience et ses discussions passionnées et passionnantes. Enfin, je remercie Bénédicte, Sophie pour leur aide logistique et administrative.

Je remercie très chaleureusement les membres de mon jury et particulièrement les rapporteurs pour leur relecture attentive et leurs remarques constructives.

Je souhaite remercier aussi tous les membres de l'équipe VENISE pour ces quatre années passées avec eux dans une ambiance agréable, avec une pensée particulière qui s'impose pour les doctorants de l'équipe : Yiran, Nawel, Inoussa, Iuliia, Yujiro, Valentin.

Je remercie enfin mes parents qui m'ont permis d'arriver jusque là, ainsi que le *China Scholarship Council* qui a financé mes quatre années de thèse et qui m'a donné l'opportunité de cette expérience à la fois personnelle et académique en France.

Plan du manuscrit

Introduction	8
Contexte et motivations	8
Organisation de la thèse	11
1 Simulations interactives, immersives et jeux sérieux : vers quels usages pour la mécanique des fluides	13
1.1 Les simulations interactives	14
1.1.1 Les simulations interactives : définitions	14
1.1.2 Applications des simulations interactives	15
1.2 Les simulations interactives en mécanique des fluides	18
1.2.1 Les équations Navier-Stokes, une modélisation précise, mais coûteuse	18
1.2.2 Stable fluids, LBM, SPH : l'apport de l'infographie aux méthodologies de calcul	19
1.2.3 L'apport des GPU pour les simulations en mécanique des fluides	22
1.2.4 L'apport de l'apprentissage aux performances de calcul	22
1.2.5 Applications des simulations interactives en mécanique des fluides	23
1.3 La visualisation et l'interaction au centre des simulations interactives	27
1.3.1 Les techniques de visualisation 2D et 3D pour la mécanique des fluides	27
1.3.2 Les techniques d'interaction dédiées aux simulations interactives	31
1.4 Des simulations numériques aux jeux sérieux	34
1.4.1 Définition du jeu sérieux	35
1.4.2 Jeu sérieux à application militaire	36
1.4.3 Les jeux dédiés à la santé et aux applications médicales	37

1.4.4	Les jeux sérieux pour promouvoir l'activité physique et le sport	38
1.4.5	Jeux sérieux dédiés à l'exploitation du pétrole et du gaz	38
1.5	Les simulations interactives en Réalité Virtuelle	40
1.5.1	La multimodalité	42
1.5.2	Applications de Réalité Virtuelle pour l'éducation	42
1.5.3	Réalité Virtuelle dans le domaine pétrolier	44
1.6	Conclusion	45

2 Une architecture interactive, immersive et distribuée dédiée à la mécanique des fluides dans un contexte de jeu sérieux **47**

2.1	Couplage fort <i>versus</i> couplage faible	48
2.2	Modalités de couplage entre les modules de calcul, de visualisation et d'interaction	49
2.2.1	Limites du couplage faible par système de fichier - retour d'expérience entre <i>Fluent</i> et <i>Unity 3D</i>	49
2.2.2	Importance des systèmes de coordonnées et des normes dans les dif- férents composants	51
2.2.3	Couplage faible par architecture distribuée entre module de simulation et d'interaction/visualisation	52
2.2.4	<i>CFDriver</i> , une librairie pour faire communiquer composants de simulation et de visualisation/interaction	53
2.2.5	Gestion des données structurées dans <i>CFDriver</i>	55
2.3	<i>Unity 3D</i> comme base d'une plateforme de jeu sérieux dédiée à la simulation inter- active de fluide	56
2.3.1	Sélection et visualisation des résultats de simulation	56
2.3.2	Modifications interactives des paramètres physiques locaux et des condi- tions aux limites	59
2.3.3	Définition interactive des conditions initiales du phénomène simulé	61
2.3.4	Modalités d'interactions dans un environnement de bureau	61
2.3.5	Adaptation de la plateforme de simulation interactive de fluide à un contexte immersif	63
2.4	Modèles de calcul et simulation interactive de fluide	65
2.4.1	La méthode Boltzmann sur réseau : un bon compromis pour les simulations interactives	65
2.4.2	La discrétisation dans la méthode LBM	66

2.4.3	Conditions aux limites de <i>Bounce-back</i>	69
2.4.4	Exemples de modèle	69
2.4.5	Les modèles de calcul asynchrones	70
2.5	Conclusion	73
3	Etude comparative de l'usage de simulation interactive de fluide de type Boltzmann sur réseau pour la prise de décision	75
3.1	Introduction	76
3.2	Objectifs et attendus de l'expérimentation	76
3.2.1	Durée et taille de la simulation ciblée	77
3.2.2	Description des conditions initiales	78
3.3	Design expérimental	79
3.3.1	Description de la tâche	79
3.3.2	Participants	81
3.3.3	Données mesurées	81
3.3.4	Procédure	82
3.4	Hypothèses	84
3.5	Résultats de l'expérimentation	85
3.5.1	Résultats quantitatifs	85
3.5.2	Résultats subjectifs	85
3.5.3	Discussion	86
3.6	Conclusion	88
4	Apport de l'immersion pour l'aide à la décision dans un contexte de simulation de fluide interactif	90
4.1	Introduction	91
4.2	Adaptation de la plateforme de simulation interactive de fluide à un contexte immersif	91
4.2.1	Equipement immersif utilisé	91
4.2.2	Adaptation des techniques d'interaction 2D au contexte immersif	92
4.3	Objectifs et attendus de l'expérimentation	93
4.4	Design expérimental	93
4.4.1	Tâche	93
4.4.2	Participants	93
4.4.3	Données mesurées	94
4.4.4	Procédure	94

4.5	Hypothèses	95
4.6	Résultats	96
4.6.1	Résultats quantitatifs	96
4.6.2	Résultats subjectifs	98
4.6.3	Discussion	98
4.7	Conclusion	100
5	Etude de l'usage de modèles de calcul asynchrone dans un contexte de simulation interactive de fluide	101
5.1	Simulation interactive et modèle de calcul asynchrone : première application et retour d'expérience	102
5.1.1	Travaux préliminaires d'application d'un schéma de calcul asynchrone à une simulation de diffusion thermique	103
5.1.2	Illustration pratique de l'intérêt d'un modèle asynchrone sur un cas limite	104
5.1.3	Scénario réaliste de l'impact d'éditeurs interactives des conditions aux limites sur un parallélisme basé sur la séparation par domaines	106
5.1.4	Conclusion	106
5.2	Etude d'impact du caractère partiel des résultats de calcul asynchrone sur la prise de décision en simulation interactive de fluide	107
5.2.1	Introduction	107
5.2.2	Objectifs et attendus de l'expérimentation	107
5.2.3	Un modèle de calcul de type LBM 2D comme simulation intégrée dans <i>Unity 3D</i>	108
5.2.4	Design expérimental	109
5.2.5	Données mesurées	112
5.2.6	Hypothèses	113
5.3	Résultats	113
5.3.1	Résultats quantitatifs	113
5.3.2	Discussion	114
5.4	Conclusion	116
	Conclusion	118

Listes des illustrations

1.1	Interface du jeu <i>Fold'it</i>	14
1.2	Simulations interactives existantes	17
1.3	Simulation du château de sable avec la méthode SPH	21
1.4	Animations d'une seule goutte tombant dans une surface de fluide avec la méthode LBM	22
1.5	Conception interactive avec pression et vitesse simulées en temps réel	23
1.6	Simulation d'écoulement du sang avec la méthode SPH pour les gros vaisseaux sanguins et la méthode 1D pour les petits vaisseaux sanguins	25
1.7	Une artiste réalisant une session de croquis avec l'application Fluid Sketching	25
1.8	Comparaison entre les éclaboussures d'encre créées dans le monde réel et dans le système de simulation.	26
1.9	Energy2D	27
1.10	Comparaison des techniques de visualisation appliquées à un même fluide 2D	30
1.11	Comparaison des techniques de visualisation appliquées à un même fluide	30
1.12	Exemple de la technique extraction des caractéristiques	31
1.13	Pan-Rotate-Zoom	32
1.14	Technique de rotation avec 3 doigts	33
1.15	Comparaison entre jeux sérieux et jeux ludiques	35
1.16	Aperçu de l'environnement 3D du jeu sérieux CatCare	37
1.17	Jeux sérieux dans le secteur pétrolier	40
1.18	Domaines d'application de certains articles au cours de deux analyses thématiques	41
1.19	<i>ClinivaVR</i> : Salle de cours et emplacement des distracteurs	43

1.20 Aperçu des objectifs, avantages et problématiques des jeux sérieux dédiés à la simulation interactive de fluide : vue globale.	46
2.1 Couplage naif entre ANSYS-Fluent et Unity 3D par système de fichier	50
2.2 Opération de transformation pour passer d'un repère indirect à un repère directe .	51
2.3 Diagramme UML de déploiement illustrant l'architecture de la plateforme de simulation interactive	53
2.4 Rendu de l'interface entre l'eau et le vide par <i>Feature Extraction</i>	57
2.5 Résultat de rendu de type VOF et d'un plan de coupe	58
2.6 Exemple des résultats de vitesse	59
2.7 Modalité de modifications interactives du phénomène simulé	60
2.8 Procédure d'édition de paramètres physiques locaux et de modification de conditions aux limites coté simulation	60
2.9 Interface de la modification de la scène	61
2.10 Propagation des conditions initiales de <i>Unity 3D</i> jusqu'à <i>Palabos</i>	62
2.11 Tableau d'étiquettes des cellules	62
2.12 Exemple de condition aux limites sous format d'image	62
2.13 Conception du modèle de simulation interactive	63
2.14 Interface du casque de Réalité Virtuelle	64
2.15 Intégration de fonctionnalités de Réalité Virtuelle dans l'infrastructure logicielle et matérielle	64
2.16 Processus de propagation d'un cellule de réseau	68
2.17 Décomposition du processus de l'algorithme sous-jacent à la LBM	69
2.18 Illustration de la procédure de <i>Bounce-back</i>	70
2.19 Modèles de calcul synchrones et asynchrones	73
3.1 Visualisation sous forme de surface du résultat de la simulation interactive	79
3.2 Tâche d'expérimentation de la simulation interactive : trouver empiriquement pendant une simulation en cours la hauteur optimal du barrage pour éviter l'inondation	81
3.3 Répartition des participants : les 24 participants sont divisés en deux groupes. 12 d'entre eux commencent par une simulation interactive, puis terminent par une simulation non interactive. Pour les autres c'est l'inverse. Un design de type carré latin a été construit pour contrebalancer ces deux condition et l'ordre choisie des 3 scènes	84

3.4	Présentation des résultats quantitatifs : moyenne de TCT (a), nombre d'échecs (b) et temps d'observation (b). Les barres d'erreur indiquent l'erreur type	86
3.5	Résultats du questionnaire <i>NASA-TLX</i> : Charge globale (a), Mental Demand (b), Physical Demand(c), Temporal Demand (d), Performance (e), Effort (f) et Frustration (g). Les barres d'erreur indiquent l'erreur type	87
4.1	Dispositif de type casque immersif Oculus Quest 2	92
4.2	Répartition des participants	95
4.3	Résultats quantitatifs	97
4.4	Résultats du questionnaire <i>NASA-TLX</i> : Charge globale (a), Mental Demand (b), Physical Demand(c), Temporal Demand (d), Performance (e), Effort (f) et Frustration (g). Les barres d'erreur indiquent l'erreur type	99
5.1	Visualisation de la simulation de transfert thermique dans une scène <i>Unity 3D</i>	104
5.2	Temps de calcul d'une simulation bien distribuée avec nombre de processeurs différent	105
5.3	Temps de calcul sur chaque processeurs avec un processeur de calcul consacré à la communication vers l'outil de visualisation	105
5.4	Plan de charge des processeurs avec un processeur de calcul consacré à la communication vers l'outil de visualisation	106
5.5	Illustration du modèle de mouvement des objets dans le fluide 2D	110
5.6	Visualisation présentée à l'utilisateur des objets dans la fluide	110
5.7	Illustration du caractère asynchrone par domaine de notre simulation de fluide	111
5.8	Répartition des participants	113
5.9	Résultats quantitatifs : pourcentage d'objets collectés sous condition 0 (temps réel), condition 1 (20 ms de retard), condition 2 (40 ms de retard), condition 3 (60 ms de retard), condition 4 (80 ms de retard)	114
5.10	Résultats quantitatifs : différence entre les intensités différentes de la tâche. 100 objets (partie 1) vs 200 objets (partie 2)	115
5.11	Résultats quantitatifs : différence entre le même taux d'asynchronisation (Partie 1) et différents taux d'asynchronisation (Partie 3) dans les zones de simulation	115
5.12	Résultats quantitatifs : la distance entre le trajectoire de l'objet et le curseur posé par l'utilisateur	115

Introduction

Contexte et motivations

Les approches par simulation interactive dessinent peu à peu de nouveaux usages en démontrant leur utilité, en science comme dans l'industrie. Les simulations interactives peuvent être définies sommairement par le fait qu'un utilisateur puisse observer et agir de manière interactive, pendant une simulation en cours, sur ses objets ou ses paramètres.

La meilleure illustration académique soulignant la puissance potentielle, notamment en sciences, est l'application *Fold'it* [78]. Cette application propose à des utilisateurs de trouver le repliement et la structure 3D de protéines, optimale en termes d'énergie, en combinant approches par simulation moléculaire interactive et par jeu sérieux, avec une dimension collaborative. Cette démarche a abouti à une proposition correcte par un groupe de joueurs, de la structure d'une protéine du virus du VIH.

Inspirés par cette réussite dans le domaine de la biologie moléculaire, nous défendons le fait que les simulations interactives appliquées à la mécanique des fluides peuvent être un outil capable de répondre à de nombreuses problématiques scientifiques, industrielles et pédagogiques. A cela s'ajoute un potentiel important pour soutenir la prise de décision, la résolution de problèmes et l'optimisation, en particulier lorsque des humains sont nécessaires à toutes les étapes de ces processus.

Paradoxalement, les approches par simulations interactives dans le domaine de la mécanique des fluides sont peu répandues, car les verrous et les freins à son développement et à son utilisation, dus à la complexité des phénomènes physiques sous-jacents, sont nombreux.

En effet, mettre en place une méthodologie par simulations interactives en mécanique des fluides suppose de combiner plusieurs composants, celui de la simulation et du calcul, celui

de visualisation et celui de l'interaction. Chacun de ces composants nécessitent une expertise particulière, en physique, en sciences du numérique, en visualisation scientifique et en rendus graphiques, et enfin en interaction humain-machine.

Les composants de calcul et de visualisation doivent répondre par ailleurs à de nombreux prérequis imposés par un contexte interactif, à savoir être capable de générer plusieurs instants par seconde coté simulation, et de rendre graphiquement ces résultats coté visualisation à une fréquence au minimum proche de la persistance rétinienne autour de 10Hz, mais idéalement au delà de plusieurs dizaines de Hz.

Habituellement, ces outils sont mobilisés de manière séquentielle, nécessitant le rapatriement des résultats du calcul, terminés ou intermédiaires, pour les visualiser et les analyser à l'aide d'outils de visualisation généralement sans fonctionnalité d'interaction, à l'exception de la navigation 2D ou 3D et de la modification des propriétés de rendu ou des modalités de visualisation.

Le développement de systèmes de la simulation interactive des fluides doit intégrer dès sa conception, les fonctionnalités liées à la simulation, celles de la visualisation et de l'interaction, afin de rendre transparent et sans effort les transferts et les adaptations des données habituellement lourds entre le composant de simulation et celui de la visualisation ou de l'analyse.

Dans la littérature, les approches par simulation interactive en mécanique des fluides redéveloppent souvent un code de calcul pour les rendre compatible avec le temps interactif. D'un point de vue technique, cette démarche se justifie en permettant d'être à la pointe du progrès en termes d'exploitation des derniers processeurs ou cartes graphiques. Elle permet aussi une interface optimisée d'un point de vue logiciel entre le composant calcul, visualisations et interaction.

Cependant, cette démarche s'oppose aux usages. En effet, les utilisateurs professionnels ou scientifiques tiennent aux codes historiques de calcul, validés et éprouvés par la communauté scientifique ou ses utilisateurs pour les codes propriétaires. Par ailleurs, ces codes passent difficilement à l'échelle du fait d'un impossible déploiement sur les grands centres de calcul. Ces propositions techniques restent donc souvent des démonstrateurs, qui ne sont pas intégrés ni dans les travaux de recherche, ni dans les processus de production industrielle. Nous soulignons l'importance de respecter ces usages et notre première contribution fut donc d'intégrer des codes couplage faible entre les outils de calcul déjà existant et les modules de visualisation et d'interaction. Au delà du respect des usages, notre approche permet de substituer facilement un code de calcul par un autre, à l'image de ce qui se pratique au niveau de la biologie moléculaire. Le travail de conception et de développement pour suivre une telle démarche a nécessité de nombreux mois, incluant l'adaptation de codes existants de calcul comme *Palabos*, le choix et le développement des outils de visualisation et d'interaction nécessaires à nos expérimentations

interactives, puis l'implémentation de la couche logicielle et réseau permettant d'interfacer ces composants.

Si les perspectives d'application et d'usage des approches par simulation interactive dans le domaine de la mécanique semblent prometteuses, elles nécessitent cependant d'être accompagnées de travaux qui évaluent leur utilité et leurs avantages autant dans le secteur académique qu'industriel. Peu de travaux scientifiques se sont attachés à tenter de démontrer un intérêt et de caractériser ces approches. L'objectif de cette thèse est donc de proposer plusieurs expérimentations qui posent les premières briques d'un travail d'évaluation de l'utilité, de l'utilisabilité, de l'intérêt des simulations interactives en mécanique des fluides, dans plusieurs contextes d'interaction.

Nous avons choisi de situer ce travail dans le cadre d'application liée à la prise de décision, avec un scénario simplifié d'inondation. Il s'agit d'outiller le décideur avec une application de simulation interactive, pour savoir dans quelles mesures la prise de décision est plus efficace, et de savoir si cette approche améliore l'évaluation de la qualité de la décision.

Deux contextes de travail ont été plus particulièrement ciblés, le contexte classique, avec un écran, une souris et un clavier, et le contexte de Réalité Virtuelle, pour comparer une éventuelle plus-value apportée par ces différents environnements. Comparativement au contexte d'interaction classique, les systèmes de Réalité Virtuelle sont en effet souvent présentés comme permettant d'immerger un ou plusieurs utilisateurs au cœur de leur objet d'intérêt, et mettant en avant une meilleure perception tridimensionnelle de l'objet simulé. Ce contexte est par ailleurs fréquemment présenté comme particulièrement approprié dans le cadre de simulation de phénomène à large échelle, du fait de fonctionnalités de navigation avancées autour de l'objet numérique étudié. Enfin, ce contexte propose davantage de techniques d'interaction directe, permettant une manipulation interactive des contenus simulés supposée plus affordante et plus transparente.

Dans les deux cas, nous avons basé notre travail expérimental sur une approche par jeu sérieux, rassemblant les fonctionnalités élémentaires de conception, simulation, visualisation et interaction dans le cadre de notre étude autour d'un scénario d'inondation. Ce choix permet de rendre l'expérimentation accessible à tous les profils, et d'éviter l'écueil d'avoir à mobiliser uniquement des experts, chacun compétent sur des phénomènes ou d'outils spécifiques de visualisation, comme de simulation. Cette approche par jeu sérieux permet aussi au sujet de se concentrer sur les décisions à prendre face à l'observation du phénomène simulé, sans avoir à se préoccuper de la gestion technique de la simulation et de la visualisation, avec des fonctionnalités rendues les plus transparentes possible.

Par ailleurs, de manière complémentaire à l'étude dans deux contextes d'interaction des simulations interactives en mécanique des fluides, nous avons investigué sur les méthodologies de

calcul les plus appropriées aux contraintes lourdes imposées par les simulations interactives. Les approches par simulation interactive ont en effet fait évoluer les techniques de calcul, notamment pour les rendre compatibles avec les exigences de l'interactivité. Au delà du choix de l'approche de la méthodologie de calcul la plus appropriée pour nos travaux expérimentaux, une nouvelle classe d'approche, par calcul asynchrone, a retenu notre attention. Cette approche permet d'obtenir des résultats plus rapidement, au prix de résultats incomplets et partiels. Pour anticiper un futur usage de méthodologies asynchrones combinées à la simulation interactive, nous avons donc tenter de savoir dans quelle mesure le caractère partiel des résultats était susceptible de perturber l'activité dans un contexte de simulation interactive, notamment dans l'activité la prise de décision.

Organisation de la thèse

Cette thèse comprend principalement trois parties : l'introduction, la partie contenant le cœur du manuscrit divisée en cinq chapitres, puis les conclusions et les perspectives.

Le chapitre 1 présente le contexte de cette thèse et une sélection de travaux relatifs aux simulations interactives et aux jeux sérieux, plus précisément appliqués au domaine de la mécanique des fluides. Ce chapitre dresse par ailleurs un panorama des techniques de simulation, de visualisation et d'interaction, utilisées dans différents contextes allant des jeux sérieux à la Réalité Virtuelle.

Le chapitre 2 présente notre contribution technique pour développer une architecture logicielle pour coupler les outils de calcul de simulation de fluides avec les outils de visualisation et d'interaction. Ce chapitre explique notamment comment ce couplage a été réalisé avec succès sur la méthode de Lattice-Boltzmann à travers l'outil de simulation *Palabos*, et avec l'outil de développement de jeu vidéo *Unity 3D*, détourné pour l'occasion. Ce couplage a été nécessaire à notre démarche expérimentale, en suivant une démarche sous forme de jeu sérieux, pour évaluer l'intérêt d'approche par simulation interactive, ainsi que le couplage avec des techniques de calcul asynchrone.

Le chapitre 3 présente le scénario choisi et conçu pour nos expérimentations, basé sur la conception d'un barrage. Avec la simulation interactive, l'utilisateur peut observer l'évolution du courant et modifier la hauteur du barrage en temps interactif, sans redémarrer la simulation depuis ses conditions initiales. Cette expérimentation a pour objectif d'étudier les avantages et les inconvénients de l'utilisation du système de la simulation interactive.

Le chapitre 4 s'inscrit dans la même démarche que le précédent, mais en évaluant l'usage

de la Réalité Virtuelle immersive. Un casque de Réalité Virtuelle est utilisé pour étudier si ce contexte permet d'améliorer la performance de la tâche dans un contexte immersif par rapport à un contexte classique.

Enfin, le chapitre 5, se penche cette fois sur l'aspect calcul, et plus particulièrement sur l'utilisation de la technique du calcul asynchrone comme optimisation du système de la simulation de fluides interactive. Le calcul asynchrone permet en effet d'améliorer la performance de la simulation, et la question étudiée est de savoir dans quelle mesure la dégradation de la qualité des résultats induit par l'aspect asynchrone du calcul, impacte la décision dans un contexte de simulation interactive, dans une démarche expérimentale mimant l'incomplétude des données, à défaut d'avoir un code de calcul asynchrone suffisamment avancé dans le domaine de la mécanique des fluides.

Simulations interactives, immersives et jeux sérieux : vers quels usages pour la mécanique des fluides

1

Résumé

Ce chapitre introduit le contexte de ce travail de doctorat avec une présentation générale des approches de simulations interactives, de jeux sérieux et de Réalité Virtuelle, en introduisant quelques notions clé ainsi que des éléments de l'état de l'art permettant de caractériser ces approches. Il dresse un large panorama non exhaustif d'exemples sélectionnés afin d'illustrer les applications de ces approches dans l'industrie et dans la vie quotidienne.

1.1 Les simulations interactives

1.1.1 Les simulations interactives : définitions

Dans sa définition large, la simulation interactive est un terme qui désigne toutes les approches intégrant la modélisation, la simulation, l'analyse des données et la visualisation de phénomènes scientifiques numériques, de manière interactive. Dans une définition plus restrictive, adoptée dans ce manuscrit, ce terme désigne une méthodologie incluant toutes les techniques qui permettent aux utilisateurs de contrôler les objets numériques d'une simulation en cours, ainsi que ses paramètres, en incluant dans les cas les plus avancés, la production à la demande d'analyses des résultats pendant le calcul. Les simulations interactives ont prouvé leur utilité pour résoudre de nombreuses questions scientifiques. L'exemple le plus illustratif est l'application *Fold'it* [78] qui a aidé la communauté scientifique de la biologie moléculaire à résoudre la structure d'une protéine du VIH en utilisant une approche de simulation moléculaire interactive, doublée d'une approche par jeu sérieux collaboratif rendant la problématique du repliement des protéines accessible à tous. En mécanique des fluides, la complexité des calculs et de la physique sous-jacente a été pendant longtemps des facteurs freinant la diffusion et l'usage des simulations interactives alors que ce type de méthodologies constituent des candidats sérieux pour soutenir la prise de décision, la résolution de problèmes et l'optimisation, dans un grand nombre de contextes d'application. Par ailleurs, ce type d'approche permet d'aborder les problématiques dans une démarche davantage centrée utilisateur, l'humain étant observateur et acteur de son objet d'étude, y compris dans la phase de calcul.

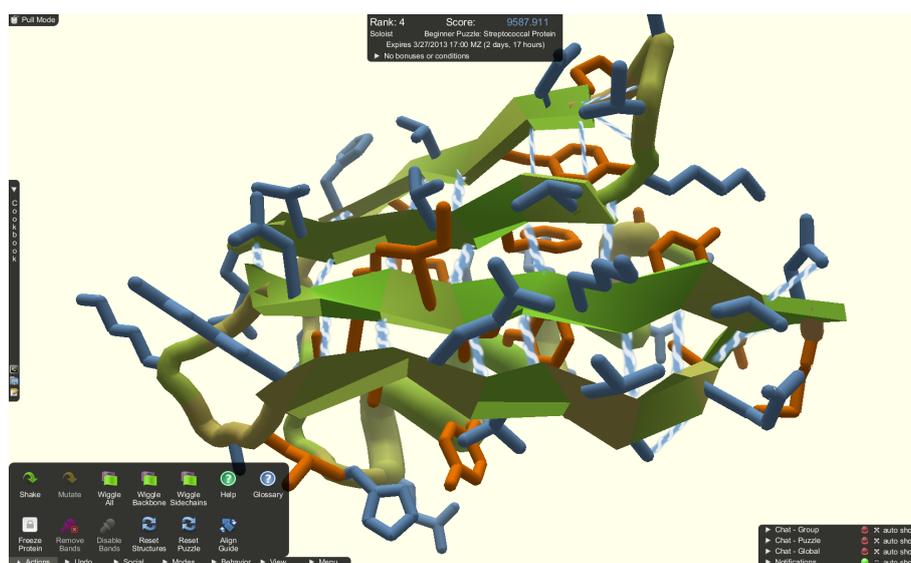


Figure 1.1: Interface du jeu *Fold'it* [78]

En mécanique des fluides comme dans tous les autres domaines, la taille et la complexité des objets numériques abordés en simulation ont augmenté avec l'explosion des capacités de calcul et de stockage, rendant le processus d'analyse et de visualisation de vastes quantités de données de plus en plus lourd et fastidieux. L'analyse et la visualisation des données sont traditionnellement effectuées après l'exécution d'une simulation. Cet usage nécessite cependant des transferts de données de plus en plus coûteux en temps et en ressources matérielles et humaines. Ce processus classique peine pourtant à être relayé par des approches plus rationnelles, notamment celles qualifiées de *in-situ*, favorisant la production d'analyses et de visualisations directement sur les lieux de stockage et de calcul. L'approche par simulation interactive converge par ailleurs avec le besoin de pilotage informatique très fin, pilotage défini comme la capacité de contrôler tous les aspects du pipeline de production de données en science du numérique [74]. Enfin l'approche par simulation est souvent présentée comme une méthodologie dont l'objectif est de réduire la complexité du processus séquentiel classique incluant la modélisation, la simulation puis l'analyse et la visualisation, en tentant d'englober et de mieux connecter toutes ces activités dans un contexte d'interaction unifié.

D'un point de vue applicatif, cette approche situant l'humain comme acteur et observateur du phénomène simulé, est aussi présentée comme susceptible de faciliter la prise de décision rapide dans les contextes de crise. Ainsi Leskens, Johannes G., et al. [74] ont développé un outil avec la simulation interactive pour simuler l'inondation, dont but est d'élaboration de stratégies pour atténuer ou s'adapter plus rapidement aux événements d'inondations dans un contexte de changement climatique.

1.1.2 Applications des simulations interactives

Simulation interactive dans le domaine médical

Le développement d'approches par simulation interactive dans le domaine de la recherche médicale et de la gestion des services de santé est très documenté [26]. Ces nombreux travaux ont essentiellement pour objectif d'assister les soignants dans l'aide à la résolution de questions complexes ou critiques dans leur travail quotidien ou dans leur formation. Le développement de ce type d'approches est conjoint au développement de l'usage de l'informatique dans le domaine de la gestion de la santé, qui a rendu possible la gestion, le stockage, l'analyse et la visualisation de volumes de données de plus en plus volumineux, hétérogènes et complexes, les rendant disponibles aux gestionnaires de manière interactive grâce à des interfaces de plus en plus performantes.

Les études menées sur les modélisations ou les simulations interactives dans le domaine de santé ont notamment ciblé la planification du personnel, l'allocation des ressources matérielles et humaines, la gestion des temps d'attente des patients, comme c'est le cas pour modéliser le flux de patients aux urgences [128]. D'autres outils ont été conçus pour évaluer l'impact de diverses configurations de distribution de médecins, sur les temps d'attente des patients dans les services d'urgence, pour optimiser la gestion de la surpopulation dans ces services [75]. D'autres travaux ont été menés pour modéliser et aider à la prise de décision en matière d'admission des patients[69] ou d'organisation des lits à l'hôpital[149].

Au delà de la gestion et de la formation relative au contrôle des flux de patients, aux ressources matérielles et humaines dans les services de santé, une application phare des simulations interactives est celle de la simulation du geste chirurgical. Les objectifs de ce type d'approches sont nombreux, essentiellement centrés sur la formation des chirurgiens et sur la préparation d'opérations. Ces simulations nécessitent une modélisation fine des comportements physiques des tissus et des outils chirurgicaux. Elles permettent l'entraînement des jeunes médecins à la pratique d'interventions chirurgicales sans risque pour les patients, ou de réduire les dommages post-opératoires grâce à une meilleure préparation des opérations en amont. Par exemple, une approche par simulation basée sur la méthode des éléments finis[141] permet de modéliser précisément l'insertion d'aiguilles et plus généralement la manipulation d'outils chirurgicaux, en intégrant les contraintes physiques tissulaires lors de la manipulation, en prenant en compte les mouvements physiologiques des organes, afin de minimiser le risque d'endommager les tissus environnants [37]. Dans cette approche, le calcul est basé sur un solveur optimisé sur GPU, permettant d'obtenir des fréquences de 60Hz, compatible avec les approches par simulations interactives, pour réaliser notamment le calcul de contact liés à l'insertion d'aiguille ou à la découpe de tissus mous hétérogènes en temps réel.

Simulation interactive pour l'éducation

Par ailleurs, l'introduction du numérique dans l'éducation a permis l'apparition de nouvelles approches pédagogiques, notamment pour l'enseignement de la physique. Partout dans le monde, il existe des preuves solides que les étudiants en sciences physiques, notamment dans les cours magistraux universitaires, n'intègrent que difficilement les concepts abstraits nécessaires à une bonne compréhension du monde physique. De plus, les étudiants quittent leurs cours avec des conceptions erronées du fait de nombreux phénomènes contre-intuitifs, qui sont difficilement compréhensibles uniquement en présentant les concepts fondamentaux du monde physique sous forme d'équations. Enfin, plus les cours, travaux pratiques, ou phénomènes étudiés sont

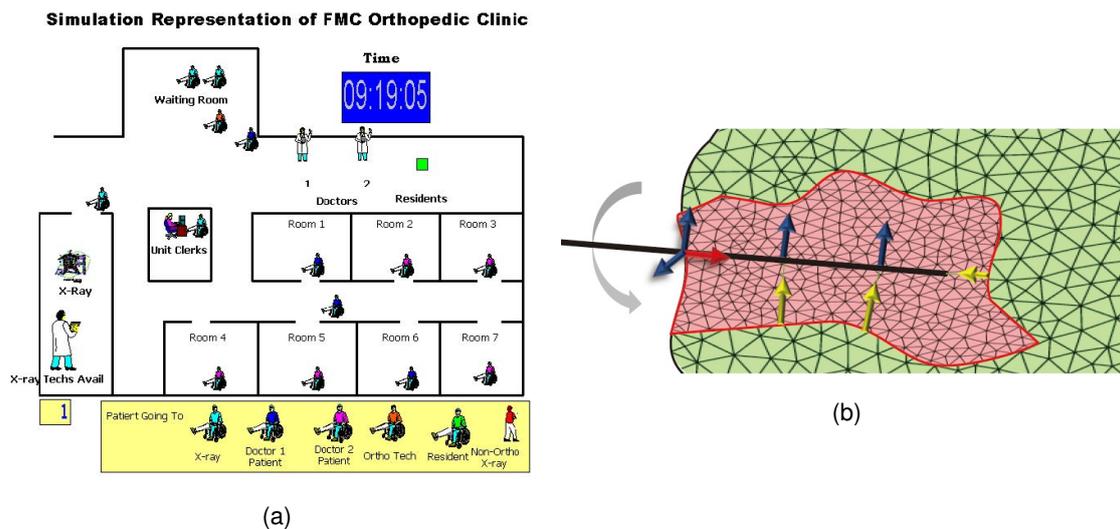


Figure 1.2: Simulations interactives existantes : Simulation de flux de patients (a) [128]. Simulation de l'insertion d'aiguille et d'outils chirurgicaux (b) [141].

éloignés de la vie quotidienne, plus la construction et l'intégration du fait scientifique restent dans le périmètre de la classe, plutôt que d'être mobilisés de manière ultérieure dans leur réflexion de la vie de tous les jours.[59][58][34]

Les plus-values de l'intégration du numérique et donc des simulations interactives dans les applications ou les jeux sérieux dédiés à la pédagogie peuvent être résumées ainsi : [19][20]:

- **Interactivité.** L'ordinateur permet à chaque étudiant de jouer un rôle actif dans le processus d'apprentissage, contrairement au rôle passif des formats de cours et de manuels. L'élève n'est plus un spectateur, mais participe cognitivement et physiquement au processus d'apprentissage. Il existe une multiplicité de situations à explorer et cela peut être fait à la discrétion de l'utilisateur.
- **Attention individuelle.** Les éducateurs savent que les élèves sont différents. Tous les élèves n'ont pas les mêmes antécédents et tous les élèves n'apprennent pas de la même manière. Cependant, bon nombre de nos approches conventionnelles de l'éducation utilisent une procédure rigide pour tous les élèves et ne permettent pas de tenir compte de ces différences. Un avantage du numérique est qu'il permet donc d'individualiser l'enseignement. Par exemple, comme tous les élèves n'apprennent pas au même rythme, les élèves ont besoin de temps différent pour parcourir et intégrer les contenus liés à l'apprentissage.

Par exemple *PhET*[119] est un projet pour concevoir des simulations dans un contexte d'enseignement et d'apprentissage de la physique. Il y a plus que 50 simulations, incluant des simulations adressant les problématiques de cinématique, d'énergétique, d'onde, de thermodynamique, d'électricité,

de rayonnement, de chimie, etc.. Le but du projet est d'utiliser les simulations pour augmenter l'engagement des élèves et améliorer l'apprentissage. Les simulations sont spécialement conçues pour aider les élèves à construire une solide compréhension conceptuelle de la physique grâce à l'exploration. Ces approches sont particulièrement attentive à faire le lien avec la vie quotidienne, à la fois pour impliquer les étudiants et pour soutenir leur apprentissage.

1.2 Les simulations interactives en mécanique des fluides

Afin de pouvoir transférer ces approches par simulation interactive et par jeu sérieux au domaine de la mécanique des fluides, des choix importants doivent être effectués afin d'utiliser les méthodes et les techniques de calcul les plus appropriés aux contraintes lourdes imposées par l'interactivité. De nombreux modèles physiques sont actuellement disponibles pour simuler des fluides dans leur environnement. Certaines de ces approches sont plus appropriées aux contraintes interactives pour accéder à une visualisation des données pendant une simulation en cours, premier prérequis à la simulation interactive. Les approches choisies doivent aussi être peu sensibles numériquement aux éditions interactives de leur paramètres (changement de conditions aux limites, changement de propriétés physiques locales du fluide), tout en restant par ailleurs physiquement pertinents.

On peut noter par ailleurs deux grandes approches discrètes pour modéliser un fluide, les approches lagrangiennes, dans lesquelles le fluide est représenté sous forme de particules situées et mobiles portant les propriétés physique ponctuelles du fluides, et les approches eulériennes, où l'espace est divisé en unité de volume statique de géométrie plus ou moins complexe, les propriété physiques étant porté par ses volumes, en leur centre, ou sur les surfaces, les arêtes ou les sommets qui les définissent. Les méthodes eulériennes classiques telles utilisant des grilles rectilinéaires régulières ou irrégulières, et les méthodes comme *Méthode des différences finies* ou *Méthode des volumes finis* pour la discrétisation des équations, ont longtemps été considérées comme les seules physiquement pertinente du fait de leur grande précision, au prix de temps de calcul très important par comparaison aux approches lagrangiennes.

1.2.1 Les équations Navier-Stokes, une modélisation précise, mais coûteuse

Pour la plupart des phénomènes physiques en mécanique des fluides, le modèle le plus reconnu, répandu et utilisé par la communauté scientifique est basé sur l'équation de Navier-Stokes qui a

la particularité d'être très précise, au prix d'une importante demande en temps de calcul, et donc voici l'expression pour un fluide incompressible.

$$\frac{D(\rho \vec{u})}{Dt} = -\vec{\nabla} p + \mu \nabla^2 \vec{u} + \rho g \quad (1.1)$$

associée à l'équation de la conservation de masse :

$$\frac{D\rho}{Dt} + \vec{\nabla} \cdot (\rho \vec{u}) = 0 \quad (1.2)$$

où u , ρ , p , g sont respectivement la vitesse, la densité, la pression et la gravité. μ est viscosité. $\frac{D}{Dt}$ décrit le taux de changement temporel d'une quantité au niveau d'un élément fluide qui est advecté avec l'écoulement.

D'un point de vue numérique, les équations de Navier-Stokes se concrétisent par l'usage de grilles rectilignes 2D ou 3D régulières ou irrégulières, chaque élément de la grille, portant les valeurs physiques évoluant à chaque pas de temps durant la simulation.

L'utilisation de méthodes précises nécessitent de mobiliser des centres de calcul dédiés et indépendants des ressources de visualisation généralement sous-dimensionnées, ce qui rend complexe la mise en place de simulation interactive. García, Manuel, et al. ont proposé un plateforme basé sur la méthode *Finite Volume Method (FVM)* et *Paraview*[137]. Cette plateforme distribué permet les clients à distance de visualiser la simulation sur un serveur en temps réel, d'effectuer le post-traitement interactive et de modifier des conditions aux limites et des paramètres.[54] L'interaction entre le l'utilisateur et le champ de fluide (contrôle au cours de la simulation) est manquant pour réaliser un système de simulation interactive.

Même si de nombreux résultats techniques ont été publiés pour soutenir l'approche de simulation interactive de fluide, cette méthodologie reste donc peu utilisée.

1.2.2 Stable fluids, LBM, SPH : l'apport de l'infographie aux méthodologies de calcul

Stable fluids : une méthodologie pionnière pour les simulations de fluides en temps interactif

J. Stam a été l'un des leaders dans développement de la simulation interactive en utilisant un modèle inconditionnellement stable[138] adapté aux rendus graphiques interactifs de fluides. Jusqu'à aujourd'hui, la plupart des applications basées sur une approche de simulation interactive utilisent des méthodes basées sur des particules telles que *Translating Eulerian Grids*[35] ou *Smoothed*

Particle Hydrodynamics (SPH) [104]. Ces méthodes visent des performances extrêmes, un rendu de haute qualité et une forte stabilité, mais au détriment de la pertinence physique, qui est souvent incompatible avec certains objectifs comme la prise de décision, l'enseignement de la physique ou la recherche.

Ces modèles, initialement conçus pour répondre à des problèmes de rendus graphiques temps réel, appliqués notamment aux jeux vidéo, ont donc longtemps été considérés comme peu pertinents d'un point de vue physique, mais ce n'était pas leur objectif.

Le modèle *stable fluid method* de J. Stam est en partie eulérienne puisqu'il suppose un partitionnement de l'espace, la partie gauche dans l'équation 1.1 est

$$\frac{D(\vec{u})}{Dt} = \frac{\partial(\vec{u})}{\partial t} + (u \cdot \vec{\nabla})u \quad (1.3)$$

Où $\frac{\partial(\vec{u})}{\partial t}$ est le taux du changement de vitesse à une grille statique qui ne se déplace pas avec le fluide. La dérivée advective $(u \cdot \vec{\nabla})u$ représente l'advection du champ de vitesse liée à la grille statique.

SPH pour Smooth Particle Hydrodynamics

Les modélisations dites eulériennes, s'opposent aux modélisations dites lagrangiennes, dans lesquelles c'est la particule, mobile, qui porte les valeurs physiques. C'est le cas par exemple, avec l'approche par hydrodynamique des particules lissées plus connue sous le nom de SPH pour *Smooth Particle Hydrodynamics*, où l'équation de la conservation de masse est garantie par la quantité constante de particules, et le fait que chaque particule a une masse constante, la masse totale restant donc constante. On peut donc ignorer l'équation 1.2. L'équation 1.1 décrit le mouvement des particules. La partie gauche décrit le changement de moment, et l'accélération des particules est décrit comme suit :

$$a_i = \frac{f_i}{\rho_i} \quad (1.4)$$

où f_i est la somme de toutes les forces sur la partie droite. Afin de simuler le mouvement des particules, la pression, la viscosité, la tension de surface et les forces externes sont exprimées par sommation discrète.

Les méthodes lagrangiennes sont souvent utilisées pour les approches par simulations interactives, notamment la méthode SPH[103]. Cette méthode a été introduit pour la première fois dans la communauté graphique par Muller et al. pour simuler l'écoulement des fluides[104]. La méthode SPH a également été utilisée pour simuler des solides élastoplastiques[105] et des

matériaux granulaires comme le sable[2]. Le couplage fluide-solide a également été réalisé par la méthode SPH, pour simuler l'interaction du fluide avec solide élastoplastique[136], avec des matériaux granulaires[87] et des matériaux poreux[88]. Enfin Les approches particulaires, sont plus propices au calcul des interactions du fluide avec des objets mobiles ou les bornes de la simulation, manipulés par l'utilisateur pendant une simulation interactive. Elles sont en effet moins sensibles aux éditions dynamiques de l'utilisateur des propriétés physiques ou des conditions aux limites.



Figure 1.3: Simulation du château de sable avec la méthode SPH [2].

Venues aussi de l'infographie, les méthodes par SPH ont peu acquis leur lettre de noblesses en simulation physique, mais plusieurs recherches ont été fait pour simuler les phénomènes physiques avec cette méthode. Par exemple Yan, Xiao, et al. [160] ont développé un cadre unifié qui étend le cadre SPH fluide multiphase pour traiter également les corps déformables et les matériaux granulaires, comme la simulation des granules de café ou d'un bonbon mou se dissolvant dans l'eau, et la simulation de l'eau versée sur le sable imprégnant le sable et le mouillant qui représente l'écoulement dans un milieu poreux.

LBM pour méthode de Boltzmann sur réseau

La méthode de Boltzmann sur réseau traite les fluides comme le résultat du comportement collectif des particules effectuant des processus de propagation et de collisions modélisés sur un maillage discret en réseau. Des recherches ont déjà effectué pour comparer la méthode LBM et SPH. Les résultats montrent que LBM est plus précise[46]. Beaucoup de recherche ont été effectué pour optimiser le calcul de visualisation interactive de tel résultat afin de visualiser le champ scalaire et vectoriel de manière interactive, afin de permettre que la visualisation soit effectuée pendant le calcul[163]. Cette visualisation en temps interactif des résultats d'une simulation en cours, a été appliqué dans les domaines différents tel que la simulation de fluide sanguin cérébral[51] ou l'animation de fluides à surface [147]. Cette approche interactive est le premier par de simulation interactive, car elle combine la simulation et la visualisation, mais elle n'intègre ni la modélisation, ni l'édition pendant une simulation en cours.

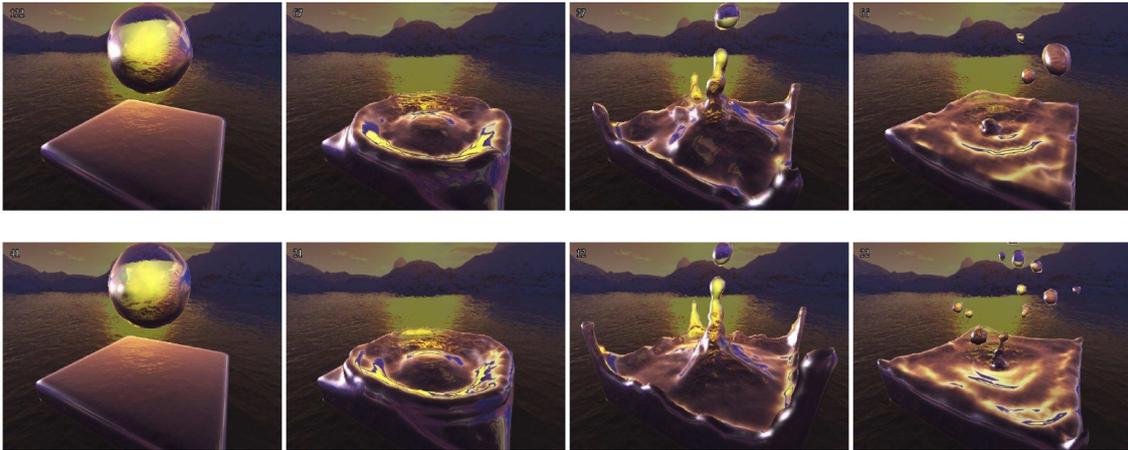


Figure 1.4: Animations d'une seule goutte tombant dans une surface de fluide avec la méthode LBM [147].

1.2.3 L'apport des GPU pour les simulations en mécanique des fluides

Le calcul parallèle est un apport important pour la simulation de fluide en matière de performance, et a permis l'émergence de méthodes comme la méthode de Boltzmann sur réseau (LBM) désormais utilisées dans une large variété d'applications. En effet, en raison de sa formulation et de sa dynamique, le LBM présente certains avantages par rapport aux autres méthodes de simulation de fluides, en particulier le fait d'être plus facilement parallélisable [145].

D'autres approches comme celle de Jo Stam, ont été aussi l'objet de travaux notamment de Harris[62] qui a introduit une technique pour les simulations de dynamique des fluides rapides, montrant que la méthode des « stable fluids » pouvait être implémentée sur le GPU avec le langage Cg spécialisé, obtenant une accélération de 6 fois par rapport à une simulation CPU équivalente. Brandvik et Pullan ont signalé une accélération de 40 fois en utilisant un GPU par rapport à l'implémentation CPU d'un solveur Euler 2D, et une accélération de 16 fois pour le solveur 3D[23][24].

1.2.4 L'apport de l'apprentissage aux performances de calcul

L'apprentissage automatique a aussi permis d'améliorer les performances de calcul, approche introduite par Umetani et Bickel[71] qui ont développé une méthode qui permet d'anticiper efficacement par apprentissage la réponse non linéaire des fluides en utilisant une régression de processus gaussien. Cette approche a été appliquée à la conception et l'optimisation d'une carrosserie de voiture.

FluidsNet est un autre système de simulation de fluide lagrangienne basé sur l'apprentissage

en profondeur. Ce système peut générer automatiquement un modèle d'animation fluide basé sur des données de simulation de fluide.[162]

1.2.5 Applications des simulations interactives en mécanique des fluides

Dans le domaine de la mécanique des fluides numérique (CFD en anglais pour Computational Fluid Dynamics), il existe de nombreuses applications des approches par la simulation interactives de fluides, dont voici une proposition de sélection.

Applications à caractère industriel

Les simulations interactives basées sur la mécanique des fluides ont été appliquées à plusieurs domaines, notamment dans l'industrie automobile comme celle déjà citée juste au-dessus développée par Umetani, N. and Bickel, B.[148]. Cette application propose un système qui permet aux utilisateurs de concevoir et d'optimiser de manière interactive une forme 3D avec la pression sur la surface et le champ de vitesse prédits en temps réel. Ce système intègre l'apprentissage automatique pour accélérer la simulation. Cette méthode est aussi utilisée par plusieurs autres applications, par exemple Ma, Pingchuan, et al. ont développé un système pour simuler l'interaction entre les objets solide et du fluide. La utilisation de l'apprentissage automatique permet de créer une simulation interactive où l'utilisateur peut contrôler la bordure de la zone de simulation.[93]

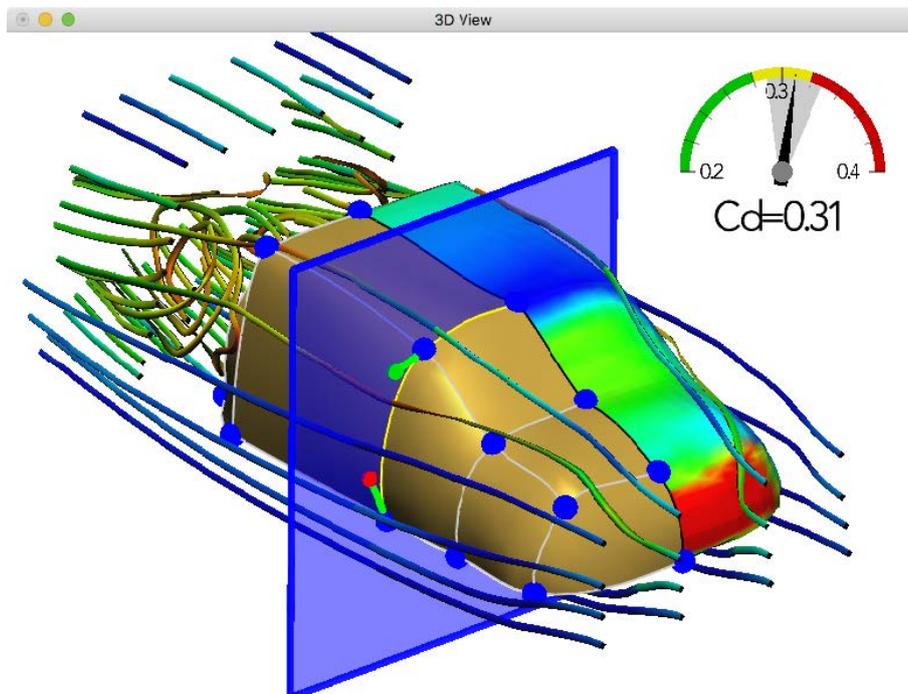


Figure 1.5: Conception interactive avec pression et vitesse simulées en temps réel [148].

Par ailleurs, en maintenance industrielle, en Asenov, Martin, et al.[8] ont développé un système de détection de fuite de gaz basé sur la simulation de fluide. En contrôlant un drone équipé d'un capteur de gaz, le système aide les utilisateurs de trouver la position de fuite.

L'usage de plus en plus fréquent de méthode de calcul rapide, comme la LBM par exemple a permis le développement des systèmes de simulation interactive de plus en plus avancés et intégrés. C'est le cas d'un outil de CAO avec simulation interactive de fluide permettant aux utilisateurs de visualiser les résultats en temps réel et d'ajouter des objets dans le modèle CAO. Le calcul est basé sur un serveur de simulation qui permet aussi aux utilisateurs de visualiser et contrôler l'application depuis une tablette ou un équipement Réalité Virtuelle.[64]

Applications à caractère hydrologique

De même l'usage de plus en plus fréquent et la précision en croissance des méthodes de calcul rapide comme la méthode *Smoothed Particle Hydrodynamics*(SPH) a permis d'intégrer des phénomènes de plus en plus complexes dans les applications. C'est ainsi que Kipfer, P., & Westermann, R. [79] ont développé un système de simulation interactive pour la simulation de rivière. Avec la performance du calcul du GPU, on peut réaliser la simulation du courant, de la trajectoire et des sorites du lit de la rivière en temps interactif.

Applications à caractère médical

Même si les applications de la méthode SPH sont souvent associées aux jeux ou aux effets de rendu haute performance, car elle est réputée d'avoir une précision qui ne suffit pas pour simuler correctement certains événements physiques, mais cette méthode est aussi utilisée dans différents applications où la précision n'est pas très important.

De plus en plus d'applications utilisent en effet avec succès cette approche pour simuler l'écoulement du sang par exemple, ayant des propriétés dynamiques complexes qui changent avec le temps, notamment le processus de coagulation [106].

Contraire au gros vaisseau sanguin qui est souvent simulé avec la méthode SPH, un petit vaisseau sanguin peut être considéré comme une simulation de fluide en 1D. On peut donc remplacer la méthode SPH par une méthode traditionnelle. Lee, J., Kim, M., et Lee, D. Y. ont développé le système pour simuler les gros vaisseaux sanguins avec la méthode particule et les petits vaisseaux sanguins avec une simulation 1D.[84]

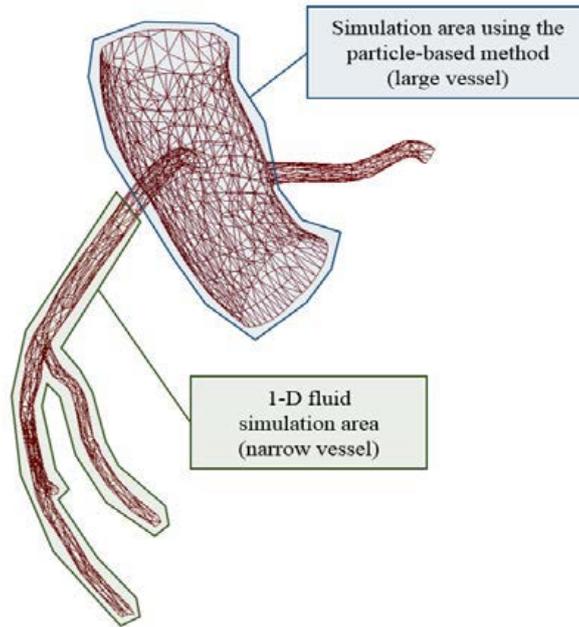


Figure 1.6: Simulation d'écoulement du sang avec la méthode SPH pour les gros vaisseaux sanguins et la méthode 1D pour les petits vaisseaux sanguins [84].

Applications à caractère artistique

Dans le domaine artistique, il existe aussi de nombreuses applications qui bénéficient et détournent les approches par simulation de fluide interactive. C'est le cas de *Fluid Sketching*[50] qui est un système permettant de créer des œuvres d'art avec des fluides en 3D dans un environnement virtuels immersifs. Il permet aux artistes de dessiner des croquis fluides en 3D et de les manipuler *via* des périphériques d'entrée à six degrés de liberté avec les fluides animés. L'utilisateur peut aussi contrôler les propriétés des fluides et sa diffusion.



Figure 1.7: Une artiste réalisant une session de croquis avec l'application Fluid Sketching [50].

Lei, Su-lan Eugene, et al. ont créé un outil qui permet aux artistes de créer des œuvres effet d'éclaboussures d'encre. Basé sur la méthode SPH, cet outil peut simuler le mouvement d'encre réaliste, comme dans la Fig.1.8. [85]

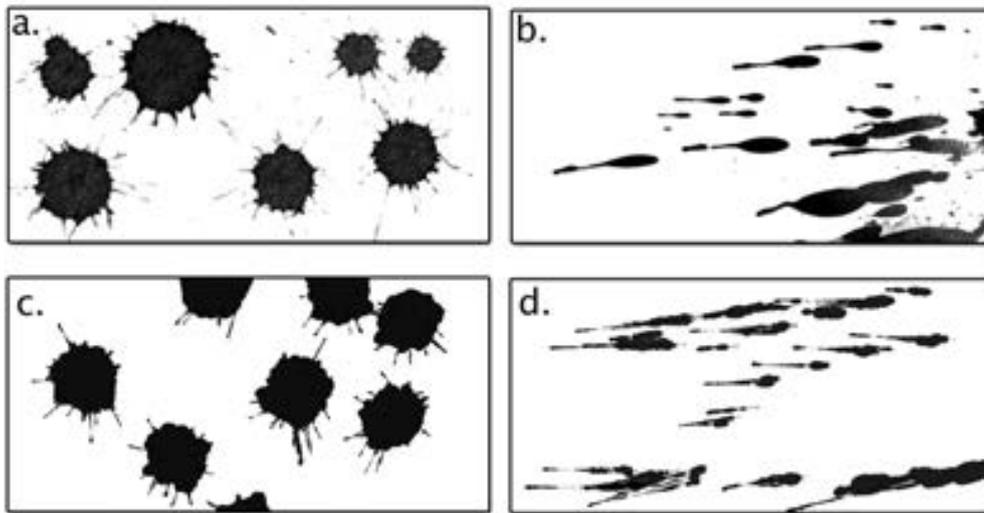


Figure 1.8: Comparaison entre les éclaboussures d'encre créées dans le monde réel (a, b) et dans le système de simulation (c, d) [85].

Applications à caractère éducatif

Pour l'éducation, de nombreuses applications en 2D ont aussi été développées pour apprendre la physique dans le domaine de mécanique des fluides ou du transfert thermique. *Energy2D* est un outil de simulation open source. Cet outil permet les enseignants et les élèves d'apprendre la physique de transfert thermique ou matériel. L'utilisation d'un algorithme rapide permet *Energy2D* d'afficher des résultats approximatifs sans perdre trop de précision, qui est suffisant pour apprendre les lois physique [159]. Une autre application utilisée pour l'enseignement est un outil de la simulation de l'effet d'ébullition violente en dilution d'acide sulfurique concentré. Cette application montre aux élèves les expérimentations chimiques dangereuses.

SWISH est un outil qui permet de simuler la sensation physique pour la manipulation d'un fluide dans un espace virtuel. Ce système utilise un casque de Réalité Virtuelle pour la visualisation de fluide et un dispositif haptique pour générer la sensation des forces de fluide.[129]

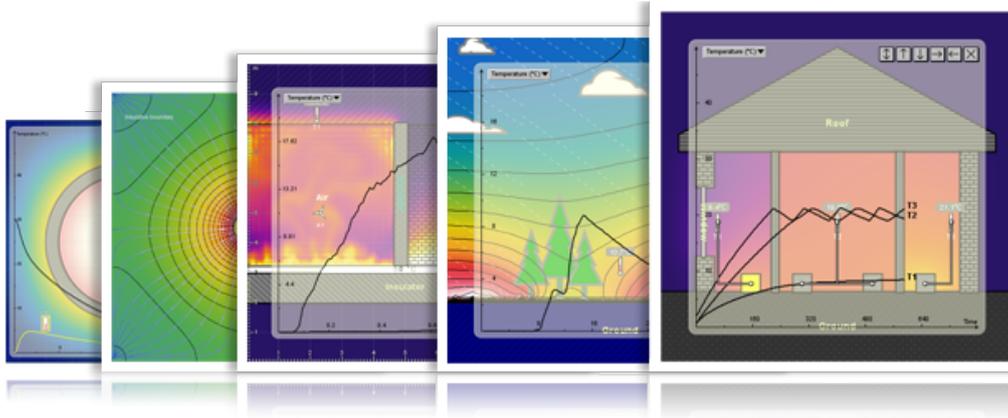


Figure 1.9: Energy2D : <https://energy.concord.org/energy2d/index.html>

1.3 La visualisation et l'interaction au centre des simulations interactives

1.3.1 Les techniques de visualisation 2D et 3D pour la mécanique des fluides

La fonctionnalité de visualisation est au cœur de la méthodologie par simulation interactive, avec contrainte importante d'effectuer à la fois des rendus qui permettent l'interaction et qui évidemment permettent de rendre les résultats à étape de calcul de la simulation. Pour la mise en place d'une application de simulation interactive de fluide, il est donc très important de choisir les outils et les techniques de visualisation appropriées.

En matière de visualisation des données de fluide en 2D ou en 3D, il existe un grand nombre de techniques différentes. Le choix de la technique ou de la modalité dépend des circonstances, notamment le type de résultats provenant des simulations, la nature du phénomène étudié, les expériences, les objectifs de l'analyse...

La visualisation de fluides est un sous-domaine de la visualisation scientifique spécialement dédiée au rendu de fluides. Elle est utilisée dans une large gamme d'applications, des prévisions météorologiques à l'étude de l'aérodynamisme des véhicules. Les objectifs académiques et industriels sont très variés, ce qui explique la richesse des représentations graphiques disponibles dans le domaine de la visualisation de fluides. Il existe de nombreuses techniques de visualisation de fluides visant à présenter tout ou partie des données de fluides, de manière compréhensible, décrites par exemple dans la classification de Post, Frits H., et al. [120].

Les données caractérisant un fluide, issues de simulations ou d'expérimentations sont car-

actérisées par des champs scalaires ou vectoriels, c'est à dire un ensemble des propriétés physiques situées dans l'espace et le temps, définissant localement le fluide à un instant donné : pression, température, vecteur vitesse, rotationnel, facteur q ... Ce positionnement local peut être statique dans le temps, dans le cas des approches eulériennes dans lesquelles c'est un volume ou un point qui porte l'information physique. Cette grille que compose ces volumes est appelée topologie ou géométrie et est définie au départ pour tout le calcul de la simulation. Ceci permet de faire varier la granularité du calcul dans certaines zones du phénomène simulé, en utilisant des grilles rectilignes irrégulières, dans lesquelles les volumes sont des parallélépipèdes, ou des approches par éléments finis, dans lesquels les volumes sont plus complexes. Ce positionnement peut être aussi dynamique, notamment dans les approches lagrangiennes dans lesquelles ce sont des particules mobiles qui portent l'information physique.

Ces données sont ensuite utilisées puis transformées en représentations graphiques 2D ou 3D pour pouvoir être interprétées par un humain. Les techniques de visualisation de fluides sont souvent classées en 3 classes, la visualisation directe, la visualisation géométrique, et la visualisation par texture.

Visualisation de fluides par visualisation directe

Le premier type de technique de visualisation est la visualisation de fluides dite "directe" qui vise à présenter les données de manière explicite avec le moins de transformation possible des données brutes lors de la construction de la représentation graphique des données. En visualisation directe en 2D ou 3D, ces transformations se limitent à afficher de manière située des éléments numériques textuels, d'utiliser des échelles de couleurs pour représenter des valeurs numériques scalaires, ou des éléments 3D orientés (appelés également glyph) pour représenter les vecteurs comme la vitesse. Cette technique s'applique à la fois aux approches eulériennes et lagrangiennes. Un exemple de visualisation directe des fluides utilise cette technique pour visualiser les particules de micro fluide[127]. Cette classe d'approche est incontournable en simulation interactive car elle permet de rendre de manière directe les propriétés des fluides dans la simulation, et permet aussi quand la méthodologie de calcul le permet la modification des propriétés du fluide du fait de la relation directe entre les éléments visuels représentés et les propriétés scalaires et vectorielles de la simulation.

Visualisation géométrique

La visualisation des fluides géométrique est également une des grandes classes de visualisation de fluides. Cette technique construit des objets géométriques abstraits à partir des données scalaires ou vectorielles dans les données. Les éléments géométriques d'un fluide peuvent être des contours, des iso surfaces, des ruissellements, des rationalisations, des stries, des chronologies et des trajectoires. On peut utiliser par exemple utiliser des lignes courbes calculées à partir du vecteur vitesse permettant de caractériser le mouvement de fluide. Ces techniques sont complémentaires offrent généralement des visualisations plus faciles à interpréter que les techniques de visualisation de fluides par visualisation directe, en accédant en un seul coup d'œil aux dynamiques et aux structures globales d'un fluide.

Visualisation à base de textures

La visualisation basée sur la texture est un autre type de technique de la visualisation des fluides. Cette technique comme son nom l'indique présente les données aux utilisateurs en les rendant par des images ou des textures généralement avec une forte résolution spatiale. La technique de l'advection croisée et la diffusion d'erreur sont deux exemples de techniques basées sur la texture pour visualiser l'incertitude dans un écoulement dépendant du temps.[21].

Extraction des caractéristiques

Outre ces techniques standards de visualisation de fluides, l'extraction des caractéristiques est une méthode de la visualisation de fluides avec un but très spécifique : il s'agit d'extraire des structures de vortex du fluide, structure qui n'est pas directement accessible dans le résultat brut d'une simulation. En effet, en général tous les points singuliers, tels que centres de vortex ou points de cisaillements [57] ont une très grande importance pour la recherche tant théorique que pratique. Par exemple Jiang [72] a proposé un algorithme simple et efficace pour la détection du centre du vortex inspirés de résultats issus de la topologie combinatoire.

Toutes ces techniques de visualisation ne s'appliquent évidemment pas à la simulation interactive. De manière générale, les techniques retenues dans un scénario interactif doivent permettre aux utilisateurs d'analyser et de comprendre rapidement le phénomène simulé, et doivent faciliter la manipulation des propriétés physiques locales du fluide ainsi que de l'environnement topologique (modification des murs, ajout d'obstacles ou tout autres conditions aux limites, positionnement et débit de la source ou des puits, etc.) qui caractérise le phénomène observé.

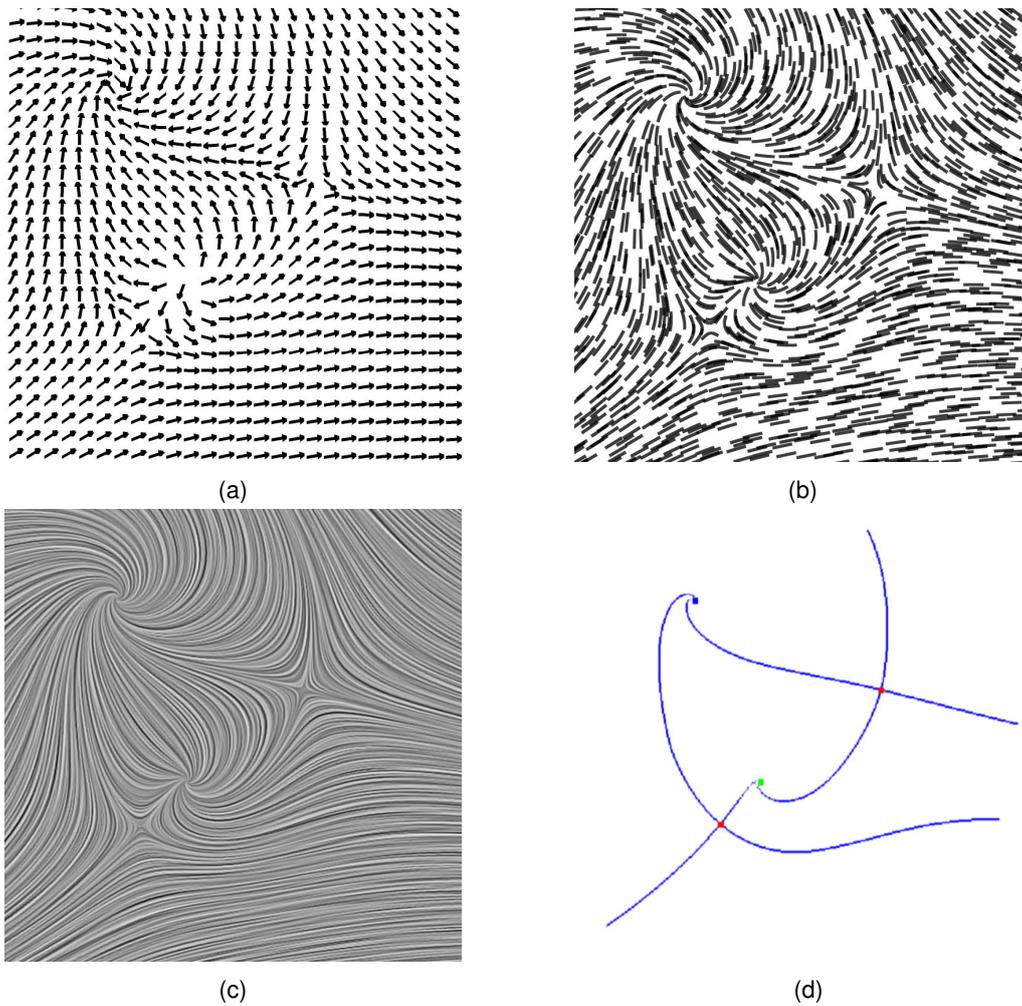


Figure 1.10: Comparaison des techniques de visualisation appliquées à un même fluide 2D : (a) arrow plot, (b) streamlets, (c) LIC, (d) topology-based [155].

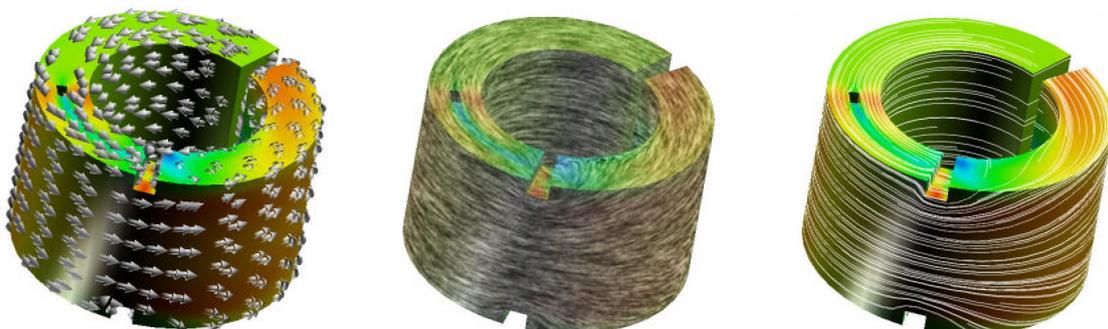


Figure 1.11: Comparaison des techniques de visualisation appliquées à un même fluide : Visualisation directe par flèche (gauche)), basée sur la texture par l'utilisation de Line Integral Convolution (milieu), et visualisation basée sur des objets géométriques, ici streamline(droite) [83].

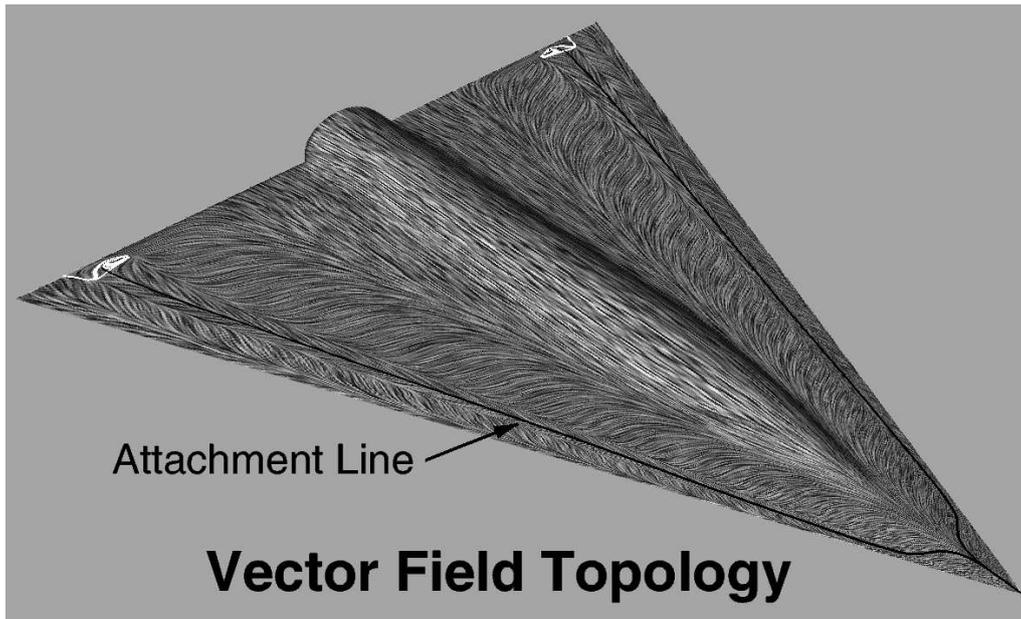


Figure 1.12: Exemple de la technique extraction des caractéristique : ligne de séparation (blanc) et attachement (noir) [57].

1.3.2 Les techniques d'interaction dédiées aux simulations interactives

De même que pour la visualisation, un des piliers des approches par simulation interactive, est par définition l'interaction. Il y a plusieurs niveau fonctionnalités interactives essentielles dans une application de simulation interactive. La navigation rendant possible de régler le point de vue de l'objet étudié, d'explorer plus en détail certaine de ses caractéristiques, de s'immerger au cœur du phénomène dans le cas de phénomène tridimensionnel. Ensuite, comme la simulation interactive suppose la manipulation et l'édition de son contenu en temps interactif, il s'agit évidemment d'être en mesure de pouvoir facilement sélectionner un ou plusieurs des composants de la simulation, pour pouvoir modifier leurs propriétés physiques. La sélection, en particulier dans les environnements 3D complexes et hétérogènes est donc prérequis à toute fonctionnalités de manipulation et d'édition pendant une simulation en cours. Chacune de ces fonctionnalités interactives de base peut faire appel au nombreuses techniques disponibles, qu'il faut savoir choisir et adapter pour les besoins des approches par simulations interactives.

La navigation ou comment explorer les données de simulation

Dans un environnement numérique, l'utilisateur a besoin de se déplacer dans l'environnement afin d'obtenir différents points de vue sur l'objet observé. Mackinlay et al.[94] a classifié quatre catégories de techniques de modification de point de vue dans l'espace 3D, dans les environ-

nements réalistes:

- Le mouvement classique : mouvement exploratoire, par exemple marcher dans une conception architecturale simulée ;
- Le mouvement ciblé : mouvement avec un but spécifique, comme se déplacer pour visualiser les détails d'un modèle d'ingénierie ;
- Déplacement instantané : mouvement à une position et orientation précisée, par exemple vers une position de visualisation spécifique par rapport à un modèle CAO : l'utilisateur doit fournir la position et orientation exacte de sa destination.
- Mouvement contraint par une trajectoire : mouvement le long d'une trajectoire de position et d'orientation, comme un mouvement de caméra cinématographique.

Dans les scènes représentant des objets plus abstraits dans le cadre de la visualisation scientifique, le mouvement exploratoire habituellement implémenté par la marche dans les scènes réalistes permettant d'explorer un bâtiment, est remplacé par la technique "Rotate-Pan-Zoom" utilisés dans presque tous les environnements de modélisation 3D comme les logiciel *Autodesk's 3ds Max*, *Catia*, ou *Unity 3D*. Cette technique n'est cependant pas sans poser problème de perte d'orientation et de points de repère, les scènes en visualisation scientifiques n'étant pas nécessairement sémantiquement orientée.

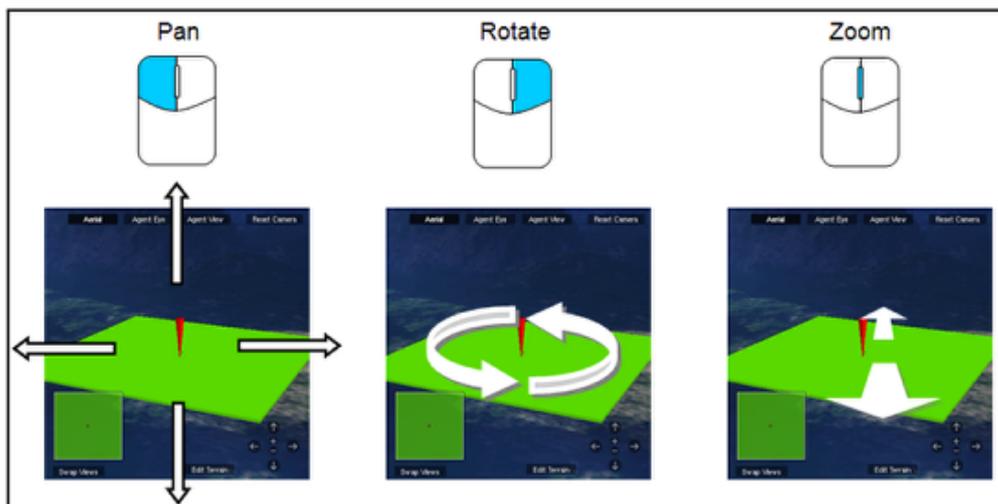


Figure 1.13: Pan-Rotate-Zoom : http://web.mit.edu/mitstep/webdav/How%20to%20Navigate%20in%20Spaceland/How_to_Navigate_in_Spaceland.html.

- Rotate : La caméra orbite autour d'un point central dans n'importe quelle direction en gardant focalisé au point central.

- Pan : La caméra se déplace sur le plan perpendiculaire à la ligne visée.
- Zoom : La caméra se déplace le long de la ligne visée.

Les techniques de rotation 3D sont généralement dédiées à une interaction avec une souris, et sont basées sur les techniques comme *Virtual Sphere*[33] et *ArcBall*[135]. Les deux techniques sont basées sur un concept de boule virtuelle avec l'objet au centre. Ils utilisent la projection de l'emplacement de la souris sur une sphère pour calculer l'axe et l'angle de rotation. Les interfaces plus modernes étant tactiles voire multitouch, certains auteurs, comme Reisman et al.[124] ont développé une méthode sur écran pour une interaction 3D multitouch, permettant aux utilisateurs de manipuler directement dans un contexte interactif 2D, de manipuler directement des objets 3D.

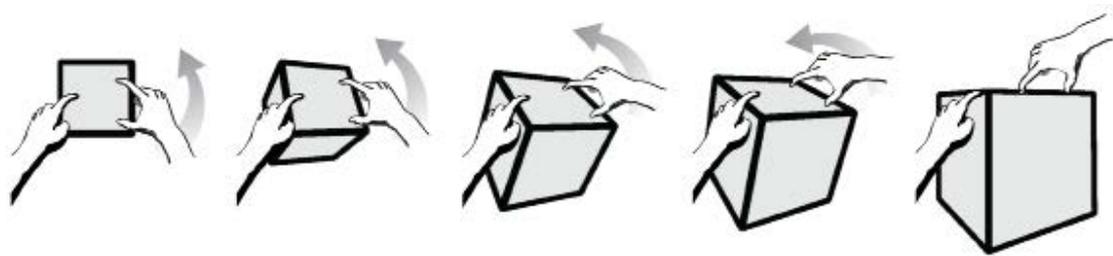


Figure 1.14: Technique de rotation avec 3 doigts : [124].

La sélection et la manipulation, deux tâches centrales dans les simulations interactives

Une autre tâche typique effectuée dans un environnement virtuel 3D est la sélection d'objets et sa manipulation directe. Les techniques de manipulation 3D comprennent trois tâches fondamentales : la translation d'objet, la rotation d'objet et la mise à l'échelle de l'objet.[60]

La technique la plus utilisée pour sélectionner des objets dans un contexte de bureau consiste à placer le curseur de la souris sur la projection 2D de l'objet 3D rendu sur l'écran, et à cliquer sur un bouton de la souris. La technique est basée sur *raycasting*[102]. Cette technique utilise le rayon du point de l'œil au pixel actuellement sélectionné par le pointeur de la souris pour trouver le premier point d'intersection partant de ce pixel vers profondeur de la scène avec un objet 3D du modèle. Il est donc parfois nécessaire de changer de point de vue pour sélectionner des objets l'un derrière l'autre, éventuellement occulté et peu visible par projection. La précision de cette approche de sélection peut aussi laisser à désirer en cas de scènes avec beaucoup d'objets sélectionnables. Elmqvist et Fekete[48] ont donc enrichi ce concept avec le pointage sémantique à un contexte 3D pour améliorer les performances de sélection d'objets.

Dans le cas de la simulation de fluides interactives, il s'agira essentiellement de sélectionner des volumes dans l'espace dans les approches eulériennes, ou des particules dans les approches

lagrangiennes. Par ailleurs, il doit être possible de modifier les conditions aux limites pour y ajouter des obstacles ou des structures dans lesquelles évolue le fluide.

Pour la modélisation de la scène de manière interactive, notamment la définition des structures autour du fluide, il est important de pouvoir facilement manipuler des objets 3D. La manipulation d'objet est donc une autre tâche essentielle de la simulation interactive, assez classique dans un environnement virtuel 3D, en particulier lors de la construction de la scène. Lors de la conception d'une scène avec plusieurs objets, les utilisateurs doivent réaligner et ajuster à plusieurs reprises différentes parties. Il est important de fournir une technique de manipulation d'objets efficace. Une technique est d'utiliser un manipulateur. Les manipulateurs de positionnement sont introduits par Nielson et Olsen[111] sur les techniques de manipulation directe d'objets 3D à l'aide de dispositifs de localisation 2D. Des techniques ont été développées pour les interactions tactiles, par exemple Au et al.[9] qui propose l'interprétation de gestes multitouch, le déplacement d'objets étant contraints par des guides lors de manipulation pour augmenter la précision du positionnement en fonction la scène 3D.

Dans le contexte de la simulation interactive en mécanique des fluide, les techniques de sélection doivent être associées à des techniques de visualisation adaptées incluant transparence et feedback de sélection pour rendre possible l'interaction avec les éléments de la simulation, leur manipulation et leur édition.

1.4 Des simulations numériques aux jeux sérieux

Les simulations interactives, en particulier dans le domaine scientifique ont permis de voir émerger les approches par jeu sérieux, avec en général de objectifs pédagogiques, mais aussi scientifiques comme l'illustre le jeu *Fold'it* [78]. Les jeux sérieux dans l'éducation comme la recherche, ont été appliqués à de nombreux domaines, de la biologie à la physique. Dans les jeux sérieux ce n'est pas l'exclusif divertissement qui est ciblé, même si l'aspect ludique est toujours recherché, mais la résolution d'une problématique ou la formation, par exemple un jeu sérieux est utilisé pour aider les étudiants qui ont des difficultés à apprendre la programmation[107]. Les jeux sérieux sont en effet particulièrement utilisés pour des objectifs de formation ou d'apprentissage, permettant de s'adapter aux spécificité de chaque apprenant, prenant en compte les aspects cognitifs ou permettant d'augmenter la motivation intrinsèque dans l'activité l'apprentissage[99].

1.4.1 Définition du jeu sérieux

Les jeux sérieux sont définis comme la technologie qui rassemble les jeux conçus pour l'entraînement, la formation, l'éducation, la recherche, ou la résolution d'une problématique. Le but du jeu sérieux est d'utiliser les avantages du jeu vidéo pour soutenir les utilisateurs dans les processus d'apprentissage, pour faciliter l'acquisition de nouvelles connaissances théoriques ou techniques[36].

Les différences entre jeu sérieux et jeu vidéo sont illustrés dans le tableau ci-dessous

	Jeux sérieux	Jeux ludique
La tâche et les expériences riches	La résolution de problème	Expériences riches
Le but	Les éléments importants pendant l'apprentissage	S'amuser
La simulation	Les hypothèses sont nécessaires pour des simulations exploitables	Processus de simulation simplifiés
La communication	Doit refléter une communication naturelle (i.e., imparfaite)	La communication est souvent parfaite

Figure 1.15: Comparaison entre jeux sérieux et jeux ludique : [142].

Les jeux sérieux sont souvent basés sur la simulation d'un phénomène, d'un processus ou d'une activité, et leur but principal n'est pas seulement le divertissement mais l'acquisition de compétences et de connaissances. Les objectifs de ces jeux peuvent aller de la publicité aux exercices d'entraînement militaire [6]. Il existe beaucoup d'application du jeu sérieux pour motiver des élèves[108][109][70]. Les compétences visées par l'usages de jeux sérieux sont variés et peuvent être classées comme suit [158] :

1. Cognitif : connaissances et compétences,
2. Motricité : acquisition et coordination,
3. Affectif : attitude et motivation,
4. Communicatif : communiquer, coopérer et négocier.

Pour les éducateurs, il permet de faire vivre aux élèves une expérience qui est souvent impossible de mettre en oeuvre autrement [6]. Par exemple, il est très difficile d'amener les étudiants sur les sites de puits de pétrole, encore moins sur les sites de puits extracôtiers ou d'observer l'évolution du fluide souterrain. Un jeu sérieux dans ce domaine, permet d'accéder à l'ensemble des connaissances et des processus d'extraction de pétrole et de gaz à l'étudiant, et même en

matière de prospection également possibles avec des jeux permettant l'analyse ludique de données sismiques. Quatre grands principes motivent le développement des jeux sérieux : [66]:

1. Rendre la science accessible,
2. Rendre la pensée visible,
3. Aider les élèves à apprendre les uns des autres,
4. Promouvoir l'apprentissage autonome.

1.4.2 Jeu sérieux à application militaire

L'armée utilise depuis très longtemps des jeux pour s'entraîner. Pour les militaires, l'utilisation du jeu sérieux permet de créer des simulations plus précises et intéressantes avec un coût nettement inférieur à celui des entraînements traditionnels. Les jeux sérieux permettent aux militaires de faire des simulations à moindre coût comparé à la mise en place de situations de formation sur le terrain. Il existe des jeux commerciaux qui ont été adaptés pour un usage militaire, par exemple *WarCraft*, *Doom*, *Close Combat* et *Operation flashpoint*[126][101].

Ainsi, il existe de nombreux exemples de jeu sérieux dans le contexte militaire. *SanTrain* est un jeu sérieux conçu pour entraîner les premiers secours militaires[45]. Ce jeu fournit une formation des soins aux blessés au combat tactique. Cela permet aux utilisateurs de comprendre les règles et les étapes liés aux sauvetages, les priorités pour la première minute après les blessures. Les objectifs de ce jeu sérieux sont de classer et de traiter les blessures qui pourraient mettre la vie en danger, d'éviter d'autres blessures pendant que les autres soldats valides continuent d'accomplir une mission militaire. Les scénarios inclus dans ce jeu proviennent de l'expérience des forces d'opérations spéciales américaines dans des scénarios de combat réels. Le traitement prioritaire et anticipé sur le champ de bataille permet d'économiser des vies de soldats blessés en danger de mort, d'où l'importance de cette formation pour les forces armées non seulement par du personnel médical, mais aussi par des soldats.

America's Army est un autre jeu sérieux qui propose d'assister les ressources humaines militaires à trouver et à recruter de nouveaux soldats volontaires, développé par l'armée américaine à la fin des années 1990[55]. Ce jeu a permis à l'armée de recruter des soldats avec seulement 15% du coût nécessaires dans d'autres programmes de recrutement. Le jeu permet par ailleurs d'aider à la formation des premiers soldats recrutés.[101]

1.4.3 Les jeux dédiés à la santé et aux applications médicales

Comme pour les simulations interactives, il existe désormais un nombre croissant d'applications de jeux sérieux liées à la santé, notamment comme support à l'enseignement, la formation ou la prévention dans le domaine de santé. C'est le cas par exemple de *CatCare*[80], un jeu sérieux qui propose une solution pour le problème de l'hygiène des mains, première source d'infections à l'hôpital. En mettant en valeur les contaminations des mains après manipulations, le jeu sérieux est rendu ludique par des scénarisations basées sur la manipulation d'animaux domestiques, pour apprendre les bons réflexes aux employés de l'hôpital en matière de nettoyage de mains.

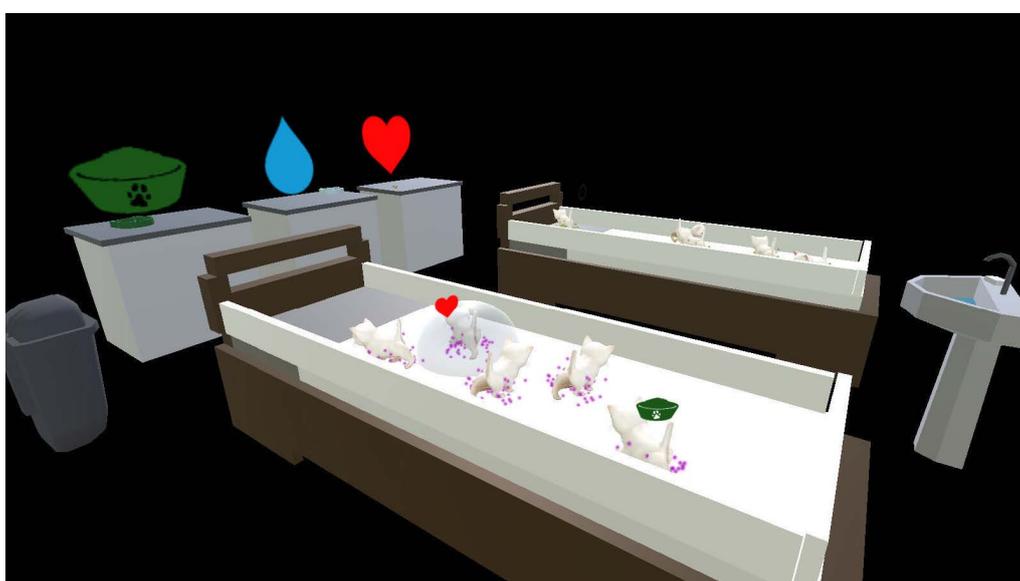


Figure 1.16: Aperçu de l'environnement 3D du jeu sérieux *CatCare*: [80].

D'autres jeux sérieux ont été développés pour entraîner les infirmiers[73] sur les techniques pour soigner les patients atteints de bronchopneumopathie chronique obstructive (BPCO) dans le contexte de soin à domicile ou à l'hôpital. L'utilisation du jeu sérieux permet les élèves en infirmière de s'entraîner à la technique et assister la prise de décision dans un environnement réaliste et sécurisé[30].

Rahouti, Anass, et al. ont par ailleurs développé un jeu sérieux pour enseigner au personnel de centres de santé les procédures en cas d'incendie[121]. De 2011 à 2015 il y avait environ 5750 incendies au centre de santé aux états unis tous les ans. Ces incendies ont causé en moyenne deux morts, 157 blessés et 50,4 millions de dollars de dommages directs chaque année[29]. Ce jeu utilise la Réalité Virtuelle pour simuler un hôpital réaliste. Il a été démontré que l'usage de jeux sérieux immersifs est plus efficace que les formations classiques (diapositive) pour améliorer les connaissances des participants sur les règles de sécurité incendie ainsi que la

restitution des connaissances acquises.

1.4.4 Les jeux sérieux pour promouvoir l'activité physique et le sport

D'autres jeux sérieux ont été développés pour promouvoir l'activité physique et la pratique du sport. C'est le cas par exemple de *Dance Dance Revolution*[42], jeu sérieux ayant pour objectif d'améliorer la santé par l'activité physique. Ces auteurs proposent l'augmentation de dispositifs physiques classiques, comme des tapis de danse ou des vélo stationnaires, par l'ajout d'éléments virtuels ludiques issus des jeux vidéo, ou d'intégrer dans les jeux vidéo des éléments physiques, montrant qu'il permettent d'augmenter la motivation lors d'exercice physique [101].

Le développement des jeux sérieux dans le domaine de santé reste complexe parce que Le développement de jeux nécessite une expertise en médecine, en éducation et en développement technologique, lors des étapes de conception, de création de modèles 3D, de réalisation technique, afin de créer un jeu sérieux qui réponde de manière efficace aux objectifs fixés. [113]. Devant cette complexité, les auteurs ont donc proposé un cadre de développement itératif et un guide structuré dédié à la conception et au développement de jeux sérieux.

1.4.5 Jeux sérieux dédiés à l'exploitation du pétrole et du gaz

L'usage des jeux sérieux émerge également dans le domaine pétrolier. Actuellement, malgré les réserves de pétrole qui restent importantes dans la couche pré-saline profonde en mer, la majeure partie des ressources proviennent d'exploitation terrestre, moins coûteuse. Afin de maintenir la cadence de production pour répondre à la demande, des efforts substantiels en termes d'investissement, d'exploration et d'exploitation sont incontournables, même dans les scénarios les plus optimistes de baisse de la consommation pour réduire les émissions de CO₂. Au delà de l'exploration, une partie de ces ressources est investie dans la formation du personnel, étant donné le risque élevé intrinsèque aux nombreuses activités liées aux processus de forage et d'exploitation des puits. Par conséquent, il est stratégique d'investir dans des formations continues efficaces pour aider à réduire considérablement les erreurs et les accidents.[25]

L'exploitation des réserves pétrolières et de gaz est aussi confrontée au défi de réduire les coûts pour rester compétitif face à des énergies émergentes. Les technologies de calcul haute performance, de visualisation et de collaboration à distance sont de plus en plus utilisées dans cet objectif [130]. Des nouvelles technologies telles que la Réalité Virtuelle permettent aux gens d'interagir avec l'environnement virtuel pour s'approcher le plus possible du geste technique afin d'acquérir les compétences nécessaires en utilisant la multimodalité intégrant rendus visuels,

audio et tactile [89], avec des coûts de formations inférieurs à ceux de formation sur site.

Dans cet objectif, des applications dédiées au domaine pétrolier et gazier simulent finement les équipements et l'environnement de travail, notamment dans les environnements difficiles sous la surface. Ainsi Brasil, Igor Saraiva et al.[25] ont développé un jeu sérieux pour former les travailleurs à l'exploitation des plates-formes de forage pétrolier à terre, avec l'intelligence artificielle et les techniques de Réalité virtuelle. Ce système peut simuler l'ensemble du processus de forage. En exploitant la plate-forme interactive de forage modélisant des puits Virtuels, il est possible d'évaluer les compétences les travailleurs sur des tâches métier spécifiques dans un environnement de travail proche de la réalité. Ce système est très efficace pour la formation des travailleurs, permettant notamment de réduire considérablement les erreurs et accidents sur site.

D'autres simulations ont été réalisées dans le même objectif, centrées pour simuler plus spécifiquement les équipements d'extraction de pétrole. La production pétrolière étant de plus en plus complexes étant donné l'évolution continue de la technologie, cette recherche vise à intégrer toutes les matérielles nécessaires à d'extraction de pétrole et de gaz. Avec cette plate-forme, le processus d'extraction de pétrole et de gaz peut être mieux anticipée au delà de la formation. Cette plate-forme fournit un environnement virtuel intégré du site de production de pétrole, permettant aux utilisateurs de voir et contrôler et de s'entraîner sur une plateforme 3D modélisant finement les puits de forage dans leur environnements[90].

L'environnement de travail peut également être simulé dans un objectif de sensibilisation ou de formation à la sécurité, un enjeu crucial dans ce secteur industriel. Ces plateformes amènent des stagiaires à être confronté et à se familiariser à des circonstances particulières et à des dangers scénarisés fréquemment rencontrés dans les environnements pétroliers. Les stagiaires apprennent ainsi les stratégies et les procédures d'urgence qui doivent impérativement être bien maîtrisées par les personnels.

S'il existe depuis longtemps des formations multimédia, les technologies du jeu et de la Réalité Virtuelle ont ouvert de nouvelles possibilités, en termes d'interaction ou d'immersion, afin de rendre de plus réaliste l'environnement d'apprentissage, permettant de minimiser le recours à des formations traditionnelles coûteuses et qui ne peuvent pas toujours répondre aux objectifs remplis par les formation virtuelle [4] [131]. Ainsi les formations numériques des personnels dans le domaine de pétrolier se rapprochent de plus en plus d'approches de type jeu sérieux, et sont de plus en plus répandue, notamment dans le contexte de plateforme pétrolière offshore [116]. Au delà du fait de fournir une approche de formation alternative aux personnes qui travaillent des environnements hostiles et isolés que constituent les plateformes pétrolières offshore, au delà de former aux procédures de sécurité et aux procédures d'urgence, au delà de permettre

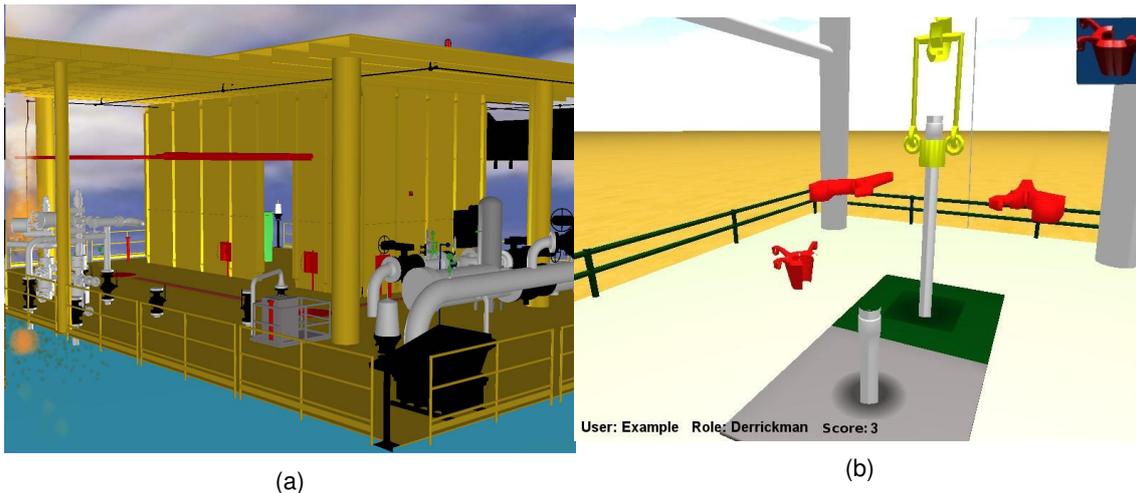


Figure 1.17: Jeux sérieux dans le secteur pétrolier : Incendie sur la plate-forme. Le stagiaire doit répondre au feu, un danger majeur sur les plateformes pétrolières (a) [116]. Interface du jeu pour s'entraîner la manipulation des outils (b) [25].

de se familiariser avec l'espace de travail, ces plateformes donnent aussi accès à une gamme d'information de plus en plus large et intégrées, comme l'accès à des bases de données, à la documentation de maintenance, ou à des applications extérieures.

Pour conclure, la croissance de l'usage de formations numériques réalistes et se rapprochant du jeu sérieux, prend sa source dans l'enjeu majeur du domaine de réduire le temps et le coût de formation aux procédures techniques complexes à réaliser dans des environnements hostiles et isolés. C'est un secteur d'activité où la sécurité [117] est un enjeu majeur, le danger étant tel que le premier objectif affiché par les industriels est de diminuer le nombre de décès ou de blessés au travail.

1.5 Les simulations interactives en Réalité Virtuelle

La Réalité Virtuelle peut être définie comme un environnement numérique interactif [28] qui suppose :

- une simulation en temps réel d'environnement artificiel
- l'immersion sensori-motrice de l'utilisateur dans un monde virtuel,
- une interaction de l'utilisateur *via* plusieurs canaux sensori-moteurs.

Les environnements virtuels permettent de présenter à l'utilisateur des informations de manière multisensorielles et multimodales, supposées plus riches et plus complexes que ceux envoyés

exclusivement par le canal visuel. L'immersion et la multi sensorialité peuvent susciter une perception profonde de réalisme, un fort sentiment de présence dans l'environnement virtuel, ou une sensation d'incarnation dans un personnage virtuel, en faisant oublier à l'utilisateur la nature artificielle du dispositif[125]. La première implémentation enregistrée d'un système de Réalité Virtuelle numérique est apparue en 1966, sous la forme d'un simulateur de vol conçu pour l'entraînement des forces aériennes des États-Unis[114]. Ensuite, Les applications sont restées principalement limitées au secteur public puis à quelques grands comptes pendant plusieurs décennies du fait d'investissements nécessaires très lourds, avant les premières tentatives de démocratisation au début des années 90, débuté par la sortie d'une série de jeux d'arcade spécialisés lancée par la société *Virtuality Group*[81]. Des études de plus en plus nombreuses sur l'utilisation de la Réalité Virtuelle ont permis d'entrevoir les avantages de tels dispositifs d'interaction, permettant d'augmenter la performance [68] ou la qualité de l'expérience utilisateur [7], l'entraînement des techniciens pour l'industrie[22], etc.. La Réalité Augmentée est aussi appliqué dans les différent domaines, par exemple *AMRA* est une application de Réalité Augmentée pour l'entraînement de la maintenance des trains[43]. Même si l'usage de la Réalité Virtuelle reste encore peu répandu aujourd'hui en dehors de l'industrie du jeu vidéo, les technologies de la Réalité Virtuelle sont cependant de plus en plus mobilisées dans les domaines de santé, de l'éducation, de la science et l'ingénierie, progression notifiée par une analyse fine de la littérature du domaine. [77].

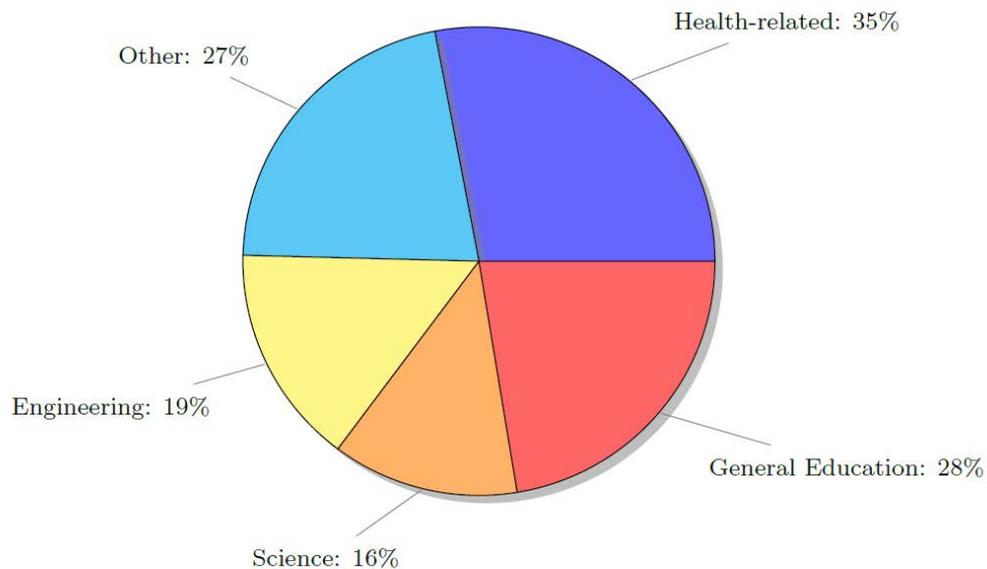


Figure 1.18: Domaines d'application de certains articles au cours de deux analyses thématiques [77].

1.5.1 La multimodalité

Au delà de la multi-sensorialité, la multimodalité permet au concepteur de combiner de manière séquentielle ou simultanée plusieurs canaux afin de transmettre des informations à l'utilisateur. La combinaison de plusieurs canaux décuple les possibilités pour transmettre plus efficacement une information en fonction de sa nature. La multimodalité permet aussi d'augmenter la *bande passante* entre la machine à l'être humain en allant au delà des limites perceptives imposés par l'usage exclusif du canal visuel.

L'utilisation de la multimodalité en Réalité Virtuelles est assez répandue, notamment dans les applications dédiées à l'analyse de phénomènes scientifiques. *ViSTA FlowLib* [132] utilise notamment des rendus haptiques permettant à l'utilisateur de ressentir physiquement les forces en jeu dans un écoulement afin d'améliorer son expérience et de faciliter sa compréhension des données de fluides instables. Un concept "proxy haptics" est utilisé pour l'entraînement de la chirurgie, où les supports réels sont utilisés qui correspond aux modèles virtuels dans l'environnement virtuel[61]. Un autre travail[151] a situé la multimodalité au cœur d'applications immersives dédiées à l'analyse de résultats scientifiques, en concevant et en évaluant une large gamme de rendus multimodaux, de l'haptique à la sonification de données pendant l'exploration de données de simulation de fluide, ou pendant une tâche de docking moléculaire protéine/protéine.

1.5.2 Applications de Réalité Virtuelle pour l'éducation

La Réalité Virtuelle offre aux utilisateurs la possibilité de participer à des expériences virtuelles inédites, impossibles à concevoir, inaccessibles, trop coûteuses ou trop dangereuses si elles étaient réalisées dans un contexte réaliste [1]. Le potentiel de la Réalité Virtuelle dans le domaine de l'éducation est immense, comme le démontre le travail de Perez-Valle & Sagasti en développant un système virtuel permettant aux élèves d'être immergé au 16e siècle en Espagne. Les élèves peuvent ainsi explorer des lieux qui n'existent plus, y compris les anciens temples tels qu'ils étaient il y a des centaines d'années ou observer les travaux des ouvriers et des architectes pour construire les structures historiques. Et tout cela depuis leur salle de classe [118].

En plus de pouvoir simuler des explorations dans des environnements virtuels, il est possible au concepteur de proposer aux utilisateurs des moyens d'interaction avec le monde virtuel, permettant d'envisager une infinité de scénarios d'apprentissage par essai/erreur ou de jouer des scénarios dans des environnements à risque qu'ils seraient impossibles à mettre en place sans la Réalité Virtuelle. C'est ce que souligne Sharma et Otunba qui ont développé un système permettant aux agents de bord de simuler des exercices d'incendie d'aéronefs. Ils insistent

sur l'importance de la Réalité Virtuelle pour créer des exercices d'incendie afin de s'entraîner à réagir à des situations de dangers, impossibles à reproduire dans la réalité du fait de risques de blessures trop important en cas d'exercices réels [134].

Pour des raisons similaires, la Réalité Virtuelle est de plus en plus utilisée en matière de formation dans le domaine médical, notamment chirurgical, l'entraînement sur un patient vivant n'étant possible qu'en fin de formation du chirurgien [52]. Djukic, Tijana et al. ont proposé une méthode qui permet aux médecins et aux étudiants de visualiser et de manipuler des modèles tridimensionnels créés à partir de tomodensitométrie ou d'IRM, et également d'analyser les résultats de simulations d'écoulement de fluide. La simulation de fluide est destinée à calculer la contrainte de cisaillement sur les parois des artères. Le système de Réalité Virtuelle peut raccourcir la durée des programmes de formation et rendre le processus éducatif plus efficace.[44]

Les applications de la Réalité Virtuelle ont été consacrées à d'autres activités médicales telles que la rééducation [112], ou pour créer des environnements pour faciliter la réadaptation des enfants atteints de troubles de l'attention.

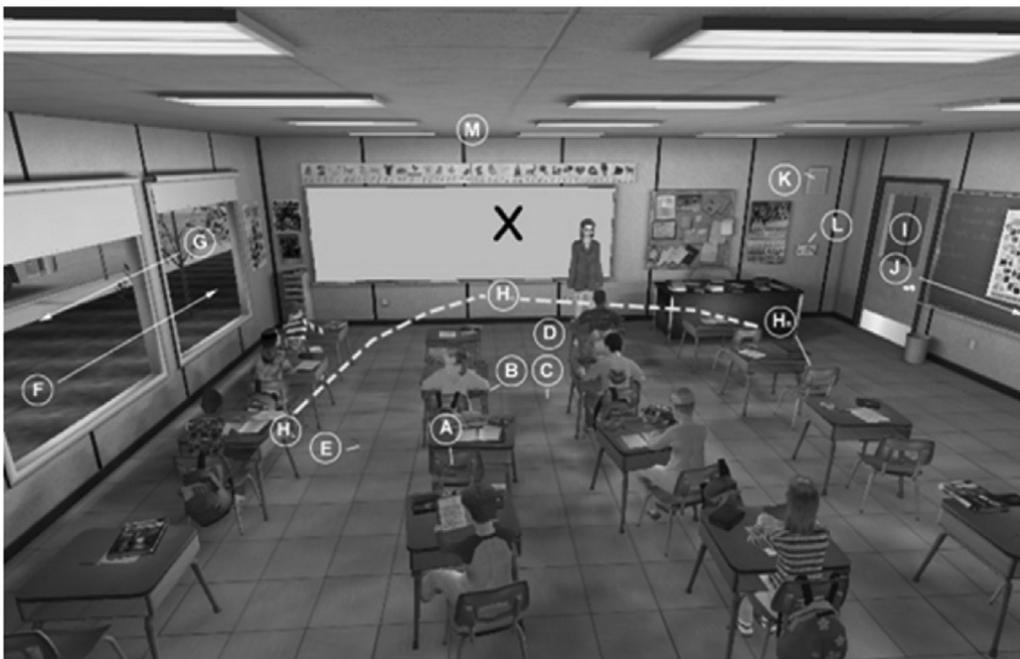


Figure 1.19: *ClinivaVR*: Salle de cours et emplacement des distracteurs [112].

Si l'un des usages de la Réalité Virtuelle est de simuler des activités irréalisables dans un contexte réel, elle est également utilisée pour simuler des situations nécessitant habituellement l'accès à des ressources rares coûteuse, ou limitées. Les ressources limitées peuvent inclure non seulement des ressources au sens traditionnel du terme, mais aussi des équipements scientifiques et même les laboratoires qui les contiennent. C'est le cas par exemple de la simu-

lation d'une usine de transformation commerciale de lait en poudre, conçu pour les élèves en génie chimique et en génie des procédés. La simulation permet aux élèves de visiter les usines virtuelles, les usines devenant de plus en plus difficiles à visiter en raison de la disponibilité et des réglementations de sécurité [1].

De même, Hristov et al. ont développé un système permettant aux élèves de manipuler virtuellement du matériel de laboratoire grâce à l'utilisation d'un environnement virtuel, permettant de limiter de manière drastique les ressources nécessaires. Ce système leur permet par ailleurs d'être accessible pendant de plus longues périodes, et à distance [67].

En matière de pédagogie, des avantages de l'usage de la Réalité Virtuelle pour les apprentissages en travaux pratiques ont été soulignés. Vergara, Diego, et al. ont développé un système basé sur Réalité Virtuelle pour l'enseignement de l'ingénierie matériaux. Ils ont observé que les étudiants réalisant les enseignements pratiques en Réalité Virtuelle avaient des notes très supérieures aux autres. L'explication avancée par les auteurs est que le contexte de Réalité Virtuelle permet de mieux cibler l'attention des étudiants que lors des expérimentations réels tout en limitant l'accès aux distracteurs, ayant pour effet une meilleure compréhension des étudiants des explications autour de l'expérimentation. [150]

LakeViz3D est un projet pour améliorer la compréhension de l'écosystèmes du lac d'eau douce à l'aide de visualisations 3D. Un bac à sable en réalité augmentée est fabriqué pour modifier le modelé du terrain puis afficher automatiquement des courbes de niveau virtuelles et un modèle d'écoulement d'eau créés à l'aide d'une caméra à balayage 3D[123][122]. La simulation de fluide utilisé dans ce système est basé sur la méthode des équations de Barré de Saint-Venant[157] qui est souvent utilisé pour la simulation de la rivière. Cette méthode suppose que l'échelle de vitesse verticale du fluide est négligeable par rapport à l'échelle de vitesse horizontale[39].

Enfin, la Réalité Virtuelle constitue aussi une possibilité pour se former à des tâches ayant dimension collaborative. Schild, Jonas, et al. ont ainsi développé un outil de formation dans ce sens, permettant à deux stagiaires de s'entraîner à un scénario d'urgence paramédical sur le choc anaphylactique améliorant les résultats d'apprentissage des techniques classiques.

1.5.3 Réalité Virtuelle dans le domaine pétrolier

Comme pour les jeux sérieux, et du fait d'une exigence de formation liée à des risques industriels importants, les applications de la Réalité Virtuelle au secteur de l'industrie pétrolière sont de plus en plus utilisées en matière de formation, mais aussi de prospection. En effet, dans l'industrie pétrolière et gazière, une grande quantité de données est récupérée lors de l'exploration de nou-

veaux réservoirs. La taille de l'écran de l'ordinateur de bureau montre très rapidement ses limites en matière la visualisation de ce type de données. La collaboration entre différents experts de différents domaines pour mieux analyser les données, ce qui mène directement au besoin de disposer de larges dispositifs de visualisation offrant un large espace de travail permettant la collaboration. Ce type de dispositif de visualisation large, doit donner accès au caractère tridimensionnel intrinsèque des données analysées, et ceci pour chacun des utilisateurs et selon son propre point de vue, ce qui est le défi technologique relevé par Fröhlich, Bernd et al. en proposant une technologie d'environnement virtuel stéréoscopique multiutilisateur inédit permettant ce travail collaboratif. En outre, ils ont proposé un nouveau dispositif d'interaction qui est parfaitement adapté à la nature de la tâche d'exploration de données géoscientifiques [53].

Enfin, s'il était besoin de démontrer l'intérêt que porte le secteur industriel aux technologies de la Réalité Virtuelle, à l'Université Laurentienne, un Laboratoire de Réalité Virtuelle a lancé le programme du Centre de technologie de veille dédiée à l'exploration minérale, des mines et de la géotechnique. Il s'agit de «la première installation de Réalité Virtuelle entièrement dédiée au secteur de l'industrie minière»[76].

1.6 Conclusion

Nous avons dans ce chapitre introduit le contexte de ce travail de doctorat avec une présentation générale des approches de simulations interactives, de jeux sérieux et de Réalité Virtuelle, en introduisant quelques notions clé ainsi que des éléments de l'état de l'art permettant de caractériser ces approches. Nous avons dressé un panorama non exhaustif d'exemples sélectionnés afin d'illustrer les applications de ces approches dans l'industrie et dans la vie quotidienne.

Comme illustré dans la figure ci-dessous, les problématiques liées à l'usage de simulation interactive de fluide dans les jeux sont variées et pluridisciplinaires.

C'est dans ce contexte que se situe notre motivation d'explorer l'usage et l'application des approches de simulation interactive au domaine de la mécanique des fluides, avec des objectifs notamment d'aide à la prise de décision, travail décrit dans les chapitres suivants.

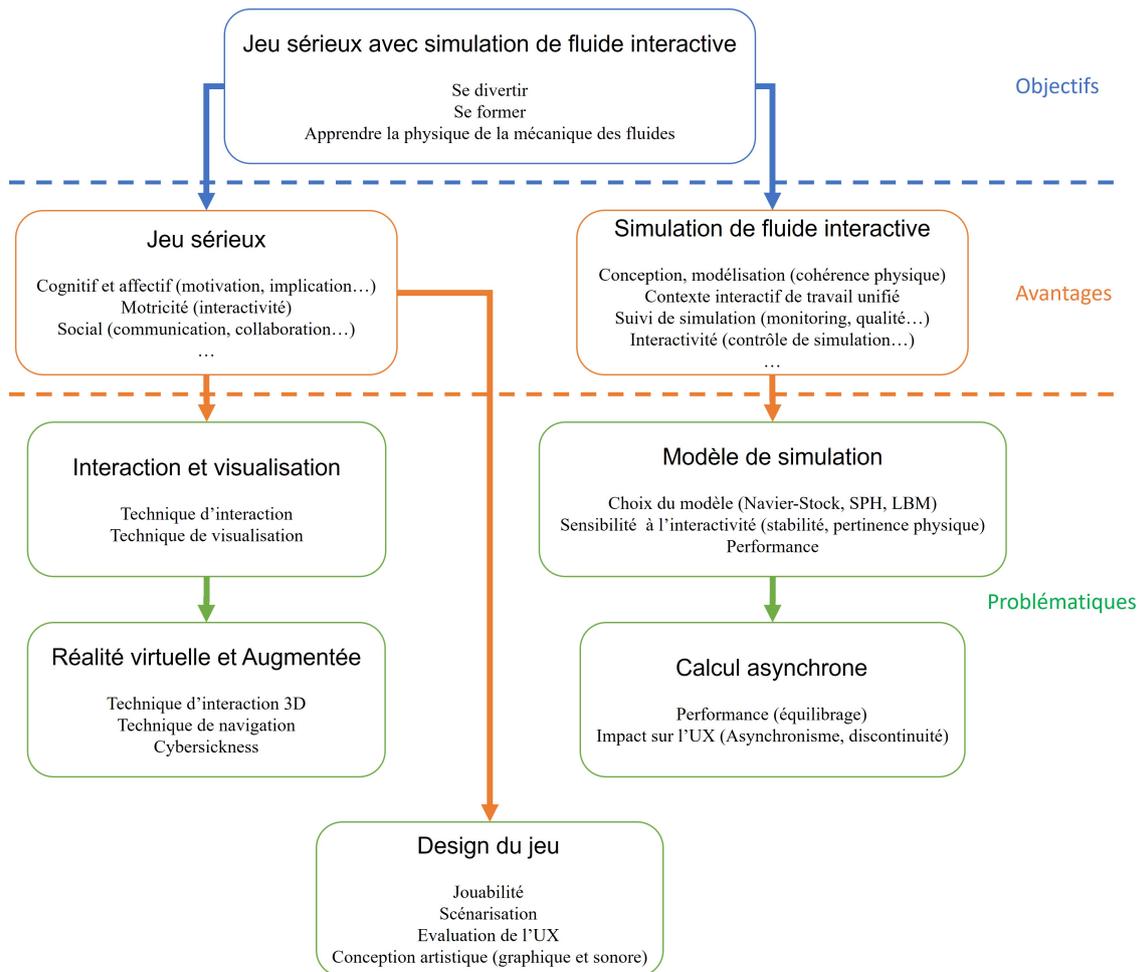


Figure 1.20: Aperçu des objectifs, avantages et problématiques des jeux sérieux dédiés à la simulation interactive de fluide : vue globale. SPH : hydrodynamique des particules lissées, LBM : Méthode de Boltzmann sur réseau, UX : Expérience utilisateur

Une architecture interactive, immersive et distribuée dédiée à la mécanique des fluides dans un contexte de jeu sérieux

2

Résumé

Ce chapitre expose les choix et les contributions de ce travail de doctorat pour mettre en place une plateforme expérimentale de simulation interactive en mécanique des fluides. Nous décrivons notamment de choix d'aller vers un couplage faible entre les composants de calcul et les composants de visualisation et d'interaction.

Quel que soit le domaine d'application, en matière de simulation interactive, au sens où nous l'avons défini dans le premier chapitre, trois fonctionnalités essentielles correspondantes à des composants logiciels distincts, doivent être mises en place pour permettre à un utilisateur de contrôler une simulation en cours de calcul. D'abord la partie calcul qui permet de simuler numériquement le phénomène physique, ensuite la partie visualisation qui permet de rendre graphiquement le résultat de la simulation en cours, et enfin la partie interaction qui permet à l'utilisateur de contrôler de manière interactive le phénomène simulé en modifiant des paramètres ou les éléments de la simulation. Les fonctionnalités de visualisation et d'interaction sont intrinsèquement liées et sont en général conçues et implémentées dans un seul composant logiciel. C'est donc sur l'articulation entre la fonctionnalité de calcul et les fonctionnalités de visualisation interactive que les approches en matière de simulation interactive se différencient.

2.1 Couplage fort *versus* couplage faible

Les approches par simulation interactive se démarquent essentiellement entre le choix qui est fait entre un couplage faible, c'est-à-dire une indépendance logicielle importante entre les modules de calcul et de visualisation et d'interaction, et un couplage fort. On parle de couplage fort quand le code de calcul est très proche, voire indissociable du code de visualisation et d'interaction. C'est typiquement le cas de nombreuses approches visant une performance maximale nécessaire pour l'interaction comme dans le travail de Florian De Vuyst et al. [40] dans lequel les processus de simulation et de visualisation sont effectués sur la même unité graphique. Ces approches permettent une performance optimale, notamment du fait d'absence d'échanges de données entre l'unité graphique et le processeur central, ce à quoi s'ajoute une exploitation adaptée du parallélisme qui décuple les performances.

Par ailleurs, ces approches par couplage fort sont particulièrement adaptées pour la conception et l'implémentation de jeu sérieux en dynamiques des fluides, car ces développements de code sur GPU permettent désormais une intégration relativement facile dans les écosystèmes de développement de jeu vidéo, dans les technologies internet, dans les applications pour smartphone, ou bien dans les plateformes professionnelles de développement de jeux vidéo comme *Unity 3D* ou *Unreal*.

Cependant, ces approches peuvent ne pas rencontrer l'adhésion des industrielles ou des chercheurs du domaine, dans la mesure où les utilisateurs sont très attachés à leur code historique et à leur méthodologie de calcul. Les études et recherches connexes à la mécanique des fluides suppose une maîtrise approfondie des outils de simulation, une garantie de qualité de résultat,

l'appui d'une communauté d'utilisateurs. Ces exigences sont assurées soit par la réalisation d'outil interne donc bien maîtrisé, soit par l'usage d'outil de simulation "sur l'étagère", validé par les communautés académiques ou professionnelles ou commercial.

Pesant les avantages et les inconvénients de chaque approche, nous avons fait plusieurs choix. D'abord, celui d'un couplage faible, dans l'objectif d'être capable d'intégrer et d'adapter n'importe quel code de simulation existants à notre plateforme de la simulation interactive. Nous nous sommes inspirés pour appliquer au domaine de la simulation de fluide, des travaux effectués en simulation moléculaire interactive. Cette communauté intègre en effet depuis longtemps dans ses codes de calcul historiques, des fonctionnalités permettant de connecter interactivement les outils de simulations les plus connus (Gromacs, NAMD...) aux outils de visualisation moléculaires usuels [139] [41].

Ensuite, concernant les choix d'implémentation des modules de visualisation et d'interaction pour notre plateforme, ciblant à la fois comme application, la simulation interactive, les jeux sérieux, des outils d'aide à la décision, nous avons choisi d'effectuer nos travaux avec l'outil de développement de jeux vidéo *Unity 3D*. Cet environnement met à disposition des développeurs tout l'écosystème nécessaire pour prototyper, développer et étendre rapidement les fonctionnalités de visualisation et d'interaction nécessaires pour notre plateforme de simulation interactive de fluide.

2.2 Modalités de couplage entre les modules de calcul, de visualisation et d'interaction

2.2.1 Limites du couplage faible par système de fichier - retour d'expérience entre *Fluent* et *Unity 3D*

Les premiers développements réalisés ont consisté à effectuer des premiers tests de couplage naïf, avec la simulation de fluide *ANSYS Fluent* pour mesurer la faisabilité d'une visualisation en temps interactif de résultat d'une simulation. *ANSYS Fluent*, qui n'est pas open source, peut cependant exporter des fichiers de résultats à chaque étape de simulation dans un format simple (voir figure 2.1). Les fichiers sont générés après certain nombre d'itérations décidées initialement par l'utilisateur, ce qui permet de choisir une fréquence de mise à jour plus faible que la fréquence de calcul.

Ces premiers tests ont permis d'évaluer et de valider différentes modalités de rendus 3D coté

visualisation dans *Unity 3D*, avec une simple lecture à une fréquence déterminée afin de présenter à l'utilisateur le résultats du calcul en cours. Un certain niveau de simulation en temps interactif a ainsi pu être mise en place, sans fournir de possibilité de modifier la simulation. Cependant, il est important de noter que cette simple fonctionnalité, c'est à dire d'être en mesure de se connecter pour visualiser à la demande une simulation en cours est la première fonctionnalité essentielle en simulation interactive. Les simulations off-line peuvent en effet être longues, il arrive très fréquemment, dans tous les domaines d'application, qu'il y ait des erreurs dans les paramètres ou bien dans les données d'initialisation. Ces erreurs doivent pouvoir être détectées le plus tôt possible pour éviter de gaspiller du temps de calcul, notamment à cause d'une prise de conscience trop tardive, ou d'erreur dans le paramétrage de la simulation. S'il est possible de mettre en place des contrôles automatiques réguliers pour s'assurer du bon déroulement d'une simulation et anticiper certains problèmes, mettre l'humain dans la boucle grâce à la visualisation reste la modalité la plus efficace car toutes les erreurs ne sont pas prévisibles.

Cette première approche préliminaire ne passe évidemment pas à l'échelle, les données étant stockées sur disque, non compressées, dans un format incompatible avec la performance attendue en simulation interactive. Sur les simulations réalisées avec cette approche, il faut par exemple 0,1 à 1 seconde par itération, selon le réseaux et le format du fichier pour charger les données de taille environ 1 Mo, ou 100 mille cellules, en format *txt* ou *vtk*, pour simulation en 3D, ce qui est évidemment rédhibitoire pour envisager toute intervention interactive de l'utilisateur.

Par ailleurs, pour l'étape suivante une simulation interactive à savoir de contrôler la simulation pendant la simulation sans recommencer par la condition initiale, nous soulignons que *ANSYS Fluent* peut difficilement être utilisé à cette fin. En effet, choisi dans un premier temps du fait que nous en avons une bonne maitrise, ce n'est cependant pas un outil *open source*, et il ne fournit pas d'API qui permettrait une édition des paramètres de simulation ou du contenus simulé pendant une simulation en cours.

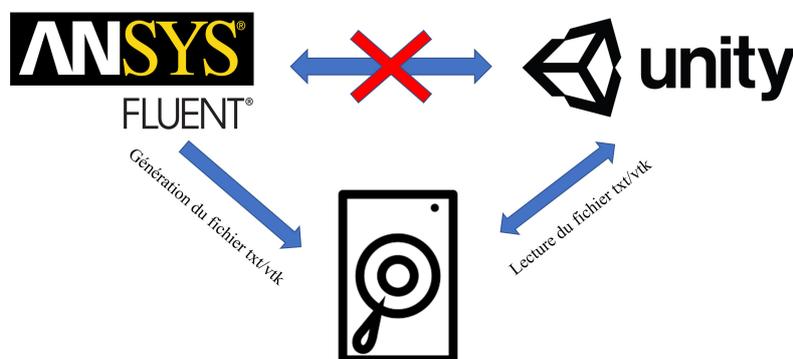


Figure 2.1: Couplage naïf entre ANSYS-Fluent et Unity 3D par système de fichier

2.2.2 Importance des systèmes de coordonnées et des normes dans les différents composants

Avant tous les travaux consistant à faire communiquer simulation et visualisation, en particulier dans le cadre de couplage faible, cela implique d'être conscient que certains choix techniques sont différents, notamment en matière de référentiel. Il est important dès le départ, lorsque qu'il s'agit de mettre ensemble module de visualisation et module de calcul, d'être conscient des choix effectués en la matière, des deux côtés. Ces choix peuvent aussi concerner des données susceptibles d'être échangées, décrite selon des normes différentes. C'est le cas par exemple des quaternions, dont la composante \mathbf{w} est située soit en début soit à la fin du vecteur. Ces considérations même si elles vont de soi et sont d'un point de vue théorique triviale, sont cependant d'autant plus importantes que les choix peuvent ne pas être explicites.

Dans notre plateforme, *Unity 3D* utilise par exemple un système de repère indirect, dit "main gauche", alors que la plupart des outils de simulation de fluides utilisent des repères directs dit "main droite". Cela permet d'effectuer les traitements triviaux nécessaires comme les changement de référentiel et les transformation de système de coordonnées, pour conserver la cohérence entre le phénomène simulé et le phénomène visualisé et manipulé.

$$M_c = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (2.1)$$
$$P = M_c \cdot P_0$$

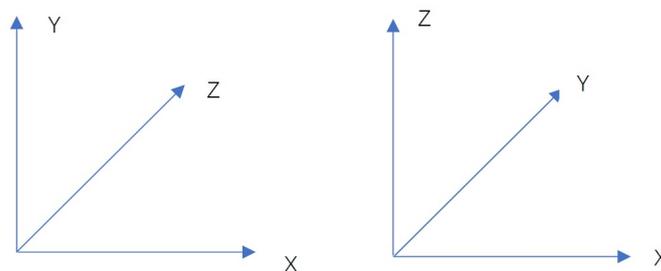


Figure 2.2: Opération de transformation pour passer d'un repère indirect à un repère directe

2.2.3 Couplage faible par architecture distribuée entre module de simulation et d'interaction/visualisation

Pour notre approche de simulation interactive de fluide, nous avons conçu et implémenté un système basé sur une architecture distribuée. Une machine est dédiée à la simulation et l'autre exécute une scène 3D comprenant des fonctionnalités de visualisation et d'interaction basées sur *Unity 3D*.

Contrairement aux approches par couplage fort, il est donc nécessaire de faire communiquer deux modules à travers le réseau. Nous avons donc conçu un protocole réseau *ad-hoc* adapté aux données à échanger, associée à l'API spécifique pour permettre d'appeler ces fonctions de communication depuis les composants de simulation et de visualisation/interaction. Cette infrastructure permet d'étendre de manière indépendantes les fonctionnalités des outils de simulation s'il s'agit d'outils de laboratoire, et celles des outils de visualisation et d'interaction dans le cadre de notre plateforme de simulation interactive.

Il existe de nombreuses approches pour visualiser les résultats d'une simulation en cours, comme celle offerte par *Paraview* qui met à disposition un cortège de composants logiciels pour connecter le composant de visualisation *Paraview* avec des outils de simulations moyennant une encapsulation préalable des données de simulation avec le framework *VTK*. Cependant la plupart de ces approches ne permettent pas une communication dans les deux sens, c'est à dire de pouvoir contrôler une simulation en cours, sont très rares, et les outils de visualisation ne fournissent donc pas de fonctionnalités de contrôle et d'édition de la simulation.

Nous rappelons donc que l'objectif principal de notre démarche, est de permettre aux utilisateurs d'interagir avec une simulation en cours en temps interactif, notamment de modifier les conditions aux limites, les paramètres de simulation, les paramètres physiques des éléments simulés, sans systématiquement arrêter la simulation. Dans certains cas, l'utilisateur doit cependant pouvoir redémarrer la simulation depuis le début si les nouveaux paramètres ou conditions aux limites l'exigent, avec de nouvelles conditions initiales pour garantir la continuité perçue du phénomène simulé. Au delà de l'échange des résultats de simulation, cela nécessite évidemment de prévoir un langage de commande embarqué dans le protocole réseau permettant le contrôle interactif de la simulation (changement de paramètres, de conditions aux limites, de valeurs physiques...). Ces commandes doivent pouvoir être interprétées par le module de simulation. C'est l'objet de la librairie *CFDriver* décrite dans la section suivante.

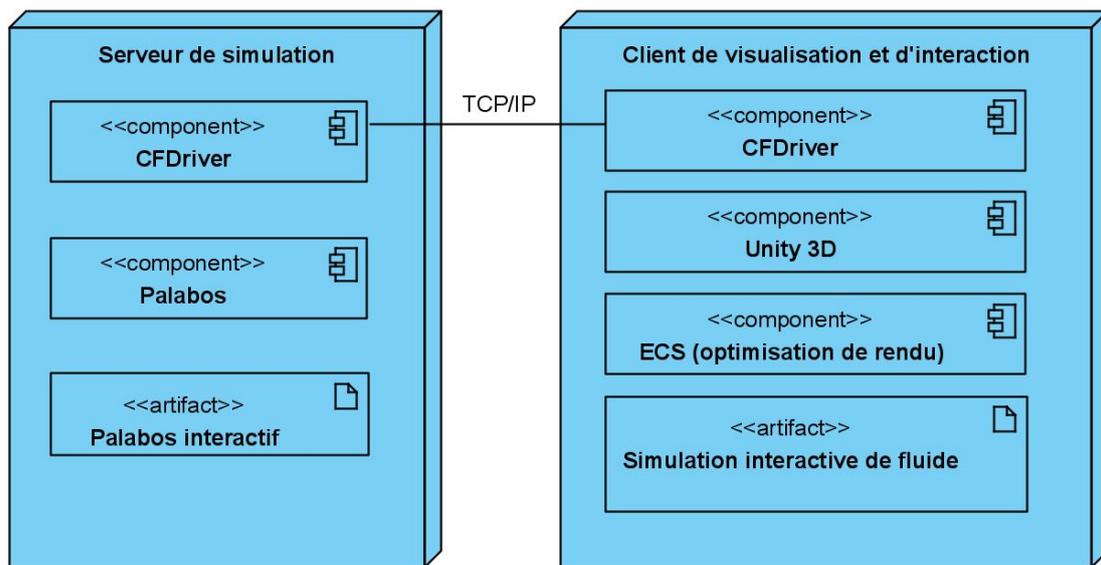


Figure 2.3: Diagramme UML de déploiement illustrant l'architecture de la plateforme de simulation interactive

2.2.4 *CFDriver*, une librairie pour faire communiquer composants de simulation et de visualisation/interaction

Pour pouvoir intégrer tous les outils de simulation utilisés par les experts, et pour garantir la qualité des résultats de la simulation en donnant la possibilité de s'appuyer sur des outils de simulation reconnues par la communauté, il est nécessaire de fournir une infrastructure logicielle pour pouvoir connecter les outils de simulation existants avec des outils de visualisation et d'interaction.

Nous avons conçu, développé, évalué et testé une librairie que nous avons appelée *CFDriver* [152] intégrant un protocole réseau pour assurer la communication de données entre un outil de simulation et un outil de visualisation/interaction. Cette librairie intègre un protocole réseau et un ensemble de commande regroupées dans une librairie logicielle permettant de recevoir les résultats de la simulation au niveau du composant de visualisation/interaction, et d'envoyer les commandes et les mises à jour des utilisateurs au serveur de simulation pendant la simulation depuis le composant de visualisation. C'est une librairie multiplateforme et elle fonctionne donc sur les systèmes d'exploitation usuels. Cette librairie a pour objectif de permettre à un expert de connecter n'importe quel outil de visualisation/interaction, avec n'importe quel outil de simulation à l'aide de quelques lignes de codes, ce qui suppose évidemment un code ouvert. Pour construire une plateforme de simulation interactive, il s'agit donc d'être en mesure d'interpréter des deux côtés les données reçues et les commandes interactives, en mobilisant les fonctionnalités en générale déjà disponible dans les outils de simulation ou de visualisation dédiés.

Cette architecture distribuée est adaptée à la mise en place de simulation interactive mobilisant les grands centres de calcul pour les grosses simulations de fluides, moyennant des aménagements de la politiques de sécurité souvent drastiques lors de l'usage de ces infrastructures.

Cette infrastructure distribuée est de type client/serveur et est basée sur TCP/IP, donc en mode connecté. Le serveur est la machine qui effectue la simulation, le client étant la machine exécutant l'outil de visualisation et d'interaction.

Cette infrastructure fournit à travers la librairie *CFDriver* un ensemble de fonctions instanciant un protocole que nous avons spécialement conçu pour l'échange de données en simulation de fluide et dédié à la transmission des commandes interactives effectuées par l'utilisateur.

Fonctions de connexion

Les premières fonctions fournies par la librairie sont évidemment les fonctions de connexion entre le client et le serveur :

Name	Fonction
init_server	Initialiser comme serveur à partir d'un port spécifié par utilisateur. Cette fonction attend la connexion d'un client
init_client	Initialiser un client et connecter à un serveur à partir de l'adresse et port du serveur

Fonctions d'échange et de gestion des commandes interactives

Une fois les deux machines connectées, le développeur dispose d'un ensemble de fonction pour mettre en place une simulation interactive de fluide. Une partie des fonctions permet d'assurer la communication pour récupérer les résultats de simulation calculé par le serveur, du côté du client de visualisation et d'interaction. Une autre partie des fonctions est de transmettre les commandes interactives de contrôle et d'édition de la simulation, pour que le serveur puisse mettre à jour ses données, et continuer la simulation en prenant en compte ces modifications.

Name	Fonction
StartSimulation	Envoyer les conditions initiales et commencer la simulation
ContinueSimulation	Envoyer les conditions aux limites modifiées et continuer la simulation en gardant l'état en cours du fluide
SendCommand	Envoyer une commande sous forme de chaîne de caractères à l'autre machine
RecvCommand	Recevoir une commande. Cette fonction non bloquante pour la simulation, et retourne 0 si rien n'est reçu
SendFile	Serialisation d'un fichier local à envoyer depuis le serveur de calcul accessible dont le chemin est spécifié en paramètre
RecvFile	Réception d'un fichier et stockage dans le chemin spécifié
SendData	Envoyer des données structurées d'une taille spécifiée
RecvData	Recevoir des données structurées, retourne la taille des données

Ces fonctions sont voulues les plus génériques possibles pour s'adapter à la grande variété des méthodologies de représentation des fluides et de calcul.

Fonctions d'optimisation

Il est souvent plus efficace de bénéficier de fonctionnalités offertes côté serveur dans le code de la simulation, notamment l'extraction de l'interface dans le cas de simulation multiphase. Cela permet une optimisation des échanges, cela permet d'éviter les calculs cotés visualisation et de rendre plus fluide l'interaction.

Nom	Fonction
GetVOF	Obtenir les résultats de Volume of Fluid
GetInterface	Envoyer une commande au serveur pour générer et envoyer l'interface de la simulation multiphase sous forme de maillage

2.2.5 Gestion des données structurées dans *CFDriver*

La donnée transmise entre le serveur et le client via *CFDriver* est totalement en binaire. Cela nous permet de déclarer une classe des données structurées. Pour faciliter la déclaration des données structurées différentes, nous avons déclaré une classe abstraite. Cette classe abstraite contient des fonctions pour l'ajout des composants, l'émission et à la réception des données. L'utilisateur doit simplement déclarer une classe encapsulant ses données et implémentant cette classe abstraite, l'obligeant à définir deux fonctions :

Name	Fonction
ToCFData	L'utilisateur doit ajouter toutes les variables membres dans la classe en appelant la fonction Add dans l'ordre dans lequel elles sont transmises sur le réseau
FromCFData	L'utilisateur doit extraire les données en appelant la fonction Get dans l'ordre

Une fois ces deux fonctions implémentées, l'utilisateur peut envoyer la classe simplement en appelant *Send* et *Recv*. Le receveur peut recevoir l'objet de la classe directement. Ces données structurées peuvent être envoyées et reçues sous forme de tableau.

Pour envoyer des données structurées tel que par exemple des vecteurs 3D (composé de 3 variables de type *float*, *x*, *y*, et *z*). L'utilisateur peut simplement déclarer le vecteur avec les variables *x*, *y* et *z* puis ajouter ces variables en appelant la fonction *ToCFData*. En suite l'utilisateur peut envoyer un vecteur ou un tableau de vecteur en appelant la fonction *Send*. Pour recevoir un vecteur il suffit de déclarer un vecteur vide puis appeler la fonction *Recv* et la fonction *FromCFData* pour recevoir les données de type vecteur.

2.3 *Unity 3D* comme base d'une plateforme de jeu sérieux dédiée à la simulation interactive de fluide

Pour la visualisation et l'interaction, nous avons choisi d'utiliser *Unity 3D*. Contrairement à d'autres outils comme *Paraview*, *Unity 3D* permet à l'utilisateur de développer rapidement et facilement des modalités de visualisation et d'interaction personnalisées nécessaires au contrôle d'une simulation en cours. Cette plateforme dédiée au développement de jeu vidéo permet par ailleurs de facilement déployer des scénarios dans le cadre d'expérimentation ou de jeu sérieux. Cette flexibilité est de plus en plus plébiscitée en matière de visualisation scientifique, avec d'autres expériences réussies dans le domaine de la biologie moléculaire avec *BioBlender*[5] puis *Unity-Mol*[92].

2.3.1 Sélection et visualisation des résultats de simulation

Une des fonctionnalités essentielles de la visualisation de données de simulation interactive de fluide est de donner à l'utilisateur la possibilité de choisir les données à transmettre et à visualiser. Cela permet de limiter la quantité de données, souvent massives à transmettre et en fonction de ce qui doit être présenté à l'utilisateur en fonction de ses besoins. Les données

choisies et la modalité pour les afficher dépendent par ailleurs grandement du phénomène observé, de l'objectif de la simulation, de la nature des données manipulées. Une fois la sélection des données brutes effectuées, il a donc s'agit de développer les rendus essentiels, dans *Unity 3D*. Par exemple, pour les expérimentations simulation multiphase, c'est la technique consistant à afficher les données de volume de fluide (VOF ou *volume of fluid* en anglais) qui représentent la fraction volumique dans chaque voxel. A partir des données brutes, différentes techniques de visualisation ont donc été implémentée dans *Unity 3D* en fonction de la nature des données et des objectifs expérimentaux décrit dans le chapitre 3, 4 et 5.

Rendu de surface et du fluide à l'interface

Nous avons ainsi implémenté dans *Unity 3D* une modalité essentielle en visualisation de fluide, notamment pour mettre en valeur les surfaces. Nous avons utilisé une technique classique de type *Feature Extraction* dont l'objectif est de générer des surfaces à l'interface entre air/vide et eau par exemple comme dans la figure 2.4. Pour résumer, un plan est généré par voxel, quand la valeur VOF de ce voxel est comprise entre 0 et 1, ce qui signifie la présence d'une interface, comme entre de l'eau et du vide. Ce plan est calculé pour être continu avec les plans des cellules voisines. En cas de rendu exclusif de cette interface, un processus de filtrage des résultats uniquement à l'interface, permet de diminuer la quantité de données à échanger entre le module de simulation et de visualisation en ne ciblant que les données à l'interface.

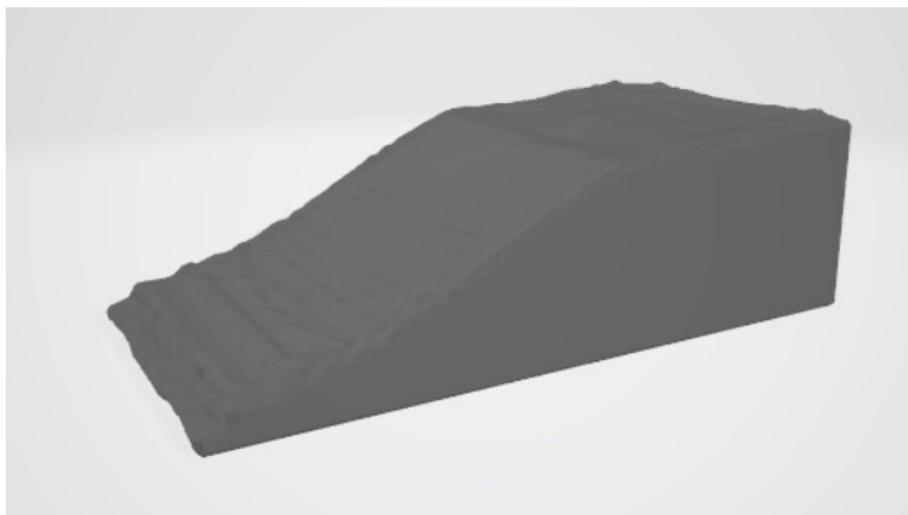


Figure 2.4: Rendu de l'interface entre l'eau et le vide par *Feature Extraction*

Rendu de valeurs projetées sur plan de coupe

En plus de rendu de type surfacique pour tout l'espace de la simulation, nous avons par ailleurs développé des fonctionnalités élémentaires et essentielles, pour afficher des plans de coupe sur les données illustrées dans la figure 2.5, sur lesquelles sont projetées les valeurs des cellules intersectant ce plan, et qui se met à jour à chaque réception de données brutes de la simulation. Cette fonctionnalité est essentielle pour observer des données scalaires, de la vitesse ou de densité, par exemple, difficilement observables avec un rendu purement volumique sur un phénomène 3D complexe.



Figure 2.5: Résultat de rendu de type VOF et d'un plan de coupe

Optimisation de rendu graphique pour gérer un très grand nombre d'objets dans *Unity 3D*

Si *Unity 3D* est très flexible, c'est grâce à son modèle orienté objet, sa gestion d'événements complexes, mais cela a un coût. Au delà d'un certain nombre d'objets, autour de plusieurs dizaines de milliers selon la performance de la machine et les complexités des objets, la performance peut diminuer fortement, et donc rendre difficile voire impossible toute interaction.

Ainsi si chaque voxel est considéré comme un objet indépendant, gestion permettant de ne pas rentrer dans des développements de rendus graphiques bas niveau qui ferait perdre son intérêt à la plateforme, ces limites sont largement dépassées même sur de petites simulations.

Des optimisations ont donc été implémentées utilisant la librairie *Entity Component System (ECS)* pour résoudre ce problème et afficher un très grand nombre d'objet dans la scène 3D. La visualisation directe des valeurs physique de chaque cellule est donc effectuée avec *ECS* en assurant un taux de rafraîchissement de 60 Hz de rendu qui permet l'interaction. Autrement, les résultats vectoriels peuvent être visualisé avec des flèches, en changeant la taille ou la couleur à la position différent.

Visualisation et interaction avec des résultats de simulation 2D

Nous avons aussi prévu dans *Unity 3D* des fonctionnalités de visualisation spécifique pour rendre et interagir avec des simulations en 2D. Toujours à partir des résultats de simulation effectués en arrière-plan par le serveur de simulation, nous avons développés des fonctionnalités de visualisation adaptés aux résultats en 2D, basés sur des textures ou des couleurs qui servent de matériaux pour présenter graphiquement la nature local du fluide 2D. Par exemple dans l'image 2.6, la couleur affiche la vitesse macroscopique des fluides. La couleur est la modalité choisie pour présenter la vitesse, le rouge correspondant à des vitesses importantes et le bleu des vitesses faibles. Les fonctionnalités d'interaction sont identiques



Figure 2.6: Exemple des résultats de vitesse

2.3.2 Modifications interactives des paramètres physiques locaux et des conditions aux limites

Dans les méthodes eulériennes, les propriétés physiques locales des fluides sont portées par les cellules du réseau, et intervenir sur ces propriétés suppose de modifier ces valeurs cellules par cellules directement depuis l'outil qui permet de visualiser les résultats du calcul en cours. Il est ainsi possible de changer les murs et les obstacles en redéfinissant la nature des cellules correspondantes à ces conditions aux limites. Ensuite, il s'agit de recalculer des valeurs physiques pour les cellules en contact avec le nouveau mur. Par ailleurs, il est possible de modifier les entrées ou les sorties de fluide, en modifiant les propriétés physiques telles que la vitesse et/ou la masse pour transformer n'importe quelle cellule en entrée ou en sortie de fluide. Cette commande ne nécessite que très peu de transfert de données entre la machine de calcul et celle dédiée à la visualisation et à l'interaction. Seules les données de cellules modifiées de manière interactive dans la scène 3D sont envoyées à la simulation. Le retour visuel de ces nouvelles conditions aux limites modifiées ou de ce nouveau contenu de cellules dans l'environnement 3D survient lorsque le client de visualisation reçoit la prochaine itération. Certaines situations particulières nécessitent le redémarrage de la simulation reprenant toutes les caractéristiques des fluides comme conditions initiales, permettant de donner à percevoir une continuité à l'utilisateur. Cette décision

est difficilement décidable automatiquement, c'est la raison pour laquelle l'utilisateur peut à tout moment revenir sur un instant de la simulation et redémarrer une simulation à partir de nouvelles conditions initiales.

En pratique dans les bibliothèques de simulation comme *Palabos*, l'état de chaque cellule est caractérisé par une étiquette, qu'il est possible de personnaliser en fonction des besoins de l'utilisateur. Nous avons utilisé cette possibilité pour définir des valeurs d'étiquette spécifiques consacrées au contrôle de la simulation en cours. Cela permet côté simulation de décider ce qu'il convient de faire pour gérer ces changements de manière physiquement appropriée : continuer la simulation sans initialiser toutes les cellules, effectuer des calculs sur les cellules voisines, redémarrer la simulation à complètement à partir des conditions initiales.

Toutes les commandes pour contrôler la simulation, telles qu'arrêter ou démarrer la simulation, sont accessibles dans l'interface *Unity 3D*, selon plusieurs modalités adaptées à chaque contexte : commande de type clavier, commande avec des boutons dédiés dans le cadre de l'interface de bureau, ou avec contrôleur 3D spécifique dans le cadre d'une utilisation en Réalité Virtuelle.

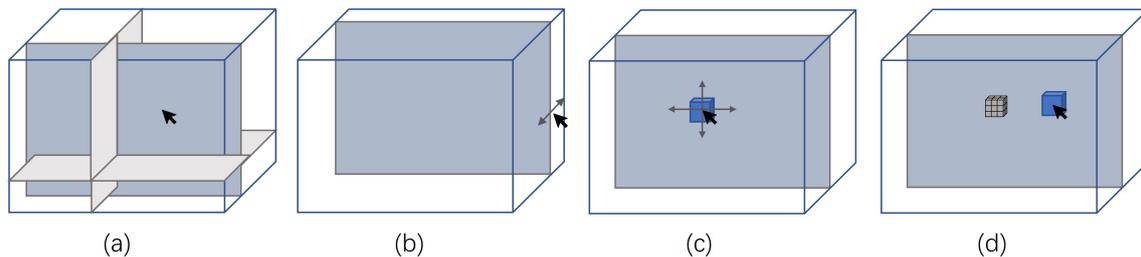


Figure 2.7: Modalité de modifications interactives du phénomène simulé : Choisir le plan (a). Déplacer le plan (b). Déplacer le curseur (c). Mettre les cellules en mur (d)

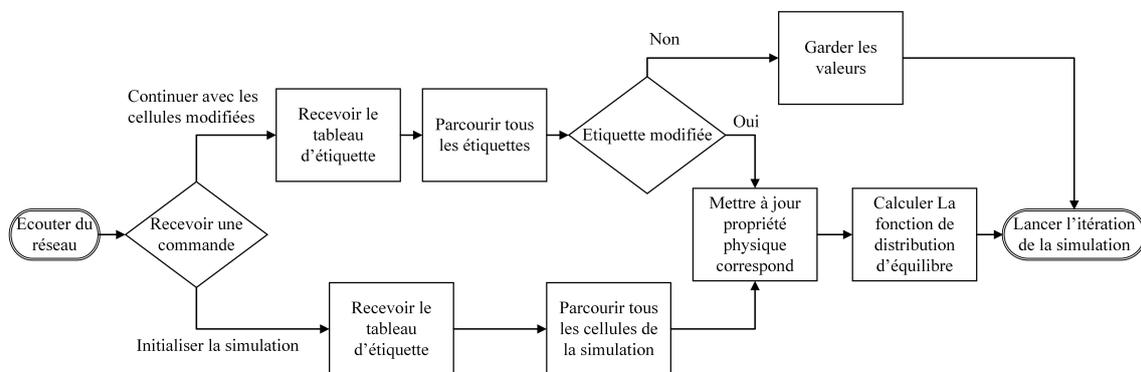


Figure 2.8: Procédure d'édition de paramètres physiques locaux et de modification de conditions aux limites côté simulation

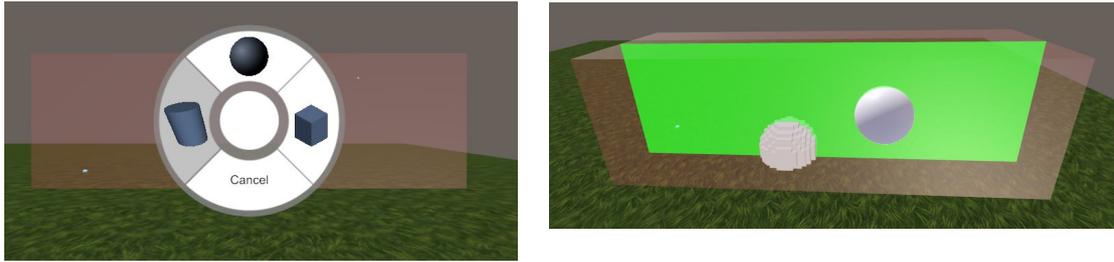


Figure 2.9: Interface de la modification de la scène : modification du mur avec le curseur sphérique

2.3.3 Définition interactive des conditions initiales du phénomène simulé

Au delà des modifications interactives pendant une simulation en cours, une des fonctionnalités ciblées dans tous les outils de simulation interactive est de faciliter la création de conditions initiales et des conditions aux limites de manière interactives depuis l'outil de visualisation et d'interaction.

Le cas d'usage qui a motivé le développement de ce type de fonctionnalité est l'importance de simplifier le processus de modélisation du phénomène simulé, en particulier pour faciliter la mise en place de scénario dans le cadre d'un jeu sérieux.

L'édition de ces conditions initiales s'effectuent selon les mêmes modalités que l'édition interactive pendant une simulation en cours.

Une fois ces conditions initiales définies côté *Unity 3D*, ces conditions initiales et aux limites sont propagées vers le serveur de simulation et sont envoyées à la simulation qui utilisera ces conditions comme point de départ. La figure 2.10 illustre la propagation des conditions initiales définies interactivement dans *Unity 3D*.

Dans le même objectif de faciliter la conception du scénario de jeu sérieux, cette fois-ci sur des fluides 2D, il est aussi possible de charger des conditions initiales prédéfinies, en utilisant une image en noir et blanc.

La partie blanche signifie le mur, où la condition aux limites de *Bounce-back* est appliqué. La partie noire signifie des fluides. Avec cette configuration de couleur, on peut simplement modifier les couleurs pour modifier les conditions aux limites.

2.3.4 Modalités d'interactions dans un environnement de bureau

Les interactions classiques avec clavier et souris pour réaliser les fonctions de base afin de manipuler un objet 3D et de naviguer dans une scène ou sont inspirés de logiciels de Conception Assistée par Ordinateur (CAO) avec une interface de type WIMP. Ainsi des techniques classiques de type "Rotate-Pan-Zoom" sont implémentées pour être réalisées avec la souris si le contexte

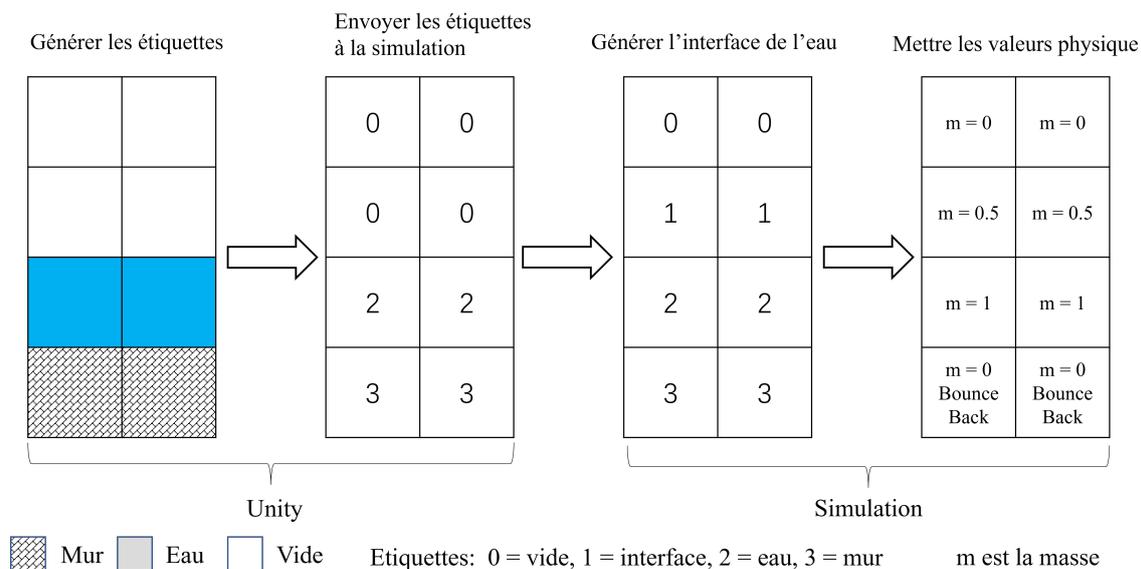


Figure 2.10: Propagation des conditions initiales de *Unity 3D* jusqu'à *Palabos*

	Nature de cellule	Valeur d'étiquette
Étiquettes initiales	Vide	0
	Interface	1
	Eau	2
	Mur	3
Étiquettes interactives	Vide	10
	Interface	11
	Eau	12
	Mur	13

Figure 2.11: Tableau d'étiquettes des cellules



Figure 2.12: Exemple de condition aux limites sous format d'image

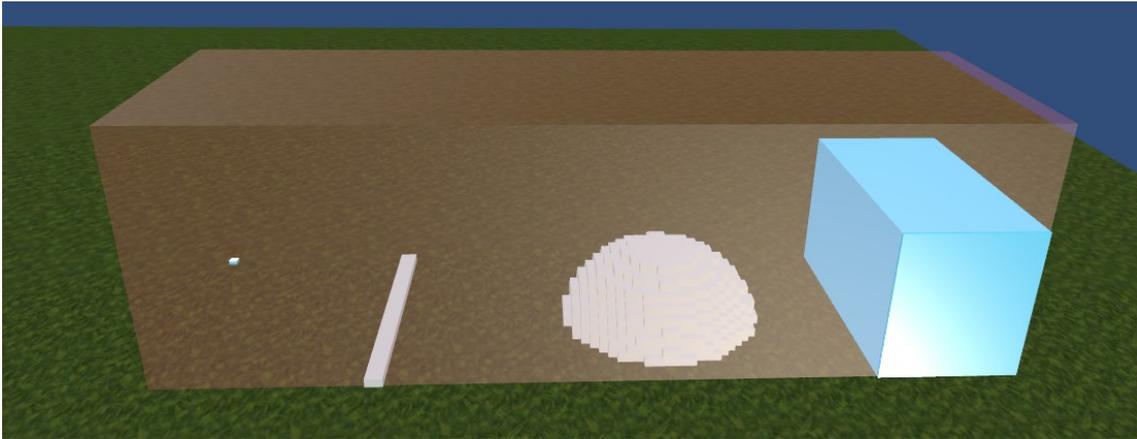


Figure 2.13: Conception de la modèle de simulation interactive. Le modèle est envoyé automatiquement à la simulation

ciblé est celui de l'environnement classique du bureau.

La modification des conditions initiales (mur, obstacle, eau, vide) et aux limites du phénomènes simulés, fonctionnalités plus spécifiques, s'effectue par plan de coupe comme illustré dans la figure 2.7, car même avec des structures de type grille rectilinéaire, la sélection directe et spécifique, non ambiguë de voxels 3D en particulier au centre du phénomène s'est avéré très difficile pour les utilisateurs.

L'utilisateur doit donc commencer par choisir un plan de coupe, puis un curseur parcourant les voxels du fluide sur ce plan de coupe lui permet de choisir les conditions initiales pour chaque cellule ou par groupe de cellules. Un système de forme prédéfinie basée sur des formes de base (sphère, cube, rectangle...) lui permet de définir l'état de cellules de manière groupée.

2.3.5 Adaptation de la plateforme de simulation interactive de fluide à un contexte immersif

Les équipements immersifs permettent de fournir des interactions plus directes que le contexte de bureau, et une meilleure immersion lors de l'observation d'un phénomène simulé et son environnement.

Au delà du déplacement classique très proches des conditions écologiques, des fonctionnalités classiques de navigation telle que la technique de téléportation a été implémentée.

De même que pour la navigation, des interactions classiques de manipulation globale de la scène 3D ont été implémentée pour utiliser le contrôleur comme le pivot pour manipuler le phénomène simulé dans la scène virtuelle.

Enfin c'est le contrôleur 3D qui est utilisé en lieu et place de la souris, pour manipuler le

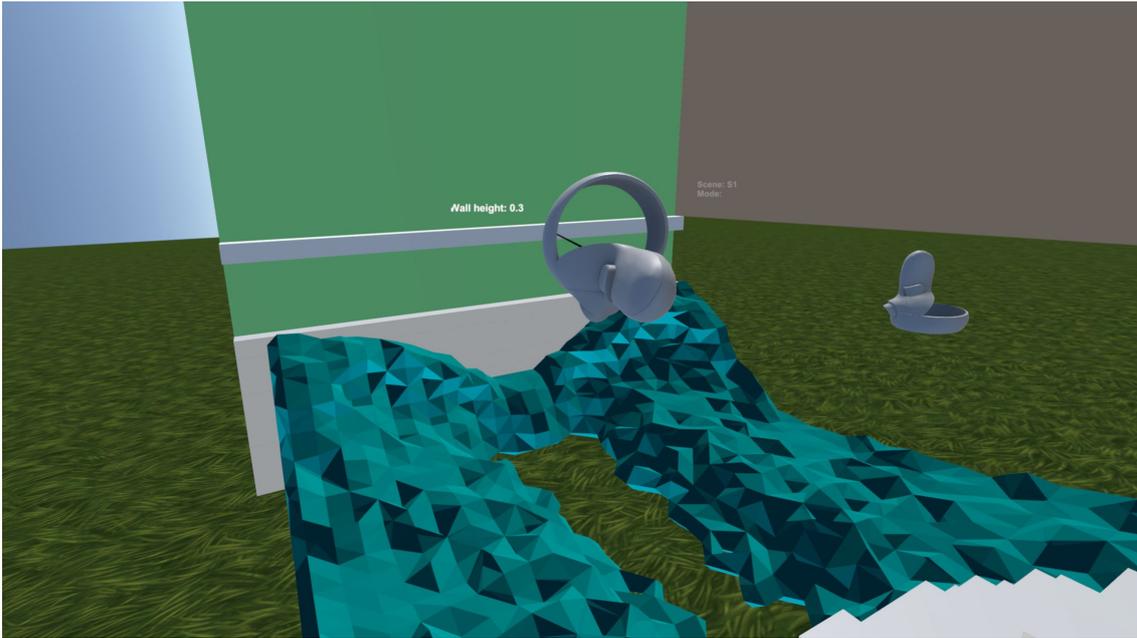


Figure 2.14: Interface du casque de Réalité Virtuelle : Modification du mur au cours de la simulation

curseur par raycasting, avec le même paradigme de gestion des modifications de condition aux limites par plan, pour garder une cohérence d'interaction avec le contexte de bureau.

Une de nos hypothèses concernant l'interaction et qui sera abordée dans les chapitre suivant, est qu'une interaction plus directe, et une meilleure immersion, permet d'améliorer la prise de décision dans le cadre de simulation interactive.

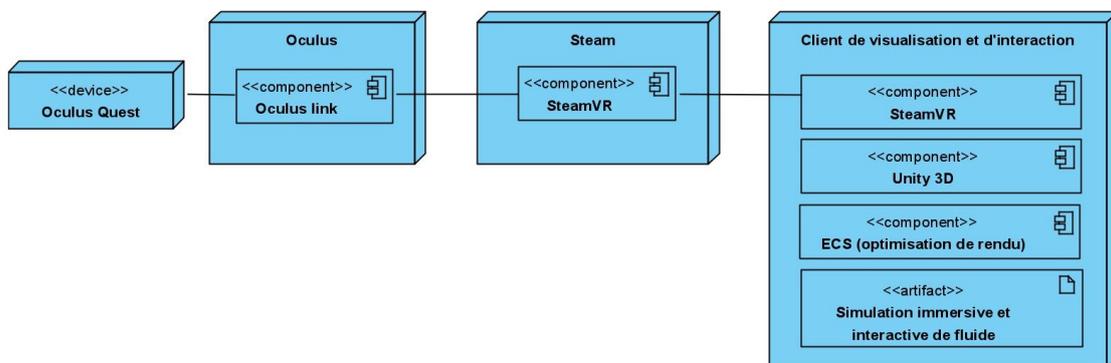


Figure 2.15: Intégration de fonctionnalités de Réalité Virtuelle dans l'infrastructure logicielle et matérielle

2.4 Modèles de calcul et simulation interactive de fluide

Dans une simulation interactive, quel que soit le domaine scientifique, une question centrale est celle de la perturbation provoquée par l'utilisateur lors du contrôle interactif d'une simulation en cours. La question la plus difficile, est celle de savoir à quelle point une modification interactive de propriété locale ou de condition aux limites, perturbe la simulation au point de la rendre non pertinente d'un point de vue physique.

La réponse à cette question est complexe, car elle dépend beaucoup du modèle de calcul utilisé (modèle eulérien ou lagrangien), plus ou moins sensible d'un point de vue théorique ou numérique à de telles perturbations. Elle dépend aussi du phénomène simulé, qui, s'il est stable et convergent, permet de considérer que le système finit toujours par retrouver un point d'équilibre malgré les perturbations interactives.

Nous avons choisi d'expérimenter notre approche sur un modèle de calcul compatible avec l'interaction du fait d'un bon compromis en termes de précision et performance, et d'une sensibilité moindre au perturbation interactive, qui est la méthode Boltzmann sur réseau ou *LBM* pour *Lattice Boltzmann Method*. Nous avons aussi tenté d'appliquer de nouvelle méthodologie de calcul, comme les schémas de calcul asynchrones pour mesurer leur intérêt pour les approches par simulations interactives, étude détaillée dans le chapitre 5.

Nous présentons donc ici de manière plus détaillé les deux modèles qui ont été utilisés dans le cadre de la mise en place de la plateforme de simulation interactive, le premier la méthode Boltzmann sur réseau, et le second une approche plus générique de calcul basé sur des schémas d'itérations asynchrones, méthodologie applicable de manière générique aux différents modèles de simulation de fluide, eulérienne ou lagrangienne.

2.4.1 La méthode Boltzmann sur réseau : un bon compromis pour les simulations interactives

Depuis sa création, la LBM a fait ses preuves pour adresser des questions et des problématiques différentes de celles de la dynamique des fluides par résolution des méthodes numériques classiques basées sur la résolution des équations de Navier-Stokes, notamment en termes de simulation de fluide multiphase, systèmes de fluides à plusieurs composants...

Un des premiers résultats de la LBM appliquée dans un cadre de système multiphasique a été présenté par Gustensen et Rothman[56]. La méthode de Gustensen, appelée aussi «Color-Method», représente des composants fluides distincts qui évoluent chacun séparément *via* leur

propre équation de Boltzmann discrétisée. Shan et Chen[133] ont introduit ensuite une autre méthode dite du "pseudo-potentiel" qui a donné une expression simple reliant les forces d'interaction entre des phases fluides disparates et l'équation d'état de chaque fluide. Swift et. al[144] ont par ailleurs développé un algorithme basé sur LBM capable de simuler des fluides hétérogène et de modéliser les forces d'interaction par un potentiel d'énergie libre. Un algorithme développé par He et Chen[65] utilisent également un paramètre d'ordre, ainsi qu'un potentiel d'interaction similaire à celui utilisé dans la méthode de Shan et Chen. Tous ces algorithmes pour les écoulements multiphasiques, à notre connaissance, sont des dérivés de la LBM de base, ou inspirées les unes des autres. La principale différence entre ces formulations LBM multiphasiques est le traitement de l'interface entre les phases fluides et l'approche utilisée pour modéliser les forces d'interaction. Ce qui est cohérent entre toutes les procédures LBM est la modélisation d'un fluide à l'échelle mésoscopique par une seule fonction de distribution des particules qui évolue selon l'équation de Boltzmann sur réseau discrétisée[65].

2.4.2 La discrétisation dans la méthode LBM

Avec la méthode LBM, la quantité discrétisée est une fonction de distribution des particules qui représente la quantité de particule dans tous les directions dans un cellule. Cette fonction de distribution des particules est liée à la vitesse, la densité, dans l'espace et dans temps. La vitesse moléculaire est une variable indépendante dans la formulation de la fonction de distribution des particules, c'est à dire qu'on doit effectuer le calcul dans chaque direction de vitesse. La méthodologie, assez triviale, pour discrétiser l'espace des vitesses est d'approcher les valeurs intégrées par pas de temps pour la densité et la vitesse. La quantité discrétisée principale, est la fonction de distribution des particules. Les caractéristiques macroscopiques d'un fluide telles que la densité et la vitesse sont liées à la densité et la vitesse de la fonction de distribution.

La LBM est basée sur des cellules de type carré en 2D ou cubique en 3D. La propriété physique sur ces cellules est l'intégrale définie par une fonction f sur une plage continue de vitesses, qui peut être approchée par une somme pondérée de cette fonction avec un nombre fini de ces vitesses. L'expression générale pour la somme des vitesses est :

$$\int f(x)dx = \sum_{i=1}^n w_i f_i(x_i) \quad (2.2)$$

Où w_i est la proportion de particules adoptant chaque direction de vitesse, direction qui est discrétisée.

Le concept de « réseau » est utilisé pour modéliser de manière discrète le champ de vitesse

à utiliser pour évaluer la fonction de distribution des particules pour chaque cellule.

La densité macroscopique est calculée à l'aide de l'équation suivante :

$$\rho(\vec{x}, t) = \sum f_i(\vec{x}, t) \quad (2.3)$$

Où $f_i(\vec{x}, t)$ est la fonction de distribution des particules dans la direction de vitesse i .

La vitesse macroscopique est quand à elle calculée grâce à l'équation suivante :

$$\vec{u}(\vec{x}, t) = \frac{1}{\rho} \sum f_i \vec{e}_i \quad (2.4)$$

Où \vec{e}_i est le vecteur de direction de la vitesse i .

L'équation LBM est [156][140] :

$$f_i(\vec{x} + \vec{c}_i \Delta t, t + \Delta t) = f_i(\vec{x}; t) + \frac{f_i^{eq}(\vec{x}; t) - f_i(\vec{x}; t)}{\tau} \quad (2.5)$$

Dans cette équation, f_i représente la distribution des particules à la position \vec{x} au moment t avec la vitesse \vec{c}_i . $f_i^{eq}(\vec{x}; t)$ est la fonction de distribution d'équilibre. Pour une simulation monophasé, il suffit d'utiliser le modèle *BGK* (Bhatnagar–Gross–Krook) [18] :

$$f_i^{eq}(\vec{x}, t) = w_i \rho + \rho s_i (\vec{u}(\vec{x}, t)) \quad (2.6)$$

et τ est le temps de relaxation vers l'équilibre. La durée nécessaire au retour à l'équilibre après la disparition d'une perturbation est appelée temps de relaxation. τ est lié à la viscosité :

$$\nu = \frac{2\tau - 1}{6} \frac{(\Delta x)^2}{\Delta t} \quad (2.7)$$

Si on réécrit l'équation 2.5 comme suit :

$$f_i(\vec{x} + \vec{c}_i \Delta t, t + \Delta t) - f_i(\vec{x}; t) = \frac{f_i^{eq}(\vec{x}; t) - f_i(\vec{x}; t)}{\tau} \quad (2.8)$$

alors la partie gauche représente la phase de propagation, et la partie droite représente la phase de collision.

La partie gauche prend la forme d'une dérivée convective. Si la fonction de la distribution des particules est la distribution d'équilibre que f est égal à f^{eq} , 2.8 devient l'équation d'advection.

Supposons que nous travaillions avec la fonction de la distribution des particules advectés avec une certaine vitesse moléculaire. Le réseau de Boltzmann devient une procédure pour

advecter la distribution des particules le long de vecteurs discrets à chaque pas de temps. Le processus d'advection est appelé « propagation ». La propagation est le processus par lequel chaque partie de la fonction de la distribution des particules est transmise aux cellules voisines.

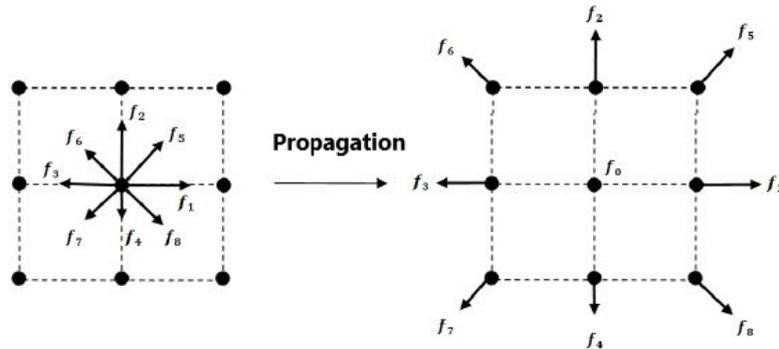


Figure 2.16: Processus de propagation d'un cellule de réseau [16].

Une fois la propagation terminée, un nouvel ensemble de composants de la fonction de la distribution des particules est calculé pour chaque réseau correspondant à partir de ceux provenant des cellules voisines. Si la fonction de la distribution des particules était en équilibre, le processus de diffusion en continu serait effectué indéfiniment sans modification de la distribution des particules à n'importe quel emplacement spatial pour tous les pas de temps. Dans ce cas, chaque partie de la fonction de la distribution des particules diffusées à partir d'un réseau donné serait remplacée par une partie identique diffusée sur le même réseau.

En supposant que la fonction de distribution des particules n'est pas en équilibre ou qu'il y a une force susceptible d'influer sur cette distribution, alors la partie droite de la fonction 2.8 n'est pas nulle. La fonction de la distribution des particules ne peut pas être advectée simplement à chaque pas de temps. Le processus de modification de la fonction de la distribution des particules à la suite de collisions ou d'une force est appelée « collision ». Le processus de collision modifie les composants la fonction de distribution des particules à chaque voxel indépendamment des autres voxels. Le processus de collision dépend souvent du calcul des forces ou d'autres propriétés macroscopiques du fluide comme la densité et la vitesse, il y a généralement une étape entre la propagation et la collision où des quantités intermédiaires sont calculées. Une fois la distribution d'équilibre locale et les forces sont calculées, l'étape de collision est effectuée, produisant un nouvel ensemble de donnée de la fonction de la distribution des particules dans toutes les directions de vitesse. Les données résultantes peuvent ensuite être transmises aux réseaux voisins par une nouvelle étape propagation.

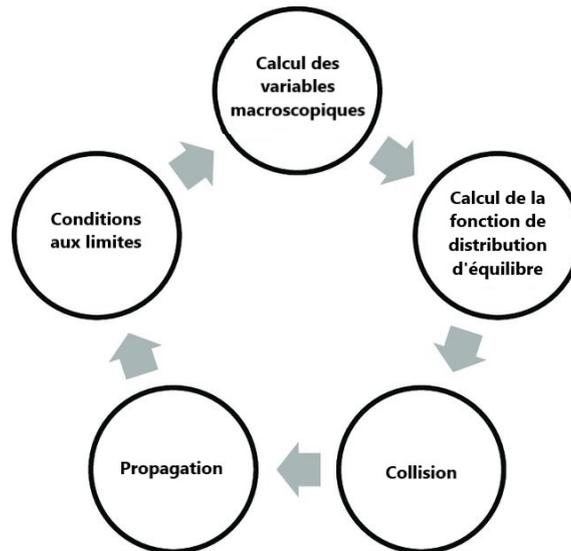


Figure 2.17: Décomposition du processus de l'algorithme sous-jacent à la LBM [143]

2.4.3 Conditions aux limites de *Bounce-back*

Les conditions aux limites de *Bounce-back* sont utilisées pour gérer le fait qu'une particule de fluide atteint la bordure, la particule se rediffusant alors dans le fluide en fonction de sa direction entrante dans ce cellule. Les conditions aux limites de *Bounce-back* existent en plusieurs variantes, par exemple le *Bounce-back on-grid* et le *Bounce-back mid-grid* [140].

Comme dans la figure 2.18, si on prend par exemple la composante qui est dans la direction f_6 , le *Bounce-back* prend l'amplitude de cette composante, et remplace la composante à la direction f_8 , par cette valeur. La même procédure est effectuée pour chacun des composants de la fonction de la distribution des particules. Une fois le *Bounce-back* appliqué, une deuxième phase de propagation est effectuée, uniquement pour les cellules au niveau des conditions aux limites.

2.4.4 Exemples de modèle

La méthode Boltzmann sur réseau classique est donc basée sur la grille régulière. Cette grille est générée par un ensemble de vitesses discrètes dans l'espace 2D ou 3D (c_i , $i = 0, 1\dots$). Elle est aussi basée sur une forme discrète de l'équation de Boltzmann, avec une collision à temps de relaxation (τ) simple (monophasé) ou multiple (multiphasé) et une solution de l'équation d'équilibre.

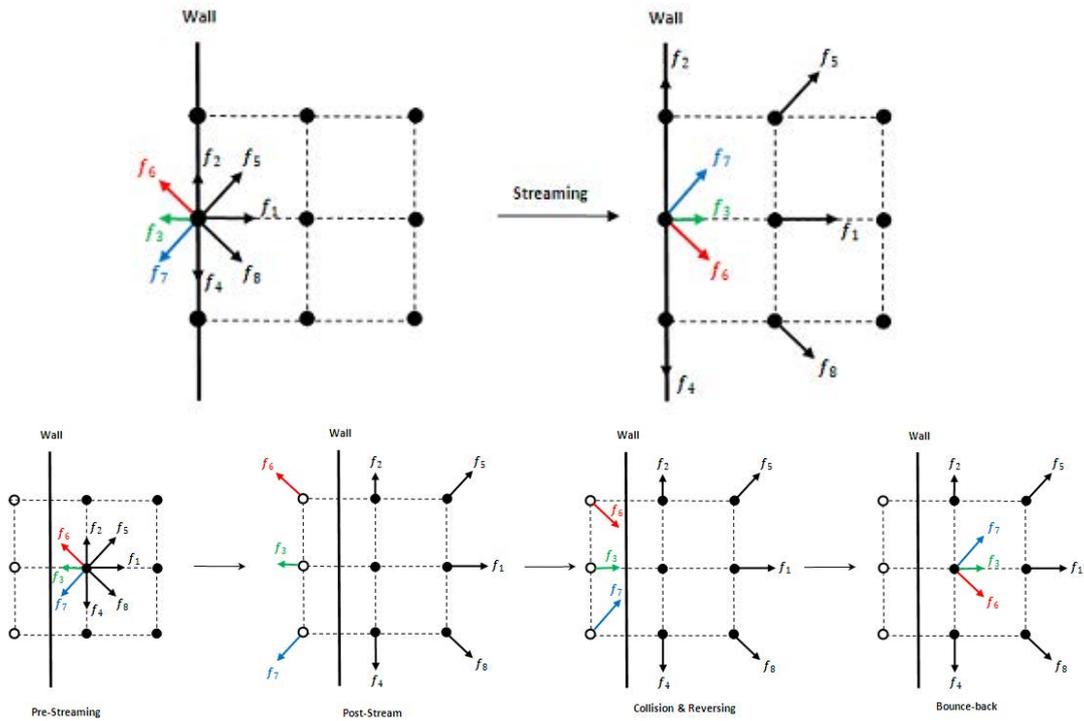


Figure 2.18: Illustration de la procédure de *Bounce-back* : *Bounce-back on-grid* (en haut) et le *Bounce-back mid-grid*(en bas) [16]

Le modèle 2D à 9 vitesses : D2Q9

Un exemple du modèle possible de discrétisation de l'espace en 2D dans la LBM est le modèle *D2Q9*. La chiffre 2 représente la dimension spatiale 2D et 9 représente le nombre de directions de vitesse. Avec le réseau *D2Q9*, il suffit donc de connaître la valeur de la fonction de la distribution des particules pour 9 valeurs de vitesse moléculaire pour évaluer une intégrale du moment. Les vitesses discrètes sont étiquetées e_{0-8} , où e_0 est le vecteur zéro et e_{1-8} pointent vers les cellules voisins. Ces vitesses peuvent être écrites comme suit :

$$e = \left\{ \left[\begin{array}{c} 0 \\ 0 \end{array} \right], \left[\begin{array}{c} 1 \\ 0 \end{array} \right], \left[\begin{array}{c} -1 \\ 0 \end{array} \right], \left[\begin{array}{c} 0 \\ 1 \end{array} \right], \left[\begin{array}{c} 0 \\ -1 \end{array} \right], \left[\begin{array}{c} 1 \\ 1 \end{array} \right], \left[\begin{array}{c} -1 \\ 1 \end{array} \right], \left[\begin{array}{c} -1 \\ -1 \end{array} \right], \left[\begin{array}{c} 1 \\ -1 \end{array} \right] \right\} \quad (2.9)$$

2.4.5 Les modèles de calcul asynchrones

Le calcul parallèle est utilisé depuis le début de l'informatique dans tous les domaines de simulation numérique. L'architecture des machines était initialement très parallèle avec des processeurs peu puissants mais nombreux, architecture qui a peu à peu été remplacée par des architectures

avec de processeurs très puissants et très centralisés, peu propice à des méthodes parallèles à grain fin. La mise en place de gros clusters de calcul, associées plus récemment à des architectures massivement parallèles à grain fin, notamment avec les GPU (Graphical Processor Unit) à l'image des premières architectures de calcul, ont fait réémergé de très nombreuses approches parallèles associées à son cortège d'outil de développement logiciel utilisé pour le calcul (MPI, OpenMP, OpenCL, CUDA...).

La parallélisation d'un code numérique peut conduire une augmentation importante de la performance, avec la réserve que des problèmes sont faciles à paralléliser, et d'autres beaucoup plus complexes, et donc avec des gains de performances très variables.

La principale problématique est l'impact important sur la performance globale de la nécessaire centralisation (synchronisation, réduction...) entre les unités de calcul effectuant les tâches de manière parallèle. Ainsi un seul calcul manquant peut faire attendre tous les autres calculs pour commencer l'itération suivante. Au delà des méthodologies parallèles, qui supposent à un moment donné une synchronisation de calcul effectué en parallèle, souvent bloquante, puisque c'est à partir de ces résultats de simulation agrégé que le pas de temps suivant peut être calculé, ont été proposées des méthodes asynchrones qui tentent d'alléger cette contrainte. Un des premiers travaux sur les itérations asynchrones a été publié en 1969 par Chazan et Miranker [32] pour résoudre des équations linéaires. De nombreux autres modèles pour le calcul parallèle et asynchrone ont été proposés [31] et la recherche dans le domaine reste très active [98][86] pour proposer des modèles de plus en plus performant, efficaces et robustes.

Le schéma de calcul asynchrone sont des schémas de calcul qui autorisent un chevauchement complet des phases de communication et de calcul lors d'itérations parallèles. Ce type d'itération ne nécessite pas que les processeurs attendent de recevoir à chaque pas de temps toutes les données nécessaires à leur calcul provenant des autres processeurs. De cette manière, les processeurs peuvent maintenir le calcul des itérations à leur propre rythme avec toutes les données reçues de manières asynchrones. Les schémas de calcul asynchrone ne sont pas aussi utilisés que les schémas de calcul synchrones pour des raisons d'équilibrage de charge, pour des raisons d'incomplétude du résultat à un instant t , mais aussi parce que les propriétés de convergence et de correction sont plus difficiles à garantir théoriquement ou numériquement. Il y a cependant de plus en plus de recherches sur les itérations asynchrones avec deux intérêts principaux pour le calcul haute performance (HPC). Premièrement, la limite supérieure standard du facteur d'accélération prévisible [3] ne s'applique pas. Deuxièmement, une plus faible sensibilité à la charge de travail déséquilibrée et aux défaillances temporaires des ressources. La Fig. 2.19 montrent comment les itérations asynchrones peuvent réduire les délais de communication pour

le calcul. Une bonne application de ces modèles asynchrones est le travail de Bahi et coll. [14] qui ont réalisé une étude sur les itérations asynchrones pour résoudre des équations linéaires et non linéaires.

Le développement dans ce domaine est très dynamique comme le montre le travail de [12][47] ou plus récemment de [97] pour calculer une erreur résiduelle sous itérations asynchrones avec une seule opération de réduction. Afin d'effectuer une simulation asynchrone, ces derniers auteurs proposent par ailleurs un framework [96] [95] pour permettre une implémentation facile de tout algorithme itératif asynchrone.

Ces performances ont été évaluées sur une problématique de type *Jacobi* asynchrone [27] et montrent que dans certains cas, elles nécessitent moins d'itérations avec des itérations asynchrones. Dans ces cas, l'explication avancée est le nombre moins important d'oscillations par rapport à l'approche synchrone. L'itération asynchrone permet également d'aboutir à un équilibrage de charge progressif [161] pour réduire le déséquilibre de charge sans réduire la stabilité ou le taux de convergence.

D'autres travaux comme *IHadoop* [49] est une adaptation de *MapReduce* à un schéma de calcul par des itérations asynchrones. Il s'agit d'un cadre de traitement à grande échelle à forte quantité de données capable de planifier des itérations de manière asynchrone, permettant d'économiser de la bande passante à la fois en termes d'entrées-sorties et des cycles de processeur. L'utilisation d'itérations asynchrones peut réduire ces facteurs jusqu'à 81% par rapport à la méthode synchrone.

Le framework *ARock* [115] un framework de mise à jour des coordonnées utilisant des itérations asynchrones. Ce framework peut mettre à jour les coordonnées partagées entre plusieurs agents (machines ou processeurs) de manière asynchrone. Dans cette recherche, l'*ARock* est comparé à une méthodologie de calcul parallèle synchrone dans lequel le nombre de tâches est également réparti sur chaque cœur, et les calculs souffrent d'un déséquilibre de charge. Le calcul à 32 cœurs a très peu d'accélération par rapport au calcul à un seul cœur, tandis que la méthodologie sous-jacente à *ARock* conserve encore 75 à 80 % d'efficacité.

Le framework *JACE* [11] est une bibliothèque multithread visant l'exécution d'algorithmes itératifs asynchrones basés sur le middleware *RMI* (pour *Java Remote Method Invocation*). Il a été amélioré avec une extension centralisée tolérante à la volatilité appelée *JaceV* [15]. Ils ont également utilisé des réseaux *Pair-à-pair* pour résoudre des problèmes scientifiques à grande échelle [13].

CRAC [38] est une bibliothèque conçue pour créer les applications asynchrones pour une architecture de grille.

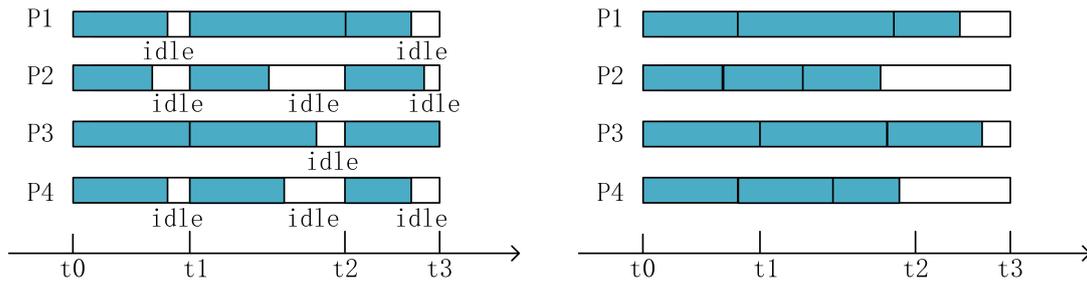


Figure 2.19: Modèles de calcul synchrones (gauche) et asynchrones (droite) : Les processeurs attendent tous les processeurs de finir l'itération actuelle pour continuer en mode synchrone. Les processeurs continuent l'itération suivante sans attendre en mode asynchrone.

Enfin, *Jack* est la première approche compatible avec une communication basée sur *MPI* pour les itérations synchrones et asynchrones. Elle fournit une interface de programmation d'application (API) similaire aux routines *MPI*.

L'adaptation des codes à des schémas d'itérations asynchrones nécessite bien plus qu'une simple mise à jour des boucles d'itérations classiques, en fonction du middleware de communication utilisé. Bertsekas et Tsitsiklis [17] ont notamment soulevé la question de l'évaluation précise d'un critère d'arrêt basé sur les résidus. Pour les itérations asynchrones, le goulot d'étranglement se reporte en effet sur les critères d'arrêt pour la fin de la simulation, comme montré dans l'exemple de la décomposition du domaine dans les méthodes appliquées aux prix des actions [164] et pour un cas dépendant du temps [165]. La plupart de cette méthodologie adresse la problématique de convergence asynchrone *via* diverses heuristiques basées sur la convergence locale de chaque processus, par conséquent, elles ne garantissent pas un état de convergence globale.

Les méthodes d'itération asynchrone sont donc très prometteuses pour dépasser les limites de performances inhérentes aux schémas de calcul synchrones, en particulier dans le cadre des architectures parallèle à grain fin. Il reste à résoudre de nombreuses problématiques liées notamment aux critères d'arrêt et à la convergence, associée à une nécessaire simplification technique et logicielle pour rendre plus accessible cette méthodologie pour une meilleure diffusion dans le domaine de la simulation numérique.

2.5 Conclusion

Dans ce chapitre nous avons présenté les travaux effectués pour mettre en place une plateforme de simulation interactive de fluide. Cette plateforme est distribuée, une machine étant dédiée au calcul de la simulation et une autre machine pour la visualisation et l'interaction. La commu-

nication entre deux machines est réalisée grâce à un protocole réseau, assurant les échanges bidirectionnels afin de pouvoir visualiser et contrôler une simulation en cours en temps interactif, et spécialement conçu pour les échanges de données de mécanique des fluides. Une bibliothèque logicielle a été conçue et utilisée pour permettre à tout développeur de rendre compatibles des outils de calcul ciblés avec notre plateforme de simulation interactive. Cette approche par couplage fort a l'avantage de pouvoir substituer n'importe quel outil de calcul à un autre, permettant par ailleurs de respecter les usages en permettant aux utilisateurs de garder, moyennant quelques modifications, le code et les méthodologies auxquelles ils sont habitués.

Nous faisons aussi état dans ce chapitre des fonctionnalités essentielles de visualisation et d'interaction développées dans *Unity 3D*. Ces contributions techniques ont pour objet de fournir un environnement de développement en matière de simulation interactive de fluide, flexible et modulaire, pour faciliter notamment le développement de matière de jeux sérieux classiques ou immersifs, appliqués à la mécanique des fluides. Cette plateforme est par ailleurs nécessaire à nos expérimentations présentées dans les chapitres suivants.

Enfin pour nos expérimentations, nous avons choisi d'expérimenter deux méthodes adossées à notre plateforme par simulation interactive de fluide. La première approche de la méthode de Boltzmann sur réseau pour la simulation de fluide, à travers l'outil *Palabos*, est donc décrite dans ce chapitre. La seconde approche propose de confronter les méthodologies de calcul asynchrone décrite aussi dans ce chapitre, avec les approches par simulation numériques interactives. Ces deux approches nous semblent en effet être deux voies très prometteuses, l'une offrant un bon compromis entre performance et précision, l'autre permettant d'améliorer encore les performances en matière de calcul parallèle, nous permettant de mettre en place les scénarios étudiés dans les expérimentations décrites dans les chapitres suivants.

Etude comparative de l'usage de simulation interactive de fluide de type Boltzmann sur réseau pour la prise de décision

3

Résumé

Ce chapitre présente une expérimentation effectuée afin d'évaluer les apports des approches par simulation interactive par rapport aux approches de simulation classique, notamment de l'impact sur la performance sur la prise de décision et sur l'expérience utilisateur. Les résultats ont été publiés via la conférence *Eurographics'2021* [154].

3.1 Introduction

L'utilisation de la dynamique des fluides numérique (CFD) nécessite différent type d'expertise et d'outils pour concevoir et exécuter une simulation, puis pour visualiser, explorer et analyser les résultats de la simulation. Différentes applications sont utilisées de manière séquentielle, notamment des logiciels de modélisation 3D, de simulation, de visualisation et d'analyse. Ce processus prend du temps car il implique de nombreuses tâches techniques et improductives telles que le traitement et l'adaptation des données à des fins d'analyse. De plus, pour obtenir des résultats physiques pertinents et précis, les simulations de fluides 3D sur des phénomènes réalistes nécessitent des ressources de calcul importantes. Il est par exemple crucial de détecter précocement les erreurs sur le modèle de simulation ou les paramètres utilisés avant la fin du calcul. Il est dans ce cas nécessaire d'extraire et d'analyser régulièrement des instantanés de simulation. Pour adresser cette problématique, nous proposons une approche par simulation interactive, notamment pour réduire le temps perdu au cours du processus séquentiel d'analyse des résultats de simulation de mécanique des fluides.

Nous faisons donc état dans ce chapitre des résultats d'expérimentation de l'usage d'environnement de simulation interactive qui permet les utilisateurs de concevoir, contrôler la simulation et visualiser le résultat en temps interactif. La fonctionnalité la plus importante dans les approches de simulation interactive est de modifier les conditions ou les paramètres en cours de la simulation sans redémarrer la simulation depuis le début.

De manière plus détaillées, ce chapitre présente les résultats d'une expérimentation qui compare l'approche par simulation de fluide classique avec une approche par simulation interactive de fluide. L'objectif de cette expérimentation est de mesurer la valeur ajoutée de la simulation interactive en termes d'efficacité et d'expérience utilisateur sur une application d'aide à la prise de décision pour éviter les débordements d'un fluide.

3.2 Objectifs et attendus de l'expérimentation

En matière de simulation interactive de fluide, peu de travaux s'attache à mesurer l'apport de simulation interactive en matière d'efficacité ou d'expérience utilisateur pour réaliser une tâche. Nous avons donc conçu cette expérimentation pour comparer les approches par simulation classique et par simulation interactive. La différence entre ces deux modes de simulation est que la simulation interactive permet aux utilisateurs de modifier les conditions sans relancer systématiquement la simulation, comme c'est le cas dans un contexte de simulation non-interactive.

Cette expérimentation écarte la tâche de modélisation du phénomène, commun à la modalité par simulation classique et par simulation interactive, tâche de modélisation nécessitant l'usage de logiciel de CAO, de logiciel de simulation pour préparer les conditions initiales du phénomène, et de logiciel de visualisation pour visualiser cet état initial. La comparaison porte uniquement sur l'apport de fonctionnalités interactives pendant une simulation en cours, inexistante dans le cadre d'une simulation classique dans lequel chaque modification conduit à un nécessaire redémarrage depuis le début.

Dans ce contexte, notre objectif est de mesurer la valeur ajoutée de la simulation interactive en termes d'expérience utilisateur et de performance pour la prise de décision dans un scénario de conception de barrage d'eau simple.

3.2.1 Durée et taille de la simulation ciblée

La taille du maillage nécessaire à la simulation a été définie comme une grille de $96 \times 30 \times 30$ cellules, soit 86 400 cellules. Cette valeur a été définie après plusieurs tests permettant de minimiser le nombre de cellules tout en conservant un niveau de précision compatible avec la tâche. En augmentant la résolution, la simulation rend difficile l'interaction, et la perception continue de l'utilisateur de l'eau s'écoulant dans la scène jusqu'au barrage. En revanche, si on diminue la résolution, la grille est trop grossière et le résultat n'est pas assez précis pour les besoins de la tâche.

Dans une simulation en 3D, l'augmentation de la résolution a un impact très important sur la performance. Après quelques tests préliminaires, nous estimons par ailleurs que pour une scène de cette taille, il prendra 3-5 minutes pour l'utilisateur de proposer une solution optimale en termes de taille du barrage, solution qui consiste à trouver la hauteur du barrage minimale pour éviter l'inondation. Ce choix est important pour aboutir à un temps d'expérimentation ne dépassant pas une heure. Avec 6 scènes d'expérimentation, auxquelles s'ajoutent 2 scènes de familiarisation, la durée de l'expérimentation est théoriquement bornée à 30 minutes. Une augmentation de la résolution par 2, avec 691 200 cellules au total, aurait conduit à une durée d'expérimentation 8 fois plus longue.

On estime donc que l'expérimentation hors familiarisation prendra 20 minutes pour une scène pour la simulation non-interactive.

3.2.2 Description des conditions initiales

Notre système a nécessité la conception interactive du modèle 3D. La plateforme a donc été utilisé par l'expérimentateur d'effectuer la conception des conditions initiales de la simulation de fluide dans *Unity 3D* ainsi que la mise en place du scénario, permettant une exécution transparente et automatique de la simulation avec les conditions initiales par l'utilisateur définies par l'expérimentateur. Le communication des conditions initiales à la machine de simulation exécutant *Palabos* utilise l'API *CFDriver* décrite dans le chapitre précédent.

Dans cette scène, la quantité d'eau est prédéfini avant l'expérimentation, différente dans chaque scène, pour éviter l'effet d'apprentissage entre les différentes sessions.

De même, l'emplacement du mur manipulable interactivement avec le clavier et la souris est prédéfini à une position toujours perpendiculaire à l'axe X, la direction d'écoulement du fluide.

Enfin, différents obstacles comme des sphères, des cubes, ont été positionnés entre le mur et la source du fluide, grâce aux fonctionnalité interactives pour définir les conditions initiales et aux limites décrites dans le chapitre 2. Le but de mettre ces obstacles est d'apporter des variations dans le comportement de l'écoulement de l'eau, permettant de varier la difficulté, rendant plus difficile la prévision de l'écoulement tout évitant l'effet d'apprentissage entre chaque essai.

Comme fonctionnalité essentielle d'une plateforme de simulation interactive de fluide, notre expérience intègre évidemment la visualisation interactive des résultats du phénomènes 3D pendant la simulation dans *Unity 3D*. Il permet par ailleurs à l'utilisateur de contrôler la simulation dans l'interface graphique *Unity 3D*, en particulier les obstacles et le barrage. Ces édition des conditions aux limites sont envoyées à la machines de simulation de la même manière que les conditions initiales, machines qui écoute les commande de l'utilisateur grâce l'API *CFDriver*, puis qui renvoi les résultats automatiquement à *Unity 3D* en prenant en compte ces modifications.

Pendant les tests préliminaires, il a été observé que la visualisation intégrale des données comporte trop d'information pour les utilisateurs et nécessite un temps d'apprentissage trop important des techniques d'interaction et tri des données durant la phase de familiarisation. Nous avons donc choisi de ne rendre que l'interface eau-vide sous forme de surface. L'interface est un maillage représentant la surface du fluide dans l'espace 3D. Selon ces observations préliminaires, cette information est suffisante pour l'objectif visé et la prise de décision, pour comprendre et visualiser l'état des fluides dans un contexte de bureau classique. Pour les obstacles et le mur, la mise en place des formes simples (cube, cylindre, sphère...) à utiliser par le sujet facilite l'interaction et la compréhension de la scène 3D, tout en permettant une large gamme de choix possibles susceptible d'impacter le comportement du fluide afin de limiter le débordement.

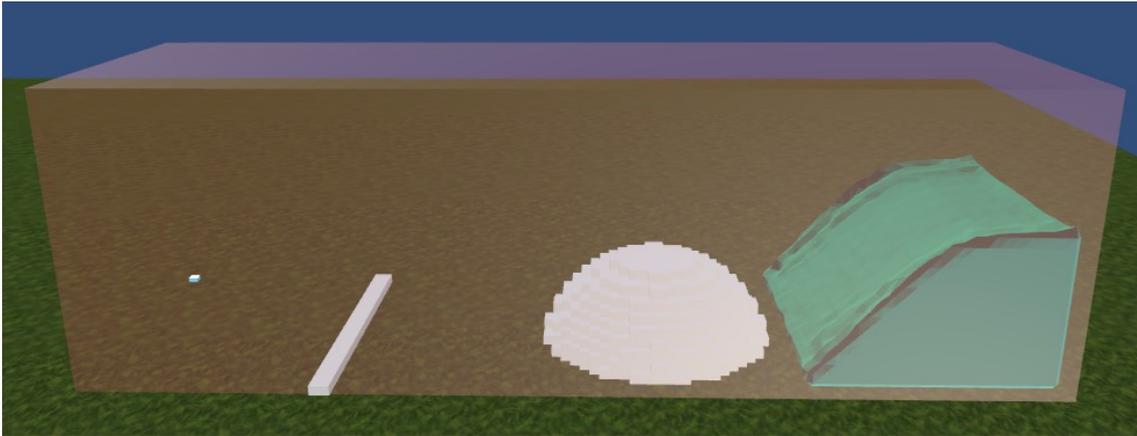


Figure 3.1: Visualisation sous forme de surface du résultat de la simulation interactive

3.3 Design expérimental

3.3.1 Description de la tâche

La tâche du participant est de trouver la hauteur minimum du barrage pour arrêter le courant de l'eau, en utilisant la simulation interactive ou classique (voir figure 3.2). Dans la scène, la position du barrage est fixée à gauche à l'opposé de la source du fluide. Cette position est fixe afin de :

- Simplifier l'interaction de l'utilisateur : trop de liberté la conception d'un modèle dans l'espace 3D rend difficile la comparaison entre la condition de simulation classique et interactive. L'utilisateur ne peut modifier que la hauteur, interaction facile à apprendre pendant la phase de familiarisation.
- Simplifier le système de détection automatique de réalisation de la tâche, comme la hauteur optimal du barrage.

Par ailleurs, différents obstacles sont placés entre e barrage et l'eau. Les obstacles ralentissent ou changent localement la direction du courant, rendant moins prévisible les possibles débordement aux niveaux du barrage, modifiant la difficulté de la tâche et permettant d'éviter l'effet d'apprentissage. Les obstacles sont donc différents en taille, en position et en forme entre les sessions. Ils ont donc un impact sur le courant d'eau et il est plus difficile pour les participants de prévoir le comportement du fluide. La quantité d'eau est aussi différente dans chaque les scènes ce qui rajoute un niveau de difficulté pour prévoir la hauteur de l'eau sur le barrage, obligeant les participants à attendre la fin de la simulation pour connaître de l'hauteur maximale de l'eau sur le barrage, et donc décider de la hauteur optimale du barrage.

Ces conditions variables et multiples, mais prédéterminées, ont permis, avant l'expérimentation,

de lancer la simulation pour toutes les possibilités afin d'obtenir dans tous les cas les résultats et la solution de chaque scène en matière de hauteur optimale du barrage. Par rapport au résultat et solutions obtenues par l'utilisateur, il y a quatre possibilités par rapport à la solution optimale précalculée :

- Le barrage est trop bas : le système détecte automatiquement le passage de l'eau. Si l'eau passe au dessus du barrage, le système envoie automatiquement une commande pour arrêter la simulation et une alerte affiche sur l'écran pour informer le participant que le barrage est trop bas, ce qui nécessite de recommencer la tâche, et l'essai de l'utilisateur étant considéré comme un échec.
- Le barrage est trop haut : quand l'eau redescende devant le mur et se stabilise, le système le détecte automatiquement, en considérant que le courant a atteint sa hauteur maximale. Le système compare alors le résultat précalculé avec la hauteur du barrage proposée par l'utilisateur. Si le barrage est plus haut que le résultat, le système envoie une commande pour arrêter la simulation et une alerte s'affiche sur l'écran pour informer le participant que le barrage est trop haut. L'essai de l'utilisateur étant considéré comme un échec.
- Le résultat est ambigu, et un redémarrage demandé : quand l'eau redescende devant le mur, le système compare le résultat précalculé avec la hauteur du barrage dans la simulation en cours. Si le barrage est plus bas que le résultat dans le système, mais que l'eau ne dépasse pas le barrage, cela peut signifier par exemple que trop d'eau est dépassé avant la stabilisation du fluide, à cause de modification de la hauteur du barrage au préalable. Le système envoie une commande pour arrêter la simulation et une alerte s'affiche sur l'écran pour informer le participant que le redémarrage de la simulation est nécessaire. L'essai de l'utilisateur n'étant considéré ni comme un échec ni comme un succès.
- Le résultat optimal est obtenu, fin de la scène : quand l'eau redescende devant le mur, le système compare le résultat précalculé avec la hauteur du barrage actuel. Si le barrage est égal au résultat optimal, le système envoie une commande pour arrêter la simulation et un message s'affiche sur l'écran pour informer le participant que la tâche est terminée. Le bouton s'affiche pour charger la scène suivante ou l'expérimentation prend la fin. L'essai est considéré comme un succès.

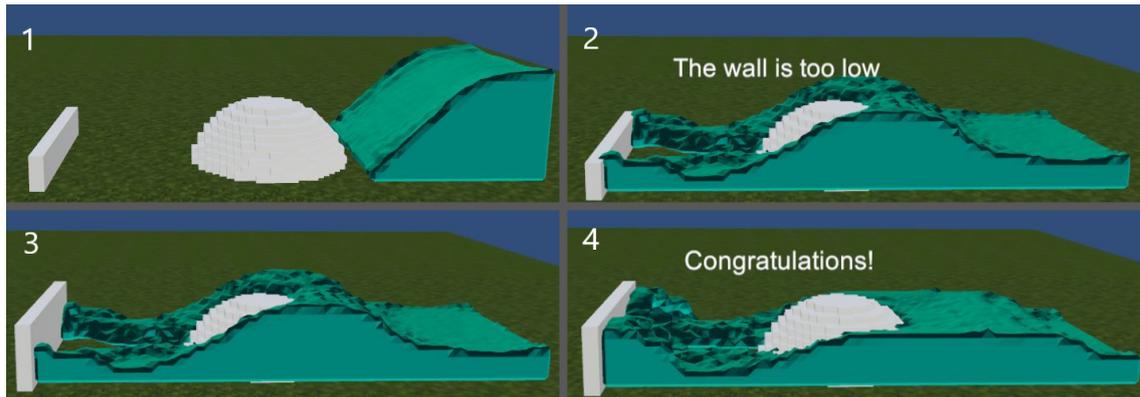


Figure 3.2: Tâche d'expérimentation de la simulation interactive : trouver empiriquement pendant une simulation en cours la hauteur optimal du barrage pour éviter l'inondation

3.3.2 Participants

24 sujets (13 hommes et 11 femmes) ont participé à notre expérimentation (N = 24). L'âge moyen est 27($\sigma = 4.83$). Les participants ne sont pas des experts dans le domaine de mécanique des fluides.

3.3.3 Données mesurées

Pendant l'expérimentation, toutes les activités de l'utilisateur ont été enregistrées. Les données brutes sont :

- Les actions de l'utilisateur : toutes les interactions comme les clics sur les buttons, les modifications du barrage, les interactions de navigation...
- Les conditions de la simulation : les conditions initiales de la simulation et les conditions après chaque modification de l'utilisateur envoyée au serveur de la simulation
- Les résultats de la simulation reçue par *Unity 3D*.

À partir de ces données brutes, les mesures suivantes ont été extraites :

- Temp d'achèvement de la tâche (TCT) : La durée nécessaire à l'utilisateur afin de terminer la tâche. La mesure commence lorsque la scène est chargée jusqu'à ce que l'utilisateur trouve la hauteur optimale du mur.
- Nombre d'échecs : le nombre d'échecs avant la découverte de la solution optimale pour chaque scène.
- Temps d'observation : le temps d'observation moyen entre chaque tentative interactive, y compris les actions de navigation.

3.3.4 Procédure

En raison de l'état sanitaire du Covid-19, une expérience en laboratoire n'était pas autorisée. Par conséquent, l'expérience s'est déroulée à distance grâce à l'outil *Teamviewer* permettant au sujet de prendre intégralement le contrôle visuel et interactif de la machine exécutant *Unity 3D* reliée à la machine consacrée à la simulation, situées au laboratoire. Pendant les expérimentations, une attention permanente sur les conditions de connexions a été effectuée pour s'assurer de la qualité et de la stabilité de l'expérience, notamment pour s'assurer d'une latence suffisamment faible pour assurer la qualité de l'interaction. Avant l'expérimentation, un e-mail contenant une notice d'information et un formulaire de consentement a été envoyé aux participants, rempli par le participant avant de commencer l'expérience. L'expérimentation est ensuite expliquée en visio pour expliquer l'expérimentation et donner des instructions. Une invitation est envoyée au participant pour se connecter à distance à mon ordinateur par *Teamviewer*. Une fois le participant est connecté, un accès à l'interface de *Unity 3D* lui est donnée pour réaliser l'expérience à distance. L'expérimentation se déroule en quatre parties principales pour chaque participant :

- la partie présentation : nous présentons à l'utilisateur la procédure de l'expérience, la nature des instructions et la durée de chaque étape au participant.
- la partie familiarisation : une scène d'essai est donnée aux participants pour se familiariser avec la plateforme expérimentale.
- la partie expérimentation : le participant doit accomplir des tâches de conception du barrage, observé par l'expérimentateur. L'étape expérimentale pour chaque participant est la même. Chaque participant teste deux conditions de simulation, interactive ou classique, avec trois scènes différentes, soit au total 6 essais.

Le scénario est le suivant :

Au début de l'expérimentation, l'environnement de l'expérimentation est choisi, puis le programme de simulation est lancé dans *Unity 3D*. La qualité de la connexion entre *Unity 3D* et la simulation est vérifiée. Le participant peut alors contrôler la simulation dans la scène *Unity 3D* dans laquelle est affiché une texture de pelouse au sol avec un pavé transparent qui présente la zone d'interaction avec la simulation.

L'expérimentation, commence, quand le participant clique sur le bouton "Next" permettant de charger la scène de l'expérimentation, incluant l'obstacle et la source d'eau. Il peut alors manipuler la caméra pour visualiser la forme et la taille de l'obstacle ainsi que la quantité

de l'eau choisie. Après cette exploration des conditions initiales, le participant est invité à modifier la hauteur du barrage qu'il estime nécessaire pour éviter un débordement. Durant cette partie, le mouvement de la caméra, c'est à dire tous les changements de point de vue par le participant, la hauteur du barrage modifiée interactivement sont enregistrées par le système.

Une fois la hauteur du barrage choisie, le participant peut lancer la simulation en cliquant sur le bouton "Start". L'eau commence donc à se déplacer et avec une mise à jour en fonction de l'arrivée des résultats de simulation. Le participant peut alors observer l'évolution du courant. Il peut toujours changer le point de vue pendant la simulation. Quand le courant monte devant le barrage, le participant peut modifier la taille du barrage pendant la simulation en cours, ou mettre explicitement la simulation en pause, à l'aide du bouton "Stop" afin de modifier le barrage en fonction de ses observations du comportement du fluide. De même, pendant cette partie, le mouvement de la caméra, le résultat de la simulation reçue par *Unity 3D*, et la modification de la hauteur du barrage est enregistré par le système.

Si l'expérimentation est en mode interactif, le participant peut continuer la simulation en cliquant sur le bouton "Continue" pour continuer la simulation avec la modification effectuée sans recommencer la simulation. La seule exception à cette situation est celle où une alerte "Please restart the simulation" s'affiche sur l'écran, qui signifie que le participant n'a pas réagi assez vite à l'évolution du courant avant que l'eau déborde le mur. L'ajout du mur est possible après avoir supprimé de l'eau dans ce cas, ce qui change le résultat final. En mode non-interactive, le participant ne peut pas continuer la simulation après les modifications. Il est obligé de la relancer avec le bouton Restart après chaque modification..

Une fois la solution correcte en termes de hauteur du barrage est trouvée, le message "Congratulations" s'affiche sur l'écran. Le participant est invité à cliquer le bouton "Next" pour charger la scène suivante. Chaque session de l'expérimentation contient 3 scènes avec ce même fonctionnement que la simulation soit interactive ou non. S'il affiche le message "Finish", la session de l'expérimentation est terminée.

- La partie questionnaire : après la session d'expérimentation, le participant est invité à remplir le questionnaire de type *NASA-TLX* [63] pour mesurer la charge cognitive et l'implication physique qui lui ont été nécessaires pour effectuer l'expérience. C'est une mesure standard pour de l'expérience utilisateur. Ce questionnaire évalue les performances sur six dimensions pour déterminer une évaluation globale de la charge de travail. Les six dimensions sont les suivantes :

- *Mental demand* : Le niveau de réflexion, de décision ou de calcul nécessaire pour effectuer la tâche.
- *Physical demand* : La quantité et l'intensité d'activité physique nécessaires pour accomplir la tâche.
- *Temporal demand* : Le niveau de pression temporelle impliquée dans l'accomplissement de la tâche.
- *Effort* : Le niveau d'effort pour maintenir son niveau de performance.
- *Performance* : Le niveau de réussite dans l'accomplissement de la tâche.
- *Frustration* : niveau d'insécurité, de découragement, de sécurité ou de contentement ressenti pendant la tâche.

L'expérimentation contient donc au total 2 sessions. La moitié des participants commence par la session avec la simulation interactive et les autres commencent par la simulation non-interactive, avec contre-balancement des ordres de passage, avec une distribution des tailles et des groupes de sujets illustrée dans la figure 3.3.

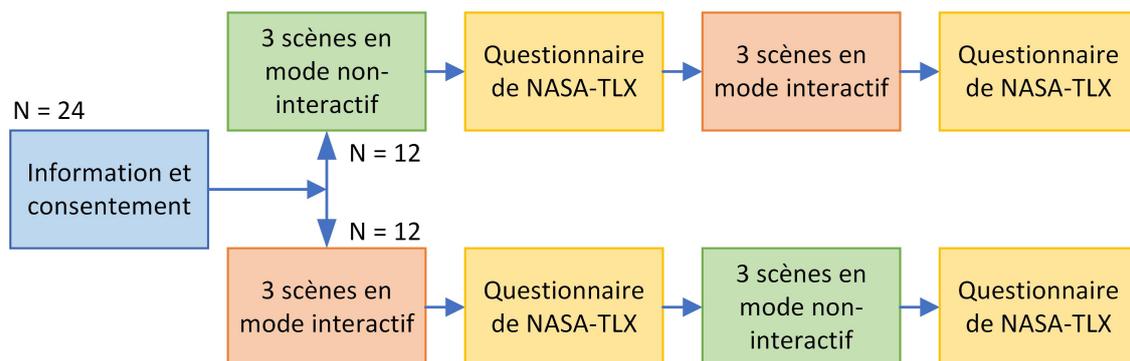


Figure 3.3: Répartition des participants : les 24 participants sont divisés en deux groupes. 12 d'entre eux commencent par une simulation interactive, puis terminent par une simulation non interactive. Pour les autres c'est l'inverse. Un design de type carré latin a été construit pour contrebalancer ces deux conditions et l'ordre choisi des 3 scènes

3.4 Hypothèses

Dans cette expérience, les hypothèses testées sont les suivantes :

- H1 : Il faut moins de temps pour trouver la hauteur correcte du barrage dans les conditions de simulation interactive.
- H2 : La simulation interactive permet aux utilisateurs de passer moins de temps à observer et analyser la scène et le résultat.

- H3 : La charge de travail globale est moindre dans le contexte de simulation interactive.
- H4: La simulation interactive permet aux utilisateurs d'explorer plus de solutions que la simulation classique.

3.5 Résultats de l'expérimentation

Les résultats présentés dans cette section ont été considérés comme statistiquement significatifs lorsque $p < 0,05$. Toutes les analyses ont été effectuées à l'aide de du logiciel *R*. Dans les graphiques à barres, les barres d'erreur représente l'erreur type (SE).

3.5.1 Résultats quantitatifs

Nous avons constaté que les données n'étaient pas toutes normalement distribuées avec un test de Shapiro. Dans la partie suivante de cette section, nous avons effectué le test Wilcoxon Signed Rank. Nous avons enregistré 144 essais : 2 conditions \times 3 scènes \times 24 participants.

Dans l'analyse de ces résultats par participants, c'est la moyenne des données de 3 scènes pour chaque participant dans chaque condition qui été utilisée. Nous exécutons donc l'analyse sur la base de 48 ensembles de données agrégées.

Pour TCT, le test Wilcoxon Signed Rank a montré que la tâche était significativement plus rapide ($p < 0,0001$) à réaliser avec la simulation interactive (moyenne = 230,4, SD = 50,0) qu'avec la simulation non interactive (moyenne = 141,625, SD = 23,4).

Pour le nombre d'échecs, le test Wilcoxon Signed Rank a montré que les participants ont fait plus d'échec ($p = 0,0003$) avec la simulation interactive (moyenne = 1,53, SD = 0,44) qu'avec la simulation non interactive (moyenne = 2,26, SD = 0,80).

Pour le temps d'observation, le test de Wilcoxon Signed Rank a montré que les participants passaient plus de temps à observer ou à naviguer dans la scène ($p = 0,002$) avec une simulation non interactive (moyenne = 8,26, SD = 3,89) qu'une simulation interactive (moyenne = 5,66, SD = 3,21).

3.5.2 Résultats subjectifs

Pour le questionnaire *NASA-TLX* [63], la moyenne des notes est considérée comme la charge de travail globale. Nous avons exécuté le test Wilcoxon Signed Rank sur cette charge de travail globale, mais nous n'avons pas trouvé de différence significative ($p = 0,346$). Par contre, une analyse plus détaillée sur l'effort nécessaire avec un test de Wilcoxon Signed Rank témoigne que

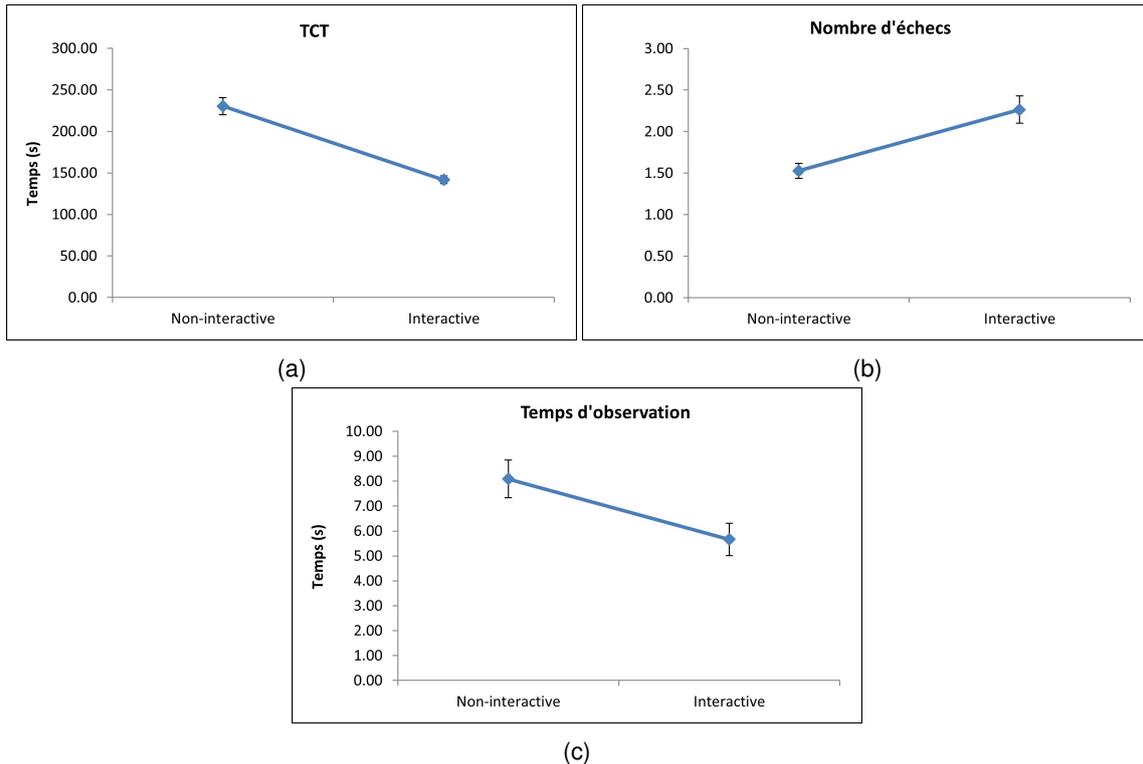


Figure 3.4: Présentation des résultats quantitatifs : moyenne de TCT (a), nombre d'échecs (b) et temps d'observation (c). Les barres d'erreur indiquent l'erreur type

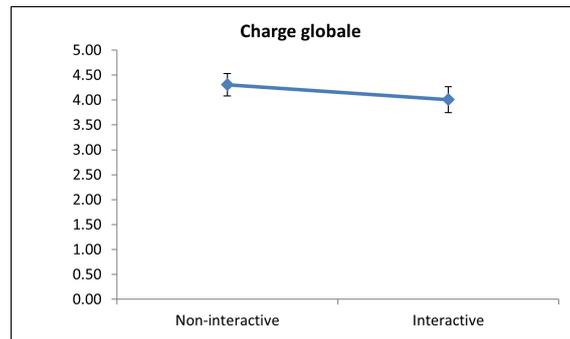
les participants font moins d'efforts ($p = 0,02$) avec la simulation interactive (moyenne = 3,71, ET = 1,83) que la simulation non interactive (moyenne = 4,38, ET = 1,93).

3.5.3 Discussion

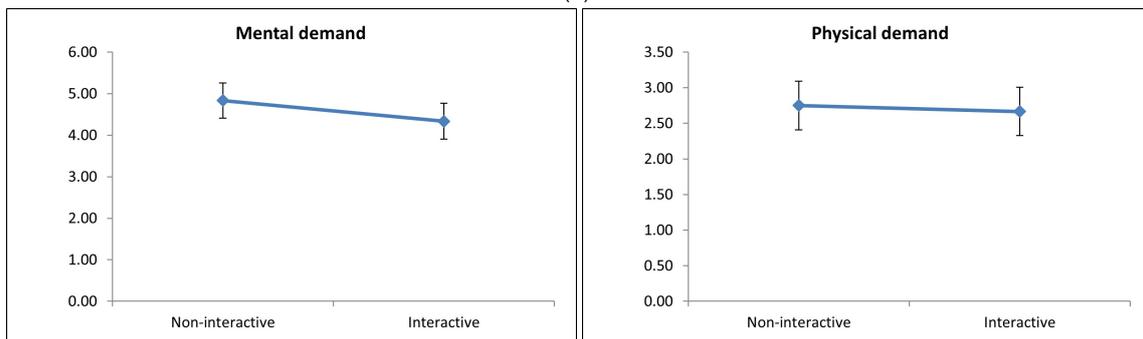
À partir des résultats, nous pouvons tout d'abord mettre en évidence que les participants ont 39% de meilleures performances de temps d'achèvement de la tâche en condition de simulation interactive (H1). La simulation interactive permet à l'utilisateur de modifier la hauteur du mur plus fréquemment sans redémarrer la simulation depuis la condition initiale.

Nous avons également constaté que les utilisateurs ont fait plus de tentatives (H4) et ont passé moins de temps à observer la scène (H2) pour accomplir la tâche dans la simulation interactive. Nous interprétons cela comme le fait que la simulation interactive nécessite moins de charge cognitive donnant la possibilité de faire plusieurs tentatives, même si les différences en termes de charge cognitive globale n'est pas significatif.

En effet, contrairement à nos attentes, nous n'avons pas trouvé de différences significatives dans la charge de travail globale, ce qui rejette H3. De la discussion avec les participants, nous avons observé que les participants n'étant pas experts en dynamique des fluides, ils avaient

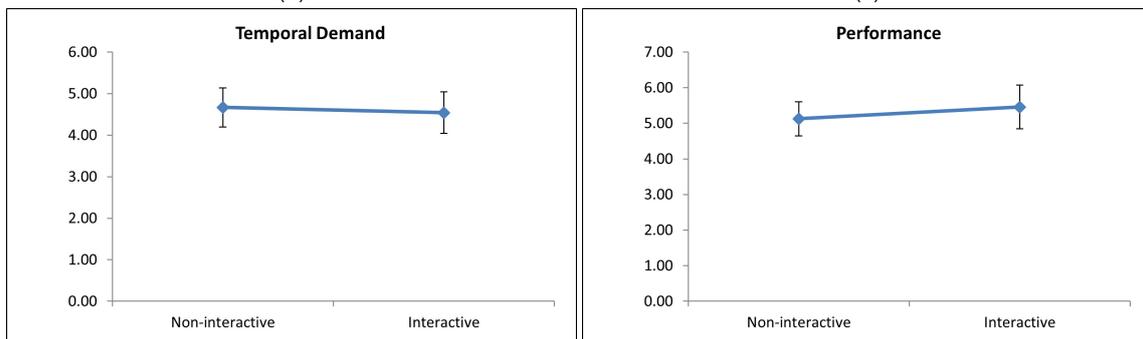


(a)



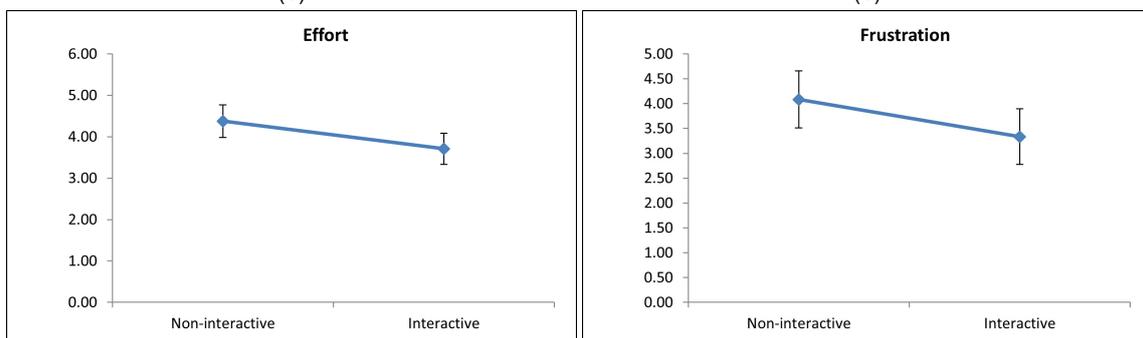
(b)

(c)



(d)

(e)



(f)

(g)

Figure 3.5: Résultats du questionnaire *NASA-TLX* : Charge globale (a), Mental Demand (b), Physical Demand (c), Temporal Demand (d), Performance (e), Effort (f) et Frustration (g). Les barres d'erreur indiquent l'erreur type

des difficultés à prévoir le mouvement de l'eau. Nous avons également supposé que le temps nécessaire pour attendre la fin de la simulation dans la condition non interactive pouvait diminuer temporairement mais fortement la charge cognitive, induisant une charge globale équivalente en contexte non interactif. Entre-temps, nous avons constaté qu'il y avait des différences significatives concernant le niveau *effort* nécessaire. Nous interprétons cela par le fait que les participants passant moins de temps à accomplir la tâche en interactif, il leur semblait que la tâche était beaucoup moins difficile de l'accomplir. Nous interprétons aussi cela par le fait, puisque les utilisateurs avaient tendance à faire plus de tentatives pour accomplir la tâche en non-interactif, ils ont eu à maintenir une charge cognitive globale moindre en interactif pour essayer de comprendre et d'anticiper le comportement global de la simulation, les manipulations interactives nécessitant moins de réflexions que la compréhension ou l'anticipation du comportement du fluide.

La stratégie des utilisateurs est par ailleurs très différente entre la condition interactive et non interactive. Il est en effet assez long d'attendre la fin de la simulation, pour en redémarrer une nouvelle avec de nouvelles conditions en mode non interactif, expliquant l'observation de stratégie d'adaptation pour minimiser le temps de complétion et le nombre d'essais. Ainsi il a été observé très fréquemment que les participants en mode non interactif laissent le fluide couler et déborder jusqu'au barrage, pour dans l'essai suivant choisir la bonne hauteur du barrage. Ce n'est pas forcément la meilleure stratégie pour minimiser le nombre d'essais, puisque le choc sur le barrage, induisant potentiellement des débordements, dépend de sa hauteur. Très rarement certain sujet a adopté une stratégie qui semble optimale en non interactif, à savoir de choisir un barrage le plus haut possible dès le premier essai, pour baisser la hauteur du barrage à son niveau optimal à l'essai suivant. Le mode interactif permet aux utilisateurs de faire plus de tentatives dans un temps limité pour atteindre l'objectif ou obtenir une meilleure solution.

3.6 Conclusion

Dans cette expérimentation, nous avons utilisé notre plate-forme interactive connectant l'outil de simulation de fluide de type LBM avec *Unity 3D* comme environnement de visualisation et d'interaction, afin de concevoir un dispositif expérimental, proche d'un jeu sérieux. Le but de ce dispositif expérimental est d'aborder de manière préliminaire la question de l'intérêt de la modification des conditions aux limites en temps interactif pendant une simulation en cours. Nous avons conçu cette expérimentation pour identifier la valeur ajoutée des approches par simulations interactives par rapport à l'approche conventionnelle dans un scénario de conception de barrage d'eau. Ces résultats préliminaires montrent que les utilisateurs ont de meilleures per-

performances dans la réalisation de la tâche avec une simulation interactive. Comme la simulation interactive est moins chronophage et demande moins d'effort, davantage de tentatives de simulation semblent remplacer un effort cognitif coûteux d'analyse du comportement simulé, ce qui induit d'après les interviews avec les utilisateurs une perception d'une difficulté moindre pour réaliser la tâche. En revanche, la simulation interactive ne semble pas réduire la charge de travail globale des utilisateurs pour accomplir la tâche dans les mesures par questionnaire standard de type *NASA-TLX*.

Apport de l'immersion pour l'aide à la décision dans un contexte de simulation de fluide interactif

4

Résumé

Ce chapitre présente une expérimentation effectuée dans le but d'évaluer l'apport de l'utilisation d'un système Réalité Virtuelle pour une tâche analogue à celle présentée dans le chapitre précédent, mais dans un contexte de simulation interactive. Dans cette expérimentation, nous avons donc comparé l'expérience utilisateur dans la réalisation d'une tâche de la simulation interactive de fluide dans un contexte classique de bureau, à celle d'un contexte immersif.

4.1 Introduction

Les phénomènes réalistes en simulation de fluide sont tridimensionnels, ce qui questionne évidemment l'usage de contexte d'interaction spécifiquement conçu pour manipuler et explorer des objets 3D comme celui de la Réalité Virtuelle. Notre plateforme a été conçue pour gérer à la fois la simulation de fluide en 2D et en 3D, dans l'objectif d'étudier justement l'apport de l'immersion et des paradigmes d'interaction 3D sous-jacents.

Les tâches de visualisation et modélisation 3D en mécanique des fluides comme dans d'autres domaines, sont particulièrement complexes à réaliser du fait d'un environnement 2D que ne prévoit que des interactions ou des visualisation basées sur des dispositifs 2D, comme l'écran ou la souris. Cette problématique est adressée par la conception et l'usage de paradigmes d'interaction et de navigation 3D particulièrement avancées, mais qui rendent ces tâches de conception, de manipulation et d'exploration 3D plus complexes. En effet, l'apprentissage de ces techniques d'interaction, de manipulation, d'édition et d'exploration de phénomènes 3D complexe, est difficile du fait d'une affordance moins forte et de modes d'interaction toujours indirectes avec des objets 3D, très loin de nos habitudes d'interaction avec nos objets 3D du quotidien. Le contexte d'interaction immersif se rapprochant du contexte de manipulation écologique de notre environnement 3D, nous avons donc réalisé cette expérience présentée dans ce chapitre, pour évaluer l'apport de la Réalité Virtuelle en comparant par rapport à un contexte classique d'ordinateur de bureau. Une des questions posées est de savoir si et dans quelles mesures le contexte immersif améliore les performances et l'expérience utilisateur dans un contexte de simulation interactive de fluide.

4.2 Adaptation de la plateforme de simulation interactive de fluide à un contexte immersif

4.2.1 Equipement immersif utilisé

Pour mener cette expérimentation, nous avons utilisé un casque Oculus Quest 2, équipé de deux écrans avec la résolution 1832×1920 par œil. Ce casque est équipé d'un processeur et d'une batterie intégré, et est directement connecté sans fil à la machine exécutant la scène *Unity 3D*, recevant et affichant les résultat de la machines de calcul.



Figure 4.1: Dispositif de type casque immersif Oculus Quest 2

4.2.2 Adaptation des techniques d'interaction 2D au contexte immersif

Les équipements de Réalité Virtuelle nous permettent d'utiliser de nombreuses techniques d'interaction avec un environnement 3D, notamment la manipulation directe des objets 3D, le déplacement écologique, i.e. en marchant dans la scène, ou des métaphores de téléportation pour naviguer dans la scène 3D. Pour évaluer l'apport de l'immersion dans une tâche de simulation interactive de fluide, il a donc fallu implémenter des interactions immersives permettant d'effectuer des opérations interactives analogues à celles offertes dans un contexte classique de bureau, présenté dans les chapitres précédents. Ainsi les techniques de navigation de type *Rotate-Pan-Zoom* et de raycasting avec clavier/souris pour sélectionner les objets 3D ont été adaptées aux dispositifs offerts par le casque de Réalité virtuelle.

Ainsi, pour se déplacer dans le monde virtuel, comme avec la souris, l'utilisateur utilise un des boutons du contrôleur associé à un mouvement de contrôleur pour déplacer la scène 3D. Nous rappelons qu'un casque de Réalité Virtuelle permet nativement à l'utilisateur de tourner la tête et d'explorer en 360° autour de lui, et qu'il peut se déplacer en marchant autour de l'objet d'intérêt dans notre cas de taille limitée, ce qui évite d'avoir à implémenter des fonctions complexes de navigation. Limiter les fonctions de navigation, notamment le contrôle de la rotation, permet de réduire le *cybersickness*, les êtres humains étant particulièrement sensibles à la manipulation de la composante rotation dans un contexte immersif [146].

Pour l'interaction avec les voxels de l'objet simulé, nécessaire à l'édition des conditions aux limites comme la hauteur du barrage, nous avons choisi de garder des modalités d'interaction les plus analogues possibles avec celles utilisées dans l'expérience précédente. Les participants choisissent d'abord un plan, puis sélectionnent la cellule à modifier parmi les cellules de ce plan. La technique de raycasting fonctionne de manière identique à ceci près que les dispositifs d'interaction permettent d'avoir directement un rayon 3D dans la scène, contrairement au contexte 2D dans lequel il faut utiliser le curseur de souris pour construire ce rayon dans la pro-

fondeur. En 2D comme en 3D, sur un plan il n'y a pas d'ambiguïté possible pour sélectionner la cellule intersectant le rayon à sélectionner.

4.3 Objectifs et attendus de l'expérimentation

Comme la technologie Réalité Virtuelle permet d'avoir une meilleure perception de phénomène 3D et fournit des modalités d'interaction directes, notre hypothèse principale est que les équipements de Réalité Virtuelle facilitent la tâche et offre de meilleures performances dans les tâches sous jacente à une simulation de fluide 3D interactive.

Des tests préliminaires ont démontré l'importance de fournir des modalités d'interaction 3D les plus élémentaires et les plus abordables possible pour éviter au maximum les biais liés au niveau de familiarisation moins important des utilisateurs avec des dispositifs d'interaction 3D.

Les fonctionnalités d'interaction 3D ont donc été choisies pour être très similaires à l'expérimentation précédente dans un contexte classique tout en limitant la difficulté de manipulation et d'apprentissage d'interaction basées sur des contrôleurs 3D immersifs.

Attentif à cette difficulté de comparaison d'interaction de nature différente, l'objectif de cette expérimentation est de mesurer l'apport d'un contexte de Réalité Virtuelle par rapport à un contexte classique pour une tâche identique dans un contexte de simulation interactive de fluide.

4.4 Design expérimental

4.4.1 Tâche

La tâche est identique à l'expérimentation précédente : Dans la scène, la position du barrage est fixée à l'avance. Différents obstacles sont placés entre le barrage et l'eau. La tâche pour les participants est de trouver la hauteur minimale du barrage pour arrêter le courant de l'eau sans débordement.

De même que dans l'exploration précédente, il a été demandé aux participants de bien observer l'évolution de l'écoulement et de proposer interactivement la solution optimale avec le moins d'échecs possibles.

4.4.2 Participants

16 participants (11 hommes et 5 femmes) ont participé notre expérimentation (N = 16). Le moyen de l'âge est 27,6($\sigma = 2,88$). Tous les participants ont rempli un questionnaire comme information

de base, qui a été utilisé pour recueillir des informations démographiques telles que leur niveau de compétences en physique, leur expérience des jeux vidéo et des dispositifs de Réalité Virtuelle, et d'autres statistiques personnelles (par exemple, l'âge, la main dominante, etc.). Tous les participants sont droitiers. Tous les participants ont utilisé leur main dominante pour effectuer la tâche. Parmi ces participants, aucun ne s'est déclaré expert dans le domaine de mécanique des fluides ou en ingénierie d'hydraulique. La moitié ont reçu une éducation en physique à l'université.

4.4.3 Données mesurées

Pendant l'expérimentation, toutes les activités de l'utilisateur ont été enregistrées. Les données brutes enregistrées sont :

- Les actions de l'utilisateur : toutes les interactions comme les clics sur les boutons, la modification de la hauteur du barrage, les phases d'observation et de navigation.
- Les conditions de la simulation : les conditions initiales de la simulation et les conditions après chaque modification de l'utilisateur envoyés au serveur de la simulation
- Les résultats de la simulation reçus par *Unity 3D*.

À partir de ces données brutes, les mesures suivantes ont été extraites :

- Le temps d'achèvement de la tâche (TCT) : la durée pendant laquelle l'utilisateur a terminé la tâche. La mesure a commencé lorsque la scène a été chargée jusqu'à ce que l'utilisateur trouve la hauteur optimale du mur.
- Le nombre d'échecs : le nombre de fois où l'utilisateur a trouvé le une hauteur du barrage non optimale.
- Le déplacement dans la scène : combien de fois l'utilisateur a déplacé la scène ou la caméra et la distance par rapport à l'objet d'intérêt.

4.4.4 Procédure

En raison de l'état sanitaire du Covid-19, le processus de nettoyage conseillé a été appliqué. Avant chaque participant, le clavier et le souris ou le casque de Réalité Virtuelle ont été nettoyés. Le participant est ensuite invité à s'asseoir devant un bureau et un formulaire de consentement qui décrit l'expérience lui est remis. Il lui est possible de poser des questions durant cette phase. Il est demandé au participant s'il est d'accord avec les modalités de l'expérimentation, il signe alors le formulaire.

Cette expérimentation se décompose en quatre parties pour chaque participant :

- La partie présentation : le déroulement de l'expérience est détaillé, avec une description des instructions et de la durée de chaque étape.
- La partie familiarisation : une scène d'essais est proposée aux participants pour se familiariser avec le dispositif expérimental.
- La partie expérimentation : Le participant effectue les tâches de conception du barrage sous l'observation de l'expérimentateur. Chaque participant effectue deux sessions avec deux scènes différentes, avec l'ordinateur de bureau et avec le casque, soit au total 4 essais.
- Partie questionnaire : Après une session de l'expérimentation, le participant est invité à remplir le questionnaire *NASA-TLX* [63].

Le scénario de travail est le même que dans l'expérimentation précédente, mais l'expérimentation s'effectue uniquement en mode interactif, avec deux conditions, l'une dans un contexte d'interaction classique et l'autre dans un contexte de Réalité Virtuelle immersive, avec un contrebalancement de l'ordre de passage de ces conditions. La répartition des participants, illustrée dans la figure 4.2 .

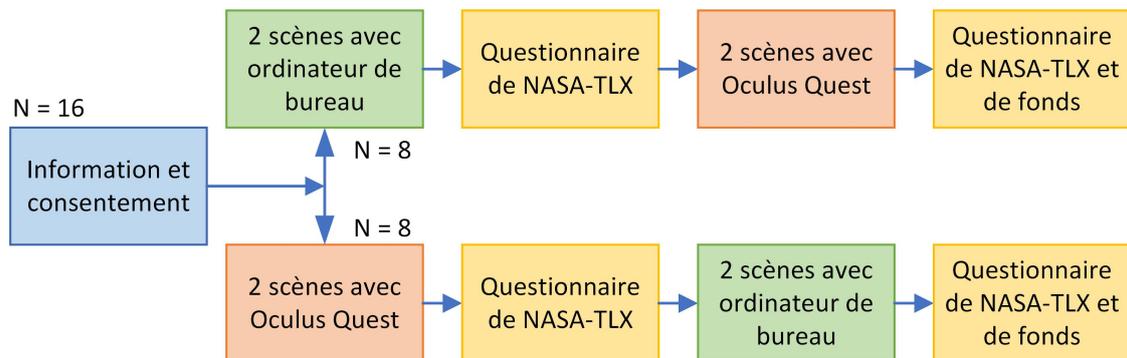


Figure 4.2: Répartition des participants : 16 participants ont été divisés en deux groupes. 8 d'entre eux ont commencé avec l'ordinateur de bureau puis se sont terminés avec le casque Oculus Quest 2. Les 8 autres ont commencé avec le casque Oculus Quest 2 puis se sont terminés avec l'ordinateur de bureau. Le design carré latin a été construit pour contrebalancer la condition et la scène.

4.5 Hypothèses

Les hypothèses testées dans l'expérience sont les suivantes :

- H1 : Le casque de Réalité Virtuelle est plus efficace pour compléter la tâche en termes de performance,

- H2 : Le casque de Réalité Virtuelle est plus efficace pour compléter la tâche en termes de nombre d'échecs,
- H3 : Le casque de Réalité Virtuelle peut réduire la charge de travail globale.

4.6 Résultats

Les résultats présentés dans cette section ont été considérés comme statistiquement significatifs lorsque $p < 0,05$. Toutes les analyses ont été effectuées à l'aide de *R*. Dans les graphiques avec des barres d'erreur, les barres d'erreur indiquent l'erreur type (SE).

4.6.1 Résultats quantitatifs

Comme dans l'expérience précédente, nous avons constaté que les données n'étaient pas toutes normalement distribuées dans un test Shapiro. Dans la partie suivante de cette section, nous avons effectué le test Wilcoxon Signed Rank. Nous avons enregistré 64 essais: 2 conditions \times 2 scènes \times 16 participants.

Nous avons calculé aussi la moyenne des données de 2 scènes pour chaque participant dans chaque condition. Nous exécutons l'analyse statistique sur la base de 16 ensembles de données agrégées.

Pour le TCT, contraire à notre prévision, test Wilcoxon Signed Rank a montré que la tâche était significativement plus lente ($p = 0,002$) à réaliser avec le casque de Réalité Virtuelle (moyenne = 153,53, SD = 32,56) qu'avec le clavier et la souris (moyenne = 122,97, SD = 18,83).

Pour le nombre d'échecs, le test de Wilcoxon Signed Rank a montré que les participant ont effectué moins d'échecs ($p = 0,03$) avec le casque de Réalité Virtuelle (moyenne = 1,06, SD = 0,98) qu'avec le clavier-souris (moyenne = 1,63, SD = 0,92).

Pour le déplacement dans la scène virtuelle, le test Wilcoxon Signed Rank a montré que les participants se sont déplacés davantage dans la scène virtuelle ($p = 0,014$) avec le casque de Réalité Virtuelle (moyenne = 25,4 SD = 21,5) qu'avec le clavier et la souris (moyenne = 11,3, SD = 10,3), et aussi plus souvent ($p = 0,003$) avec le casque de Réalité Virtuelle (moyenne = 11,4, SD = 7,2) qu'avec le clavier-souris (moyenne = 3,78, SD = 3,79).

Nous avons aussi mesuré la précision, en mesurant le nombre de clics ratés pour sélectionner une cellule, avec la souris comme avec le contrôleur 3D. Un test de Wilcoxon Signed Rank a montré que la technique raycasting est moins précise ($p < 0,0001$) avec le casque de Réalité Virtuelle (moyenne = 0,62, SD = 0,12) qu'avec le clavier-souris (moyenne = 0,86, SD = 0,10).

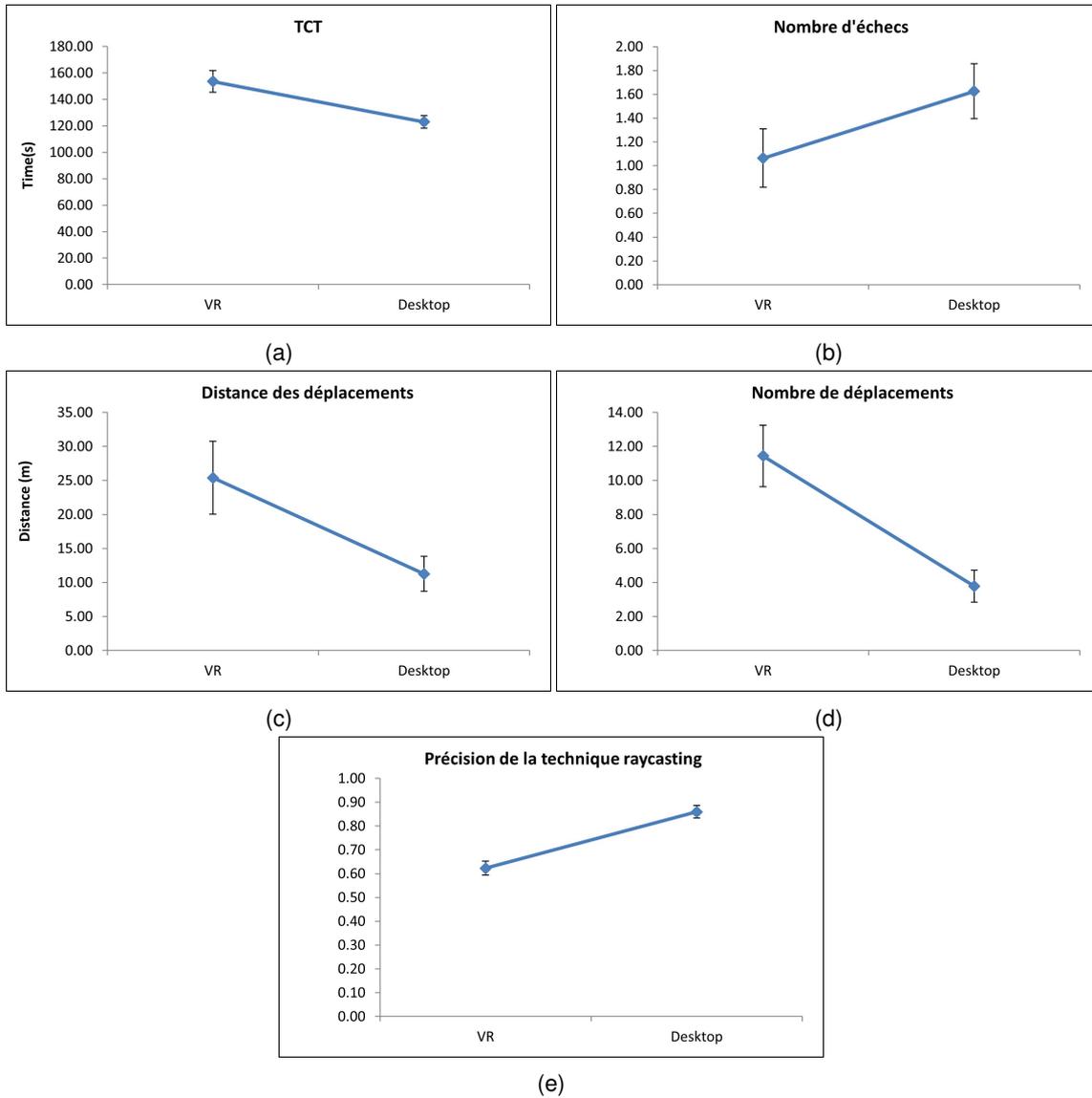


Figure 4.3: TCT (a), nombre d'échecs (b), distance des déplacements(c), nombre de déplacements (d) et précision (e). Les barres d'erreur indiquent l'erreur type.

4.6.2 Résultats subjectifs

Pour le questionnaire *NASA-TLX*, les notes sont moyennées pour calculer la charge de travail globale. Nous avons exécuté le test Wilcoxon Signed Rank pour cette charge de travail globale. La charge de travail globale est plus importante ($p = 0,005$) avec le casque de Réalité Virtuelle (moyenne = 3,79, SD = 1,62) qu'avec le clavier-souris (moyenne = 3,03, SD = 1,30).

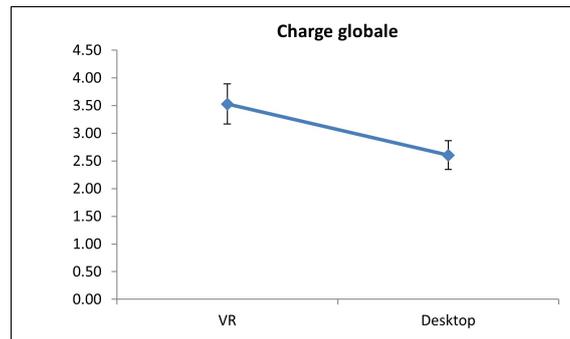
4.6.3 Discussion

Contrairement à nos attentes, l'expérience montre que le casque de Réalité Virtuelle est 25% moins performant que le clavier-souris pour réaliser la tâche, ce qui rejette H1. Pour expliquer ce résultat, nous avons observé que la manipulation est moins efficace avec le casque de Réalité Virtuelle. En effet nous avons constaté qu'avec le casque de Réalité Virtuelle, la tâche de sélection est 27,9% moins précise qu'avec le clavier-souris. Nous avons clairement sous-estimé l'impact de la différence de performance d'une tâche élémentaire, notamment de sélection entre le contexte classique et le contexte immersif, ce qui aurait pu être anticipé, et que nous aurions pu corriger avec des outils d'aide à la sélection appropriés.

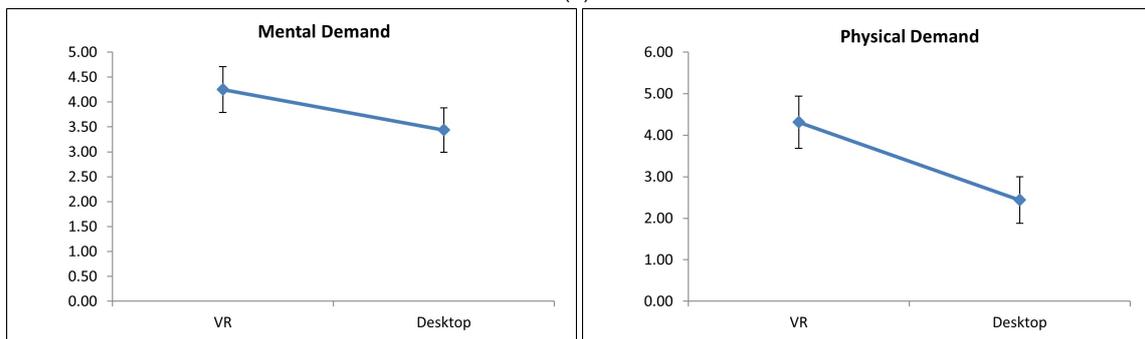
En revanche, l'utilisation de casque de Réalité Virtuelle permet de minimiser le nombre d'échecs, 1,65 des échecs sont effectués par chaque scène dans un contexte classique, contre 1,06 dans un contexte immersif, en plus des résultats significatifs sur cette mesure, validant H2.

En ce qui concerne la charge de travail, contrairement à nos attentes, nous avons mesuré que la charge de travail globale est significativement plus importante avec le casque de Réalité Virtuelle, ce qui rejette l'hypothèse H3. De la discussion avec les participants, les participants constatent subjectivement que si le casque de Réalité Virtuelle donne une meilleure qualité de visualisation et perception globale du phénomène, il s'avère de manière cohérente avec la problématique de sélection, que l'interaction était plus complexe dans le contexte immersif qu'avec le clavier et la souris. On souligne aussi dans les questionnaires *NASA-TLX* et de manière significative que les participants ont remarqué une plus grande frustration ("Frustration") en contexte immersif, mais que nous ne sommes pas en mesure de lier uniquement à la problématique de sélection.

Enfin, et ce résultat est en général attendu en contexte immersif, que l'implication physique est perçue comme plus importante par le participant, notamment du fait de la posture debout durant toute l'expérimentation, et parce que la navigation s'effectue de manière plus écologique par le déplacement physique autour de l'objet d'intérêt. Une plus forte implication physique, cohérente notamment avec la tâche ne doit pas être vue comme un résultat négatif, puisque de nombreux

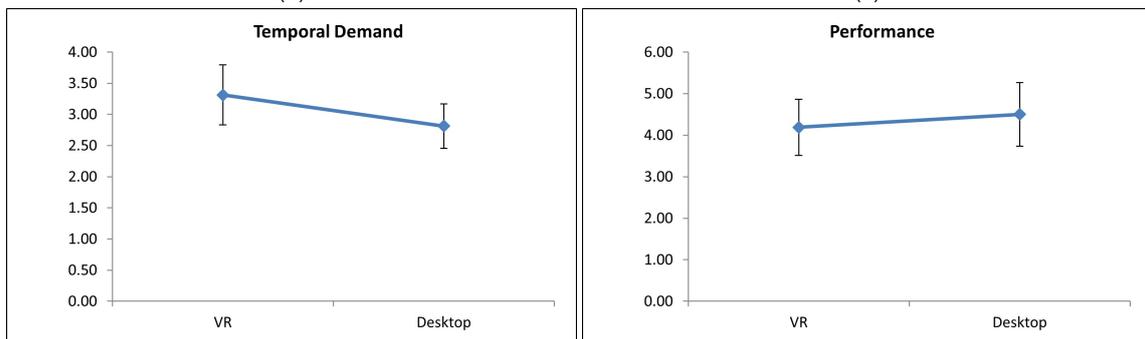


(a)



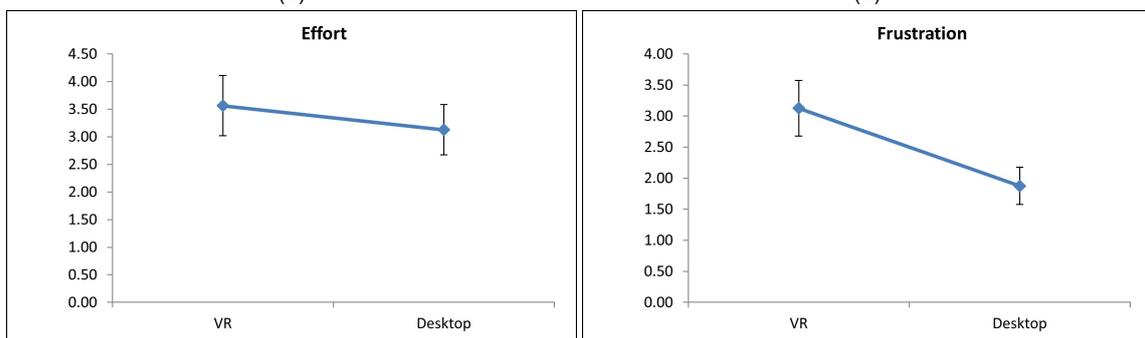
(b)

(c)



(d)

(e)



(f)

(g)

Figure 4.4: Résultats du questionnaire *NASA-TLX* : Charge globale (a), Mental Demand (b), Physical Demand(c), Temporal Demand (d), Performance (e), Effort (f) et Frustration (g). Les barres d'erreur indiquent l'erreur type

travaux de la littérature, l'implication physique est essentielle en contexte immersif pour diminuer le cybersickness, en particulier pour les tâches de navigation. Le casque de Réalité Virtuelle semble faciliter effectivement la navigation et les déplacements dans le monde virtuel, améliorant potentiellement la perception 3D du phénomène, avec déplacement autant en quantité qu'en fréquence (voir figure 4.3).

4.7 Conclusion

Dans cette expérimentation, nous avons mené une étude préliminaire sur l'impact de l'usage de contexte immersif, dans une tâche de prise de décision liée à une problématique classique en simulation de fluide. Si les participants sont moins efficaces en termes de temps, résultat que nous estimons dû à un biais expérimental lié à une plus grande difficulté de sélection dans le contexte immersif, nous avons noté que le nombre d'échecs étaient significativement moins important en Réalité Virtuelle, alors que le temps de réalisation est plus important. Associé à une charge mentale globale plus importante, une physique plus forte, et des déplacements plus importants en contexte immersif, nous en concluons que le contexte immersif améliore la qualité de l'exploration, de la prédiction et donc de la prise de décision en termes de nombre d'essais, grâce à une meilleure implication globale de l'utilisateur. Pour définitivement conclure sur un apport de l'immersif dans cette tâche, nous pensons par ailleurs qu'un travail sur la technique de sélection immersive utilisée, pour la rendre aussi performante qu'en 2D, est nécessaire pour valider l'hypothèse H1, c'est à dire une meilleure performance en termes de temps de complétion.

Etude de l'usage de modèles de calcul asynchrone dans un contexte de simulation interactive de fluide

5

Résumé

Après ces expérimentations concernant l'apport d'une approche par simulation interactive, puis du contexte immersif sur la prise de décision sur une tâche liée à la simulation de fluide, ce chapitre aborde un autre volet de notre travail, sur l'utilisation de méthodologie de calcul asynchrone dans le contexte de simulation interactive de fluide, d'abord à travers une première application de schéma asynchrone sur un phénomène simplifié de diffusion de chaleur, puis en évaluant l'impact de la nature partielle des résultats présentés à l'utilisateur, intrinsèques aux méthodologies asynchrone de calcul.

5.1 Simulation interactive et modèle de calcul asynchrone : première application et retour d'expérience

Avec le développement de la technologie du calcul parallèle, les algorithmes spécifiques sont conçus et utilisés pour optimiser l'utilisation simultanée de nombreux processeurs pour augmenter les performances de simulation, requises pour l'interactivité. Dans la conception de ces algorithmes, l'équilibrage de charge sur tous les processeurs est un aspect très important. L'équilibrage de charge signifie que le travail doit être réparti de manière équitable entre tous les processeurs. Par ailleurs, dans les schémas de calcul synchrones, si certains processeurs terminent le calcul plus rapidement que d'autres, ces processeurs doivent attendre que tous les autres processeurs aient terminé leur tâche pour passer à l'itération suivante, ce qui peut impacter très négativement les performances.

Les méthodes par itérations asynchrones adressent ce double problème d'équilibrage de charge et d'attente avant la synchronisation. Ainsi les processeurs les plus rapides n'attendent pas les autres processeurs et continuent le calcul, en garantissant malgré tout des critères de convergence qui assurent la cohérence de la simulation globale. Cela minimise un certain nombre de contraintes, comme le fait de nécessairement distribuer les travaux de manière équitable pour chaque processeur, ou de diviser le phénomène en domaines géométriques de tailles équivalentes. Cela nous permet d'être beaucoup plus flexible sur la gestion des sous-domaines ou sur la politique de maillages qui peut être plus complexe ou hétérogène quand un niveau de précision local plus élevé est requis.

Dans ce travail exploratoire, guidé par l'idée que les méthodes asynchrones pourraient être une réponse aux exigences de hautes performances imposées par les approches par simulation interactive, nous avons d'abord tenté d'appliquer cette approche sur un phénomène simple pour évaluer la faisabilité d'utiliser un modèle de calcul asynchrone avec notre plateforme de simulation interactive.

Nous rappelons que notre plateforme de simulation interactive permet à ses utilisateurs de visualiser et contrôler la simulation pendant une simulation en cours. Dans ce contexte, l'utilisation d'itérations asynchrones pour la simulation interactive présente plusieurs avantages, notamment celui de fournir un résultat local rapidement, résultat local qui peut être suffisant pour la prise de décision, notamment concernant le contrôle interactif pour répondre à une problématique d'optimisation.

Notre système de simulation interactive nécessite la communication entre l'outil de simula-

tion et l'outil de visualisation. Le temps dédié à la communication des données de simulation, à la fois entre les processeurs de calcul pendant une itération, et à la fin de chaque itération vers la machine dédiée à la visualisation, peut prendre une proportion importante, et impacter de manière importante la performance. En itérations asynchrones les communications sont susceptibles d'être fortement réduites, la communication étant effectuée à la volée, localement, sans attendre la synchronisation et l'achèvement de l'itération.

Par ailleurs, une des caractéristiques des simulations interactives est de rendre plus difficile les optimisations pour augmenter les performances de calcul, notamment car les conditions aux limites ne sont pas figées, mais modifiables de manière interactive. Ces caractéristiques empêchent de planifier une distribution de travail correctement équilibrée sur tous les processeurs pour le calcul. Les modifications interactives peuvent ainsi avoir un impact sur les performances en rendant caduque les optimisations de performance avant de lancer la simulation.

5.1.1 Travaux préliminaires d'application d'un schéma de calcul asynchrone à une simulation de diffusion thermique

Nous avons d'abord tenté d'appliquer les méthodes asynchrones sur un phénomène simple. Nous avons donc adapté un calcul synchrone avec le framework de calcul à schéma asynchrone *JACK2* [95] sur phénomène simulé est un transfert de chaleur en 2D. Il contient 200×200 cellules. Ces cellules sont divisées en 8 zones pour réaliser le calcul parallèle. Chaque processeur s'occupe de 5000 cellules. Nous avons essayé plusieurs configurations différentes pour évaluer comment et dans quelles mesures nous pourrions profiter des calculs par itérations asynchrones. Les résultats ont été publiés via la conférence *DCABES 2019* [153].

La simulation est basée sur l'équation de conduction thermique :

$$\frac{\partial T}{\partial t} = k \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) \quad (5.1)$$

La figure 5.1 illustre ce phénomène de transfert de chaleur. La chaleur est diffusée depuis le centre. Comme le but de cette simulation est de comparer le calcul synchrone et asynchrone, on a utilisé un modèle simple avec l'équation explicite. ∂t est défini comme 1 seconde pour chaque itération. La fin de la simulation est définie comme 10 000 itérations.

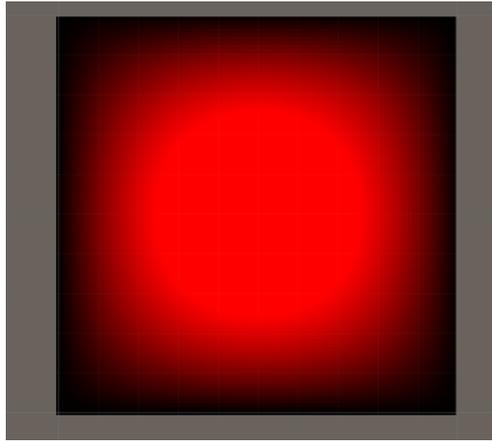


Figure 5.1: Visualisation de la simulation de transfert thermique dans une scène *Unity 3D*

5.1.2 Illustration pratique de l'intérêt d'un modèle asynchrone sur un cas limite

Le calcul parallélisé dans cette application a une très mauvaise efficacité car les processeurs s'échangent de manière systématique toutes les données, et que le temps de calcul est très réduit, avec une équation physique simulé est très simple. C'est donc la communication entre processeurs qui prend beaucoup de temps.

Par ailleurs, pour réaliser une visualisation en temps interactif de ces résultats, nous avons évidemment toujours besoin de transmettre toutes les données entre l'outil de simulation et l'outil de visualisation.

De plus, la communication entre les machines de simulation et de visualisation prendra beaucoup plus de temps que le calcul d'une itération, même si le calcul est bien balancé entre les processeurs, et le rendement de la parallélisation selon un modèle synchrone est très peu efficace.

Nous avons mis en place deux expérimentation dans lesquelles est évaluée la performance du calcul asynchrone par rapport au calcul synchrone.

La première expérimentation consiste à observer la performance avec un calcul parfaitement équilibré, ce qui signifie que la quantité de travail sur chaque processeur est presque la même, sans visualisation.

Dans la seconde expérimentation nous avons consacré, en plus du calcul, un processeur à la communication avec l'outil de visualisation *Unity 3D*.

Dans la première expérimentation, comparant un modèle synchrone et asynchrone, la figure 5.2 montre le temps nécessaire pour finaliser la simulation. On peut observer une différence

légère entre un modèle par itérations synchrones et par itérations asynchrones, même dans le cas d'un partage très équilibré favorable au modèle synchrone, on peut observer que les itérations asynchrones peuvent améliorer les performances.

Dans la seconde expérimentation, pour illustrer de manière caricaturale l'intérêt de modèle asynchrone, un processeur, P1, est consacré à la fois au calcul et à la communication des données à l'outil de visualisation. Ce processeur est volontairement très chargé par cette tâche de communication en plus de la tâche de calcul, est attendu par tous autres processeurs comme illustré dans la figure 5.4), ce qui impacte très négativement la performance de la simulation synchrone, comme illustré dans la figure 5.3. En asynchrone, les autres processeurs n'ont pas besoin d'attendre le résultat du processeur le plus lent, en évitant ce problème d'attente, ce qui induit que la performance de la simulation est très supérieure à un modèle asynchrone.

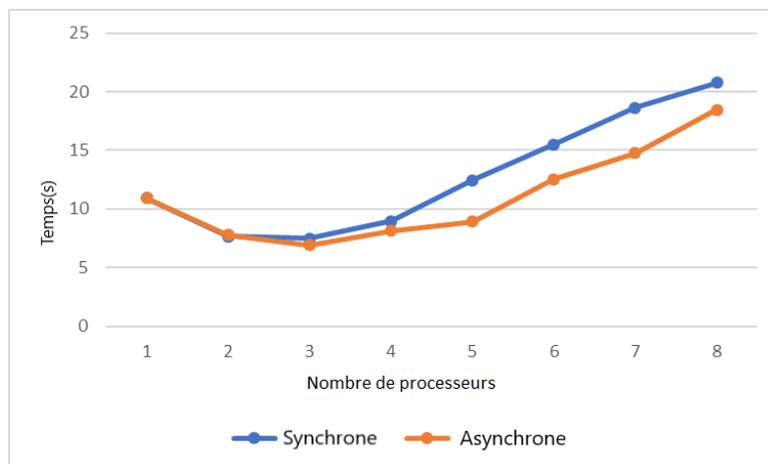


Figure 5.2: Temps de calcul d'une simulation bien distribuée avec nombre de processeurs différent

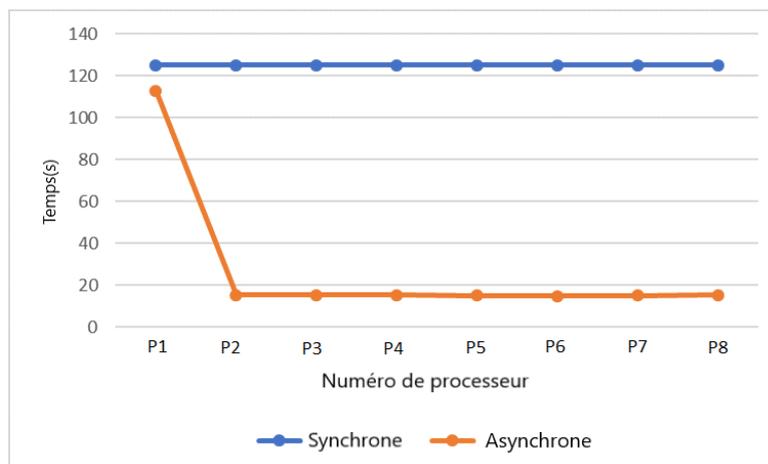


Figure 5.3: Temps de calcul sur chaque processeurs avec un processeur de calcul consacré à la communication vers l'outil de visualisation

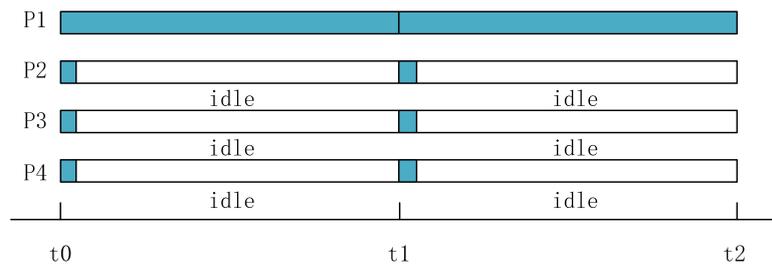


Figure 5.4: Plan de charge des processeurs avec un processeur de calcul consacré à la communication vers l'outil de visualisation

La seconde configuration a été reproduite avec une simulation plus volumineuse en 3D avec le transfert de données d'un maillage d'un poids de 12,8 Mbits. Sur un réseau local avec une bande passante de 100 Mbps et un temps de latence est de 1 milliseconde, environ 100 millisecondes ont été consacrée à la transmission des données, et avec un temps passé sur le calcul total d'environ 1 minute, il a fallu 15 minutes pour terminer la simulation.

5.1.3 Scénario réaliste de l'impact d'édérations interactives des conditions aux limites sur un parallélisme basé sur la séparation par domaines

Dans le cadre de simulation de fluide proche de notre cas d'application, c'est à dire un fluide qui doit être freiné par des obstacles et un barrage, une séparation par domaine comme schéma de calcul parallèle est classique. Dans un cadre de simulation classique, on segmente le phénomène en domaine en rectangle de taille équivalente, ou en faisant en sorte d'avoir des domaines contenant un nombre de cellules actives, c'est à dire contenant du fluide et donc nécessitant des calculs, afin d'optimiser l'équilibrage de charge. Dans le cas de simulation interactive, comme les conditions aux limites changent, certains domaines peuvent se retrouver plus chargés en cellules actives, pendant que d'autre peuvent se retrouver sans aucune cellules actives, devenant par exemple inaccessible au fluide, avec son processeur associé sans activité. Dans ce contexte, un schéma de calcul asynchrone est particulièrement intéressant pour adresser cette problématique de déséquilibre de charge dû à l'éditation des conditions aux limites.

5.1.4 Conclusion

Dans cette partie, nous avons expérimenté en pratique le couplage d'un modèle asynchrone à une simulation interactive. Dans notre étude préliminaire, nous illustrons dans des cas limitant l'intérêt de l'utilisation de schéma de calcul asynchrone. Ces résultats volontairement caricaturaux aux limites avec le pire déséquilibre de charge possible, illustrent sur des cas pratiques

comment les schémas avec itération asynchrones permettent d'adresser une problématique relativement fréquente. Dans un contexte de simulation scientifique classique, il n'est en effet pas toujours facile d'équilibrer la charge, notamment dans le cas avec des maillages ou des grilles hétérogènes si des niveaux de précisions importants sont localement nécessaires. Dans un contexte de simulation interactive, ce problème est potentiellement amplifié pour des éditions interactives des conditions aux limites.

5.2 Etude d'impact du caractère partiel des résultats de calcul asynchrone sur la prise de décision en simulation interactive de fluide

5.2.1 Introduction

Les schémas asynchrones peuvent s'avérer théoriquement efficaces en cas de déséquilibre de charge, et dans un contexte de simulation interactive dans lequel les éditions des conditions aux limites peuvent remettre en cause les optimisations mises en place à priori. Cependant, l'absence de synchronisation des données à chaque itération, le fait que les résultats peuvent être partiels et incohérents à chaque interaction peut impacter la qualité de la prise de décision. C'est dans ce contexte que se situe l'expérimentation décrite dans la suite de ce chapitre. Il s'agit de comparer la performance de l'utilisateur pour exécuter plusieurs tâches de simulation interactive de fluide, avec le calcul asynchrone et le calcul synchrone. Pour faciliter le contrôle des variables indépendantes, on a simulé le calcul asynchrone en contrôlant le caractère partiel et non synchronisé dans chaque zone de la simulation.

5.2.2 Objectifs et attendus de l'expérimentation

Avec cette expérimentation, le but est de savoir si le calcul asynchrone peut avoir un impact sur la performance de l'utilisateur pour réaliser la tâche pendant une simulation interactive de fluide. Avec le calcul asynchrone, toutes les données ne sont pas nécessairement synchronisées ou continues à chaque itération, notamment dans le cas de travaux de calcul mal distribués. Cela peut par exemple causer des fréquences de rafraîchissement différentes dans chaque zone.

On a donc cherché à mesurer l'impact sur l'utilisateur de différents niveaux d'asynchronisation, en changeant le taux de rafraîchissement des résultats dans chaque zone de manière hétérogène.

Notre hypothèse est qu'en dessous d'un certain seuil d'asynchronisme, à déterminer, l'impact est négligeable sur la tâche et l'expérience utilisateur. Si le calcul asynchrone est en dessous de ce seuil critique il peut raisonnablement être utilisé pour améliorer la performance de la simulation, avec des performances de l'utilisateur peu dégradées pour finaliser les tâches. Comme il est attendu que ce niveau dépend aussi de la tâche à réaliser, nous avons reproduit cette expérience sur des tâches différentes.

5.2.3 Un modèle de calcul de type LBM 2D comme simulation intégrée dans *Unity 3D*

Pour cette expérience, pour cause de confinement, et sans accès facile et permanent aux machines de calcul au laboratoire, nous avons dû réécrire un code de simulation 2D de type LBM dans *Unity 3D* pour que les sujets puissent directement télécharger une seule application. Adapté à la performance de la simulation intégré dans *Unity 3D*, j'ai choisi le modèle D2Q9 en 2D. Cette nouvelle contrainte nous a permis de tester la robustesse et la modularité de notre plateforme de simulation interactive.

La simulation choisie pour cette expérimentation concerne un fluide incompressible avec de l'eau, un mur et du vide. Les écoulements sont considérés laminaire, le nombre de Reynolds est environ 100.

Comme la simulation est une simulation des écoulements monophasés, il est suffisant d'utiliser uniquement la collision de type *BGK* (Bhatnagar Gross-Krook). La fonction de distribution d'équilibre est :

$$f_i^{eq}(\vec{x}, t) = \omega_i \rho + \rho s_i(\vec{u}(\vec{x}, t)) \quad (5.2)$$

où s_i est défini comme

$$s_i(\vec{u}) = \omega_i \left(\frac{\vec{e}_i \cdot \vec{u}}{c^2} + \frac{(\vec{e}_i \cdot \vec{u})^2}{2c^4} - \frac{\vec{u} \cdot \vec{u}}{c^2} \right) \quad (5.3)$$

Où le ω_i est la proportion en constante.

$$\omega_i = \begin{cases} \frac{4}{9} & i = 0 \\ \frac{1}{9} & i = 2, 4, 6, 8 \\ \frac{1}{36} & i = 1, 3, 5, 7 \end{cases} \quad (5.4)$$

c est la vitesse du réseau :

$$c = \frac{\Delta x}{\Delta t} \quad (5.5)$$

La viscosité cinématique du fluide dans le modèle D2Q9 est liée au temps de relaxation par :

$$\nu = \frac{2\tau - 1}{6} \frac{(\Delta x)^2}{\Delta t} \quad (5.6)$$

L'algorithme principal utilisé est le suivant :

- Initialiser ρ , \vec{u} , f_i et f_i^{eq}
- Appliquer l'étape propagation : propagation de f_i dans le sens de \vec{e}_i avec la partie gauche de l'équation 2.8.
- Calculer les données macroscopiques : ρ , \vec{u} avec l'équation 2.3 et 2.4.
- Calculer f_i^{eq} avec la partie gauche de l'équation 5.2.
- Appliquer l'étape collision avec la partie droite de l'équation 2.8.
- Recommencer par l'étape propagation avec la partie gauche de l'équation 2.8.

Stabilité de la simulation interactive avec la méthode LBM

Avec le modèle BGK, la méthode LBM peut induire des problèmes de stabilité, surtout quand le nombre de Reynolds est important[82]. Par exemple la fonction de distribution peut être négative, ce qui donne la densité négative. Le modèle BGK ne garantit pas que la fonction de distribution soit positive, source de l'instabilité d'un modèle de collision de type BGK[91][110].

Pour éviter le problème de stabilité, nous avons choisi un nombre de Reynolds faible et avec la viscosité importante pour construire notre système et pour faire les expérimentations d'utilisateur.

5.2.4 Design expérimental

Description de la tâche

Pour faciliter le contrôle des variables indépendantes, on a conçu un scénario de simulation de fluide en 2D pour tester le système de simulation. Comme les expérimentation du chapitre 3 et 4, la simulation est aussi basée sur LBM mais intégré dans *Unity 3D*. La simulation inclut des objets qui se déplacent avec l'eau sur place. Les objets se déplace dans le sens et avec la vitesse égal à la direction et la vitesse locale de fluide comme dans la figure 5.5.

Pendant chaque session de la simulation, le système génère automatiquement 100 ou 200 objets à l'amont de fluide, l'utilisateur doit collecter le plus d'objet possible avec un récepteur.

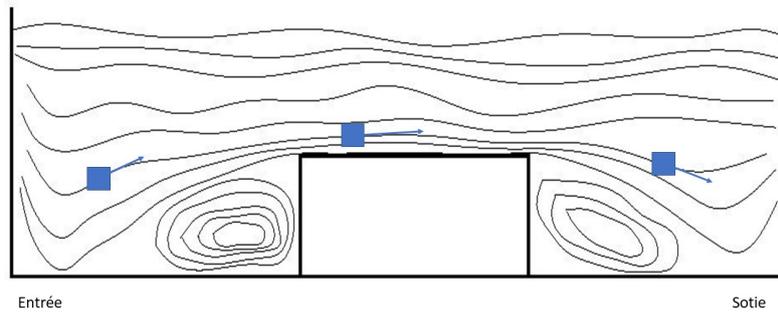


Figure 5.5: Illustration du modèle de mouvement des objets dans le fluide 2D

Le récepteur d'objet peut être déplacé dans une zone prédéfinie. Le participant peut déplacer le récepteur avec la souris dans cette zone pour collecter le plus d'objets possible. La vitesse des fluides, la quantité d'objets et les obstacles ont été déterminés pendant des tests préliminaires pour que la tâche soit équilibrée. En effet, quand il y a trop d'objets, ou les objets déplacent trop vite, les participants déplacent le récepteur totalement aléatoirement. Si la tâche est trop facile, par exemple le courant est trop lent ou bien qu'il y a trop peu d'objets, on ne peut pas observer de différence entre les différentes conditions de désynchronisation car tous les objets sont collectés.



Figure 5.6: Visualisation présentée à l'utilisateur des objets dans la fluide

Participants

16 sujets (9 hommes et 7 femmes) ont participé notre expérimentation (N = 16). Le moyen de l'âge est 28,6 ($\sigma = 7,1$).

Variables indépendantes

Dans cette expérience, deux variables indépendantes sont contrôlées pour étudier leur impact sur la performance de l'utilisateur pour réaliser les tâches. Voici une description détaillée de ces variables indépendantes :

- Le taux d'asynchronisation, concrétisé par un taux de rafraichissement local des résultats de la simulation
- Le nombre d'objets, permettant de régler la difficulté de la tâche.

Comme dans la figure 5.7, on sépare l'aire de la simulation en plusieurs secteurs. On simule le fait que les résultats dans chaque secteur ne sont pas toujours synchronisés. A l'instant t , dans certain secteur, les résultats affichés sont les données à l'instant t , d'autres secteurs peuvent afficher les données à l'instant $t - 1$, $t - 2$, $t - 3$, etc. Avec différents niveaux d'asynchronisation, on peut savoir l'impact de l'asynchronisation sur l'utilisateur dans une scène de la simulation interactive.

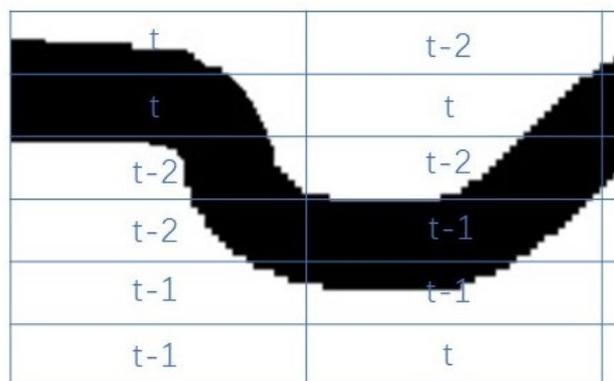


Figure 5.7: Illustration du caractère asynchrone par domaine de notre simulation de fluide

La 2ème variable indépendante est le nombre d'objets flottants simulés, ce qui change la difficulté de la tâche. On a comparé des scènes incluant entre 100 objets et 200 objets si le nombre d'objets change l'impact de la taux d'asynchronisation sur la réalisation de la tâche.

Procédure

Tout d'abord, chaque participant a été invité à signer un consentement éclairé, puis un aperçu de l'expérience a été fourni. Cette expérimentation s'est déroulée en quatre phases principales pour chaque participant :

- La phase de présentation : nous avons expliqué au participant le déroulement de l'expérience, la nature des consignes et la durée de chaque étape.
- La phase de familiarisation : le système génère en continue les objets dans une scène différente de la scène de l'expérimentation.
- La phase d'expérimentation, séparée entre quatre parties. Dans la première partie, Les objets apparaissent en amont du courant qui se situe à gauche de la scène et termine par la droite de la scène. Le participant peut déplacer un récepteur avec souris. Le récepteur déplace avec la souris dans une zone prédéfinie de la scène. Si le curseur de souris est dehors la zone, le récepteur reste au bord de la zone. En déplaçant le récepteur, le participant doit collecter le plus d'objet possible. Tous les participants ont testé une scène avec la condition de contrôle synchrone, et 4 scènes avec chacune un taux différent d'asynchronisation. Le taux est en revanche identique pour toutes les zones de la simulation. Le système génère 100 objets en total pendant chaque session. Dans la seconde partie, c'est analogue à la première partie, mais le système génère 200 objets au total pendant chaque session. Dans la troisième partie, analogue à la première partie, le taux d'asynchronisation sont différents dans chaque zone de simulation. Ces trois parties contiennent toutes une session de contrôle, c'est à dire une Simulation synchrone avec les positions de tous les objets affichés en temps interactif. Dans la quatrième partie, un seul objet est généré. Le participant doit mémoriser la trajectoire de l'objet. Quand l'objet arrive à la bordure droite, un curseur apparaît sur une position aléatoire. Le participant peut déplacer le curseur vers le haut ou le bas (sur une ligne verticale). La tâche demandée consiste à mettre le curseur à la position où l'objet est passé sur la ligne verticale.

Les trois premières parties sont dans l'ordre aléatoire pour éviter l'effet d'apprentissage. La durée de la scène avec chaque condition est d'environ une minute.

5.2.5 Données mesurées

Pendant les trois premières parties de l'expérimentation, on mesure le nombre d'objets collectés par les participants. On calcule l'efficacité de chaque utilisateur dans chaque condition.

Pendant la quatrième partie, la distance entre la position de l'objet et la position du curseur défini par le participant est mesurée.

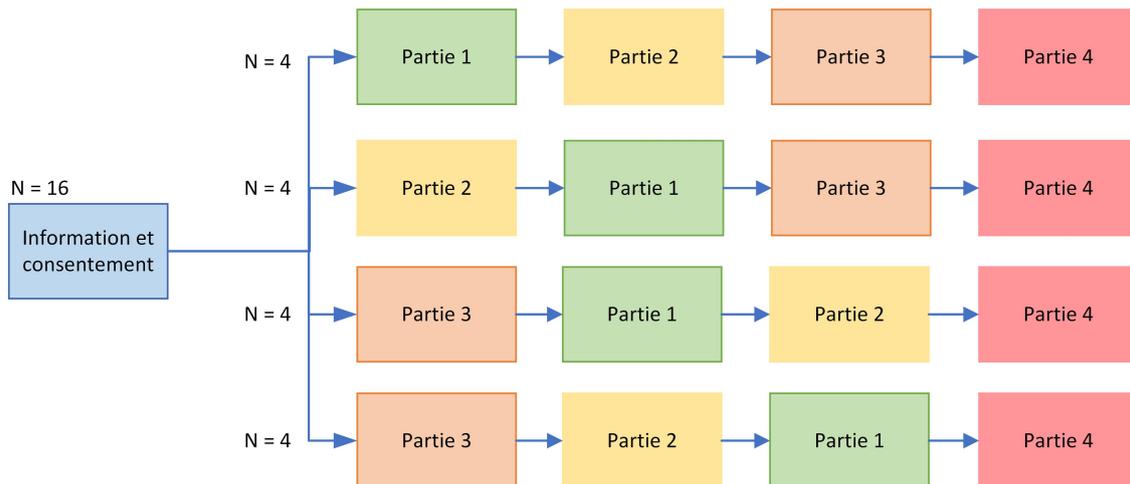


Figure 5.8: Répartition des participants : 16 participants ont été divisés en 4 groupes pour éviter l'effet d'apprentissage. Chaque condition contient une session de contrôle et 4 sessions avec les taux d'asynchronisation différents. La condition 4 est à la fin de l'expérimentation avec une tâche et une scène différente.

5.2.6 Hypothèses

- Une légère asynchronisation de la simulation n'a pas d'impact évident sur la collection des objets mais un taux d'asynchronisation important a un impact visible si la tâche demande une réaction rapide et successive.
- L'impact de l'asynchronisation est plus important si la tâche est plus difficile.
- L'impact de l'asynchronisation ne change pas si les zones ont des taux d'asynchronisation différents.
- L'impact de l'asynchronisation est négligeable si la tâche n'est pas difficile et demande l'analyse ou la mémorisation par le participant.

5.3 Résultats

Les résultats présentés dans cette section ont été considérés comme statistiquement significatifs lorsque $p < 0,05$. Toutes les analyses ont été effectuées à l'aide de *R*. Dans les graphiques à barres, les barres d'erreur indiquent l'erreur type (SE).

5.3.1 Résultats quantitatifs

Nous avons constaté que les données n'étaient pas toutes normalement distribuées dans un test de Shapiro. Dans la partie suivante de cette section, nous avons effectué le test de Wilcoxon

Signed Rank. Nous avons enregistré 320 essais : 4 parties × 5 conditions × 16 participants.

Le test de Friedman a montré que le taux d'asynchronisation a un impact sur la performance ($p < 0,0001$) de la collection des objets. On a utilisé le test de Nemenyi pour poursuivre la différence entre les conditions avec calcul synchronisé. Le test de Nemenyi a montré que la condition 2 ($p = 0,92$) et 3 ($p = 0,56$) ne sont pas significativement moins efficaces que la condition 1. La condition 4 ($p = 0,018$) et 5 ($p = 0,001$) sont significativement moins efficaces que la condition 1.

Pour le nombre d'objet, le test de Friedman a montré que les tâches avec 200 objets n'ont pas été plus impacté par l'asynchronisation ($p = 0,74$).

Le test de Friedman a montré que les différents taux d'asynchronisation n'ont pas plus d'impact sur l'efficacité pour finaliser la tâche de collection des objets ($p = 0,27$).

Finalement pour la tâche de la mémorisation des trajectoires d'objet, le test de Friedman a montré que l'asynchronisation n'a pas d'impact significative ($p = 0,16$).

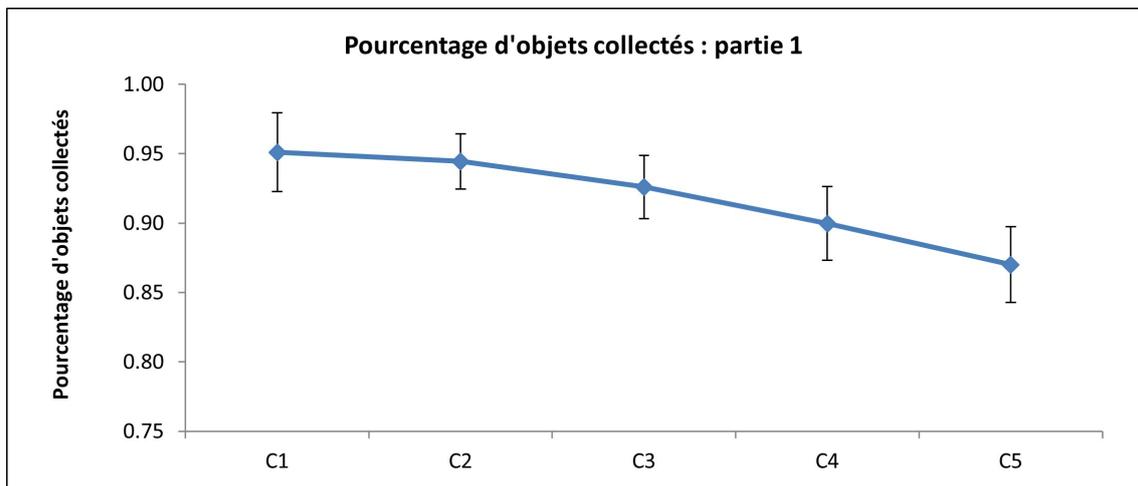


Figure 5.9: Résultats quantitatives : pourcentage d'objets collectés sous condition 0 (temps réel), condition 1 (20 ms de retard), condition 2 (40 ms de retard), condition 3 (60 ms de retard), condition 4 (80 ms de retard)

5.3.2 Discussion

À partir des résultats, nous pouvons tout d'abord mettre en évidence que les participants ont collecté 8,4% moins d'objet avec la condition 5 et 5,3% moins d'objet avec la condition 4, mais que les conditions 3 et 2 ne font pas ressortir de différence significative avec la Partie 1 (H1). Cette tâche demande à l'utilisateur des réactions successives rapides. Dans ce cas, les résultats affichés en retard, du fait du calcul asynchrone, peut avoir un impact important sur la performance de l'utilisateur. Selon l'expérimentation effectuée, l'asynchronisation de plus de 60 ms ont un impact important, mais est négligeable si l'asynchronisation est de moins 40 ms.

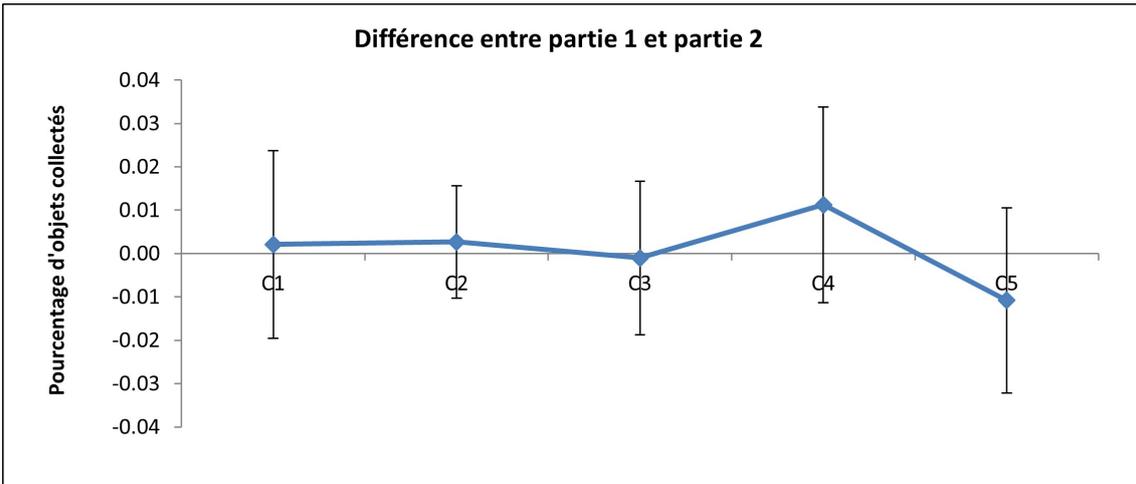


Figure 5.10: Résultats quantitatifs : différence entre les intensités différentes de la tâche. 100 objets (partie 1) vs 200 objets (partie 2)

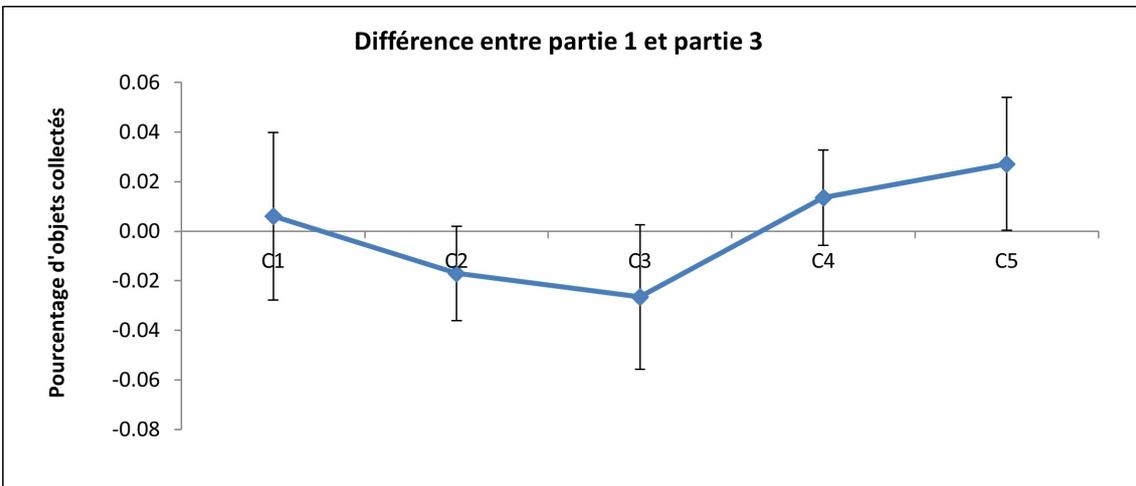


Figure 5.11: Résultats quantitatifs : différence entre le même taux d'asynchronisation (Partie 1) et différents taux d'asynchronisation (Partie 3) dans les zones de simulation

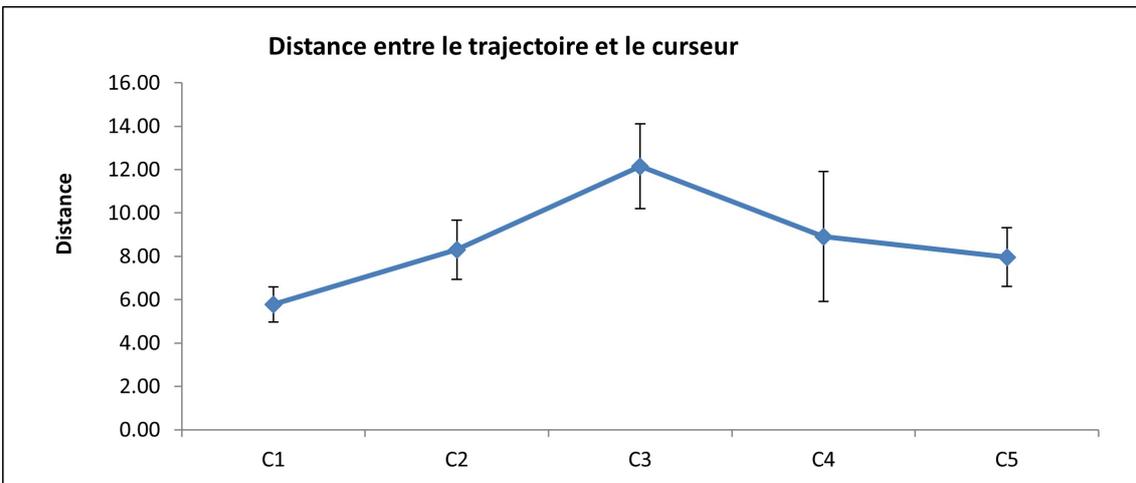


Figure 5.12: Résultats quantitatifs : la distance entre le trajectoire de l'objet et le curseur posé par l'utilisateur

Nous avons également constaté que le nombre d'objets dans la scène n'impacte pas les résultats, ce qui rejette H2. Le nombre d'objets représente la difficulté de la tâche pour notre scène d'expérimentation.

Pour mimer de manière plus réaliste un cas réel du calcul asynchrone, nous avons prévu une condition avec l'asynchronisation aléatoire par zone (Partie 3). Le calcul asynchrone est susceptible de résoudre la problématique liée à un travail déséquilibré, mais la fréquence de synchronisation des résultats dans chaque zone n'est pas homogène. Dans cette condition expérimentale, les résultats semblent montrer que la performance de l'utilisateur pour réaliser la tâche est identique à la première condition. Nous expliquons ce résultat par des moyennes de retard d'affichage analogues à la condition ou une fréquence de synchronisation homogène. Au regard de ces résultats, on peut conclure qu'une distribution non homogène de travail dans chaque zone ne change globalement pas la performance de l'utilisateur (H3).

Enfin, on a testé une tâche qui consiste à demander à l'utilisateur de mémoriser la trajectoire d'un objet sur quelques secondes. Nous n'avons pas trouvé de différence significative entre le calcul synchrone et le calcul asynchrone (H4) pour cette tâche. Nous expliquons ces résultats par le fait que la mémorisation est suffisamment imparfaite, pour que l'impact causé par les conséquences d'une méthodologie asynchrone n'a que peu d'impact sur cette tâche de mémorisation.

5.4 Conclusion

Nous avons dans ce chapitre présenté un travail exploratoire concernant l'usage de méthodologie asynchrone sur les performances d'une tâche de simulation interactive consistant à récupérer des objets transportés par un fluide 2D.

Après une implémentation de simulation de phénomène simple avec un modèle asynchrone pour évaluer la faisabilité et l'utilisabilité du framework asynchrone dans un contexte de simulation interactive *JACK2* [95], nous avons utilisé notre plate-forme de simulation interactive en y ajoutant l'implémentation d'une simulation de fluide en utilisant une méthode LBM intégrée. Cette implémentation relevant d'un couplage fort nous a été imposée par la nécessité pour nos expérimentations d'avoir une application facilement téléchargeable et exécutable sur un seul ordinateur mobile, en dehors du laboratoire, pour cause de confinement.

Pour résumer nos résultats préliminaires sur l'intérêt des méthodologies asynchrones dans un contexte de simulation interactive, nous avons d'abord illustré l'apport de ce modèle sur la performance en cas de déséquilibre de charge, nécessairement induit par l'interactivité sur les conditions aux limites. Ensuite concernant l'expérimentation simulant le caractère partiel de modèles

asynchrones pour mesurer l'impact de cette spécificité sur l'expérience utilisateur en contexte de simulation interactive, les premiers résultats montrent qu'un retard de synchronisation de 40 millisecondes n'a pas d'impact significatif sur la performance de l'utilisateur pour réaliser les tâches, que ce retard soit homogène ou pas. Au delà de 40 millisecondes, l'impact sur l'utilisateur devient significatif puis rédhibitoire à 60 millisecondes. C'est donc le retard global qui compte, ce qui nous fait conclure que l'hétérogénéité des résultats caractéristique du modèle asynchrone n'est pas un facteur bloquant en dessous d'un seuil de retard moyen. Enfin sur une tâche de mémorisation l'impact de l'asynchronisation est lui aussi négligeable, même avec le retard de synchronisation de 80 millisecondes. Nous supposons que ces résultats sont dû aux limites de mémorisation des sujets, le retard étant un facteur négligeable pour la réalisation de la tâche.

A l'issue de ces travaux préliminaires, nous sommes convaincus que la technique de calcul asynchrone est une approche prometteuse dans le cadre de simulation interactive. Au delà du fait que ces modèles offrent des performances plus importantes que les modèles garantissant une meilleure qualité d'interaction, l'impact de l'asynchronisme et de la non-homogénéité des résultats, caractéristique des modèles asynchrones ne semble pas être un facteur bloquant pour l'usage de modèles asynchrones.

Pour confirmer et mesurer plus précisément l'apport de modèles de calcul asynchrone pour la simulation interactive de fluide, il est cependant nécessaire de compléter ces travaux préliminaires. Appliquer et évaluer des modèles asynchrones à des modèles de simulation plus complexes de type LBM, pour aller au delà des calculs simples de diffusion de chaleur, et les appliquer à des simulations de fluide 3D, n'est pas trivial et est assez fastidieuse, d'autant plus que la variété des modèles LBM est importante. Ce travail pourrait faire l'objet du travail d'une ou plusieurs thèses.

Conclusion et perspectives

Cette thèse s'articule autour de simulation interactive de fluide dont la caractéristique essentielle est de donner la possibilité de contrôler les éléments simulés et leur environnement pendant une simulation en cours.

De nombreuses études ont été menées dans le but d'appliquer des approches par simulations interactives et par jeux sérieux, dans le domaine de l'éducation, de la formation, que ce soit dans le milieu académique ou industriel. Pour augmenter le réalisme de la visualisation et de l'interaction, notamment dans un contexte de formation, de nombreuses applications intégrant une dimension ludique mais sérieuse, et l'usage de simulation interactive pour plus de réalisme, sont développées pour des environnements de Réalité Virtuelle ou Augmentée, et sont bien documentées dans la littérature.

Dans toutes ces approches, nous pouvons cependant remarquer qu'elles n'adressent que très rarement des problématiques liées à simulation de fluides réaliste. Cette thèse s'inscrit dans ce contexte avec la perspective d'appliquer la simulation interactive pour la simulation de fluide réaliste dans le cadre de jeu sérieux, que ce soit dans un contexte interactif classique de bureau, ou dans un contexte immersif de Réalité Virtuelle. L'objectif est d'arriver au même niveau de développement et d'usage que celui qui est obtenu en dynamique moléculaire, avec des outils de simulations interactives disponibles depuis une quinzaine d'années, jusqu'au jeu collaboratif *Fold'It* permettant à des groupes de joueurs de comprendre et même de résoudre des problématiques complexes de repliement des protéines.

Ce type d'approche par simulation interactive et immersive et par jeu sérieux, appliqué à la mécanique des fluides, nous paraît avoir un grand potentiel pour l'enseignement et la formation, pour adresser des problématiques de prises de décision, pour résoudre des problèmes d'optimisation et de contrôle, dans une démarche centrée utilisateur, complémentaire à des approches par ap-

prentissage automatique.

Pour atteindre cet objectif, de nombreux travaux sont nécessaires dans le domaine de la simulation de fluide, notamment pour les adapter au niveau de performance requise, et pour les rendre moins sensibles à des éditions interactives pendant une simulation en cours. Il s'agit par ailleurs d'adresser des problématiques d'interaction, pour répondre à une large gamme de fonctionnalités interactives pour contrôler un fluide et son environnement, ainsi que les paramètres de sa simulation. Il s'agit non seulement de choisir ou développer les techniques d'interaction adaptées, mais aussi de les évaluer, dans des environnements numériques classiques de bureau jusqu'aux environnements immersifs de Réalité Virtuelle.

Les problématiques de visualisation et d'interaction ont été largement étudiées et très présentes dans la littérature, et ne sont donc pas la contribution centrale de ce travail. En revanche, l'apport des simulations interactives par rapport aux simulations classiques, notamment pour la prise de décision, le choix et le développement et l'évaluation de modèles de calcul adaptés au contexte de simulation interactive, et plus généralement les problématiques liées au développement, à la diffusion et à l'usage d'outils, d'applications ou de jeux sérieux dans le domaine de la mécanique des fluides, sont finalement assez peu étudiés. C'est dans ce cadre que s'inscrivent les contributions de ce travail de doctorat, présentées ci-dessous.

Contributions

Le progrès dans la performance de calcul intensif en simulation numérique associées aux avancées méthodologiques en simulation des fluides, est un contexte qui nous permet désormais d'envisager des approches abouties de simulation des fluides interactives.

Cependant, il reste beaucoup de contraintes liées à l'interactivité, notamment en termes de performance et de sensibilité des simulations, qui imposent un travail spécifique sur les modèles de simulation, et sur leurs articulations avec les fonctionnalités et les outils de visualisation et d'interaction nécessaires à la mise en place de systèmes de simulation interactif dédiés à la mécanique des fluides.

Une des contributions, essentiellement technique fut donc dans un premier temps de travailler sur l'articulation entre composants de simulation d'une part, et composants de visualisation et d'interaction d'autre part, en posant les premières bases d'une plateforme de simulation interactive en mécanique des fluides dédié au jeu sérieux. Un des premiers choix fut d'opter pour un modèle par couplage faible, avec l'objectif que notre approche soit suffisamment modulaire pour s'adapter à n'importe quel outil de simulation.

Dans ce travail, nous avons par exemple éprouvé notre approche par simulation de fluide interactif par l'usage d'une méthode de type Boltzmann sur réseau, par l'adaptation du code *Palabos* libre de droit et disponible sur l'étagère. La méthode de Boltzmann sur réseau peut être en effet facilement parallélisée et donc répondre aux exigences de performance de l'interactivité. Elle est aussi numériquement et théoriquement moins sensible aux éditions interactives que les méthodes classiques basées sur la résolution des équations de Navier-Stokes.

Par ailleurs, l'objectif n'étant pas de redévelopper depuis le départ une plateforme interactive et de visualisation en mécanique des fluides, nous avons pour le module d'interaction et visualisation choisi *Unity3D* pour offrir les fonctions élémentaires de visualisation et interaction. Au delà des capacités de *Unity3D* à offrir des fonctionnalités de visualisation et de rendu graphique, cet environnement de développement de jeu vidéo offre aussi toute une gamme de fonctionnalités interactives initialement conçues pour le jeu vidéo. Cela nous a permis de développer les fonctionnalités interactives élémentaires de contrôle du fluide et de modifier son environnement pendant une simulation en cours.

La raison pour laquelle on a choisi d'utiliser *Unity 3D*, en lieu et place d'outils de visualisation scientifique sur l'étagère comme *Paraview*, est son niveau de flexibilité qui nous a permis de développer les fonctionnalités interactives sur une simulation en cours, fonctions peu présentes ou nécessitant des développements importants sur d'autres plateformes. Une autre motivation du développement de notre plateforme sous *Unity 3D* est l'usage direct sans développement lourd de dispositifs de Réalité Virtuelle avec *SteamVR*, qui est un contexte adressé dans notre travail. Enfin, dans une perspective d'usage de notre plateforme en jeu sérieux, *Unity 3D* est l'environnement idéal pour la gamification de tâche de simulation de fluide, puisque cet environnement a été conçu pour le développement de jeux vidéo.

Pour assurer cette approche de couplage faible, une des problématiques est l'articulation performante entre la plateforme de visualisation et d'interaction, et le composant de simulation. Cette articulation doit être générique pour être adaptable à la plus grande variété possible d'outils de simulation. Nous avons développé la librairie et le protocole *CFDriver* pour supporter la communication entre les deux systèmes, dans un sens pour exposer des fonctionnalités pour récupérer les résultats de simulation à chaque itération coté *Unity 3D*, dans l'autre sens pour envoyer les commandes d'édition au composant dédié au calcul.

La réalisation de cette plateforme n'était évidemment pas une fin en soit, puisque les objectifs de ce travail sont d'étudier l'apport du contexte de simulation interactive par rapport au simulation classique, l'apport du contexte immersif par rapport au contexte classique, et enfin l'apport de modèles comme LBM ou des modèles asynchrones aux approches par simulation interaction de

fluide.

Nos contributions en matière expérimentale se divisent donc en trois parties. Nous avons d'abord réalisé une expérimentation pour comparer l'approche par simulation interactive par rapport à une approche par simulation classique. Les résultats montrent que la simulation interactive est plus efficace pour la prise de décision, et offre une meilleure expérience utilisateur, dans un scénario de la simulation des fluides consistant à éviter le débordement d'une vague, expérimentations et résultats présentés dans le chapitre 3.

Nous avons ensuite tenté de mesurer l'apport de la Réalité Virtuelle dans un contexte de simulation interactive de fluide. Une expérimentation analogue a donc été menée, pour comparer cette fois deux environnements d'interaction. Les résultats montrent que l'environnement virtuel a facilité les déplacements et l'exploration du phénomène, et permet de minimiser le nombre d'essais, mais sans diminuer l'efficacité en termes de temps de la prise de décision. Ces résultats sont présentés dans le chapitre 4.

Enfin nous avons dans un travail très exploratoire essayé de mesurer les apports de modèle de calcul asynchrone dans un contexte de simulation interactive de fluide. Au delà du fait que le calcul asynchrone permet d'améliorer la performance de la simulation susceptible d'améliorer l'expérience d'interaction, cette méthodologie permet de répondre à la problématique d'équilibrage de charge entre les processeurs, qui peut être variable dans un contexte de simulation interactif. Nous avons donc d'abord illustré le gain de performance de calcul asynchrone sur des simulations simples. Puis nous avons conçu une expérimentation pour évaluer l'impact du calcul asynchrone sur la prise de décision de l'utilisateur pendant une simulation interactive. Dans cette expérimentation, nous avons simulé le fait que les résultats de la simulation asynchrone dans chaque secteur ne sont ni homogènes ni synchronisés pour mesurer cette spécificité sur la prise de décision et l'expérience utilisateur.

Perspectives

Les résultats, techniques et expérimentaux de travail de thèse sont un premier pas pour concrétiser des approches par simulation interactive de fluide et de jeu sérieux. Ce travail ouvre de nombreuses directions à explorer, dans le domaine du jeu sérieux, de l'interaction, de la Réalité Virtuelle, et de l'adaptation des modèles de calcul pour les simulations interactives.

Avec le choix de *Unity 3D* pour notre plateforme de simulation interactive de fluides, nous offrons les premières briques d'un environnement de développement propice à la mise en place de jeu sérieux nécessitant l'usage de simulation de fluide, dans des contextes allant du web à la

Réalité Virtuelle et augmentée. Nous soulignons le fait que notre approche par couplage faible, permet d'envisager l'usage de simulation de fluide réaliste en donnant la possibilité d'intégrer des outils de simulation reconnus et sur l'étagère, ou des codes de laboratoire.

Une des perspectives de notre travail est aussi de faire émerger des jeux sérieux à l'image de ce qui est fait dans d'autres domaines [10], mais appliqués à la mécanique des fluides, avec des objectifs d'application en matière de prise de décision, d'enseignement et de recherche notamment dans des approches d'optimisation centrées utilisateurs. Les travaux et les modèles pour guider le design de jeux sérieux sont nombreux, ainsi que ceux qui ont proposé des méthodologies d'évaluation dédiées [108], travaux sur lesquels il est possible de s'appuyer pour concevoir des jeux sérieux en mécanique des fluides, notamment pour faciliter la conception des scénarios pour permettre aux étudiants de mieux comprendre les phénomènes de la physique des fluides.

Les jeux sérieux permettent aussi aux apprenants d'adapter l'apprentissage à leurs besoins cognitifs et d'augmenter motivation de l'étudiant[100]. Inspiré par *Fold'it* [78], nous avons donc développé notre plateforme comme base pour concevoir des jeux sérieux pour illustrer les concepts de la mécanique des fluides à travers des scénarios appropriés conçus avec des experts, mais aussi pour savoir si les approches de jeux sérieux et de science citoyenne pourraient résoudre des problèmes de conception ou d'optimisation en mécanique des fluides.

Concernant les perspectives en matière de recherche en méthodologie de simulation de fluides, nous avons à peine effleuré la problématique de l'interactivité, avec l'usage de simulation de type LBM. Par ailleurs, les modèles asynchrones semblent prometteurs pour accélérer les simulations et pour résoudre des problématiques d'équilibrage de charge, potentiellement remises en cause par l'édition interactive d'un fluide, voire à moyen terme, de la topologie de la simulation, pour adapter le degré de précision en fonction du besoin ou de l'interaction au cours de la simulation. Il reste de nombreux travaux à mener pour étudier la problématique de stabilité liée à la modification des conditions limites au cours de la simulation, la méthode LBM utilisée dans cette thèse, ou SPH, étant les plus favorables, comparées aux approches classiques par résolution des équations de Navier Stokes, ou dans le cas d'un nombre de Reynolds important. Pour les simulations multiphasées, il est aussi d'adresser la problématique de l'évaluation de la consistance physique lors d'éditions interactives. De manière générale, le changement en temps interactifs des conditions limites, la modification des propriétés physiques d'un fluide pendant une simulation en cours, ouvrent donc de très nombreuses questions de recherche.

Bibliography

- [1] E. Abdul Rahim, A. Duenser, M. Billingham, A. Herritsch, K. Unsworth, A. Mckinnon, and P. Gostomski. A desktop virtual reality application for chemical and process engineering education. In *Proceedings of the 24th Australian Computer-Human Interaction Conference*, pages 1–8, 2012.
- [2] I. Alduán and M. A. Otaduy. Sph granular flow with friction and cohesion. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics symposium on computer animation*, pages 25–32, 2011.
- [3] G. M. Amdahl. Validity of the single processor approach to achieving large scale computing capabilities. In *Proceedings of the April 18-20, 1967, Spring Joint Computer Conference*, AFIPS '67 (Spring), pages 483–485, New York, NY, USA, 1967. ACM.
- [4] D. Amemado. Changements et évolution des universités conventionnelles sous l'influence des technologies de l'information et de la communication (tic): le cas du contexte universitaire nord-américain. 2010.
- [5] R. M. Andrei, M. Callieri, M. F. Zini, T. Loni, G. Maraziti, M. C. Pan, and M. Zoppè. Intuitive representation of surface properties of biomolecules using bioblender. *BMC Bioinformatics*, 13(4):S16, 2012. doi: 10.1186/1471-2105-13-S4-S16. URL <https://doi.org/10.1186/1471-2105-13-S4-S16>.
- [6] L. A. Annetta. The “i’s” have it: A framework for serious educational game design. *Review of General Psychology*, 14(2):105–113, 2010.
- [7] P. Apostolellis and D. A. Bowman. Evaluating the effects of orchestrated, game-based learning in virtual environments for informal education. In *Proceedings of the 11th Conference on Advances in Computer Entertainment Technology*, pages 1–10, 2014.

- [8] M. Asenov, M. Rutkauskas, D. Reid, K. Subr, and S. Ramamoorthy. Active localization of gas leaks using fluid simulation. *IEEE Robotics and Automation Letters*, 4(2):1776–1783, 2019.
- [9] O. K.-C. Au, C.-L. Tai, and H. Fu. Multitouch gestures for constrained transformation of 3d objects. In *Computer Graphics Forum*, volume 31, pages 651–660. Wiley Online Library, 2012.
- [10] P. Backlund, H. Engstrom, M. Johannesson, and M. Lebram. Games and traffic safety-an experimental study in a game-based simulation environment. In *2007 11th International Conference Information Visualization (IV'07)*, pages 908–916. IEEE, 2007.
- [11] J. Bahi, S. Domas, and K. Mazouzi. Jace: a java environment for distributed asynchronous iterative computations. In *12th Euromicro Conference on Parallel, Distributed and Network-Based Processing, 2004. Proceedings.*, pages 350–357, Feb 2004. doi: 10.1109/EMPDP.2004.1271465.
- [12] J. M. Bahi, S. Contassot-Vivier, R. Couturier, and F. Vernier. A decentralized convergence detection algorithm for asynchronous parallel iterative algorithms. *IEEE Transactions on Parallel and Distributed Systems*, 16(1):4–13, 2005.
- [13] J. M. Bahi, R. Couturier, and P. Vuillemin. Jacep2p: an environment for asynchronous computations on peer-to-peer networks. In *2006 IEEE International Conference on Cluster Computing*, pages 1–10, Sep. 2006. doi: 10.1109/CLUSTER.2006.311906.
- [14] J. M. Bahi, S. Contassot-Vivier, and R. Couturier. *Parallel iterative algorithms: from sequential to grid computing*. Chapman and Hall/CRC, 2007.
- [15] J. M. Bahi, R. Couturier, and P. Vuillemin. Jacev: A programming and execution environment for asynchronous iterative computations on volatile nodes. In M. Daydé, J. M. L. M. Palma, Á. L. G. A. Coutinho, E. Pacitti, and J. C. Lopes, editors, *High Performance Computing for Computational Science - VECPAR 2006*, pages 79–92, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. ISBN 978-3-540-71351-7.
- [16] Y. B. Bao and J. Meskas. Lattice boltzmann method for fluid simulations. *Department of Mathematics, Courant Institute of Mathematical Sciences, New York University*, page 44, 2011.

- [17] D. P. Bertsekas and J. N. Tsitsiklis. Convergence rate and termination of asynchronous iterative algorithms. In *Proceedings of the 3rd International Conference on Supercomputing, ICS '89*, pages 461–470, New York, NY, USA, 1989. ACM. ISBN 0-89791-309-4.
- [18] P. L. Bhatnagar, E. P. Gross, and M. Krook. A model for collision processes in gases. i. small amplitude processes in charged and neutral one-component systems. *Physical review*, 94(3):511, 1954.
- [19] A. Bork. Computer-based instruction in physics. *Physics Today*, 34(9):24–30, 1981.
- [20] A. Bork. Computers in learning physics: what should we be doing? In *Conference on Computers in Physics Instruction*, 1988.
- [21] R. P. Botchen, D. Weiskopf, and T. Ertl. Texture-based visualization of uncertainty in flow fields. In *Visualization, 2005. VIS 05. IEEE*, pages 647–654. IEEE, 2005.
- [22] P. Boulanger. Application of augmented reality to industrial tele-training. In *First Canadian Conference on Computer and Robot Vision, 2004. Proceedings.*, pages 320–328. IEEE, 2004.
- [23] T. Brandvik and G. Pullan. Acceleration of a two-dimensional euler flow solver using commodity graphics hardware. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 221(12):1745–1748, 2007.
- [24] T. Brandvik and G. Pullan. Acceleration of a 3d euler solver using commodity graphics hardware. In *46th AIAA aerospace sciences meeting and exhibit*, page 607, 2008.
- [25] I. Brasil, F. MENDES NETO, J. Silva Chagas, R. Lima, D. Souza, M. Bonates, and A. Dantas. An intelligent agent-based virtual game for oil drilling operators training. pages 9 – 17, 06 2011. doi: 10.1109/SVR.2011.13.
- [26] S. Brenner, Z. Zeng, Y. Liu, J. Wang, J. Li, and P. K. Howard. Modeling and analysis of the emergency department at university of kentucky chandler hospital using simulations. *Journal of emergency nursing*, 36(4):303–310, 2010.
- [27] M. Bull and L. Freeman. *Numerical performance of an asynchronous Jacobi iteration*, volume 634, pages 361–366. 01 2006. doi: 10.1007/3-540-55895-0_431.
- [28] G. C. Burdea. Virtual rehabilitation—benefits and challenges. *Methods of information in medicine*, 42(05):519–523, 2003.

- [29] R. Campbell. Structure fires in health care facilities. 2017.
- [30] R. P. Cant and S. J. Cooper. Simulation in the internet age: The place of web-based simulation in nursing education. an integrative review. *Nurse Education Today*, 34(12): 1435–1442, 2014. ISSN 0260-6917. doi: <https://doi.org/10.1016/j.nedt.2014.08.001>. URL <https://www.sciencedirect.com/science/article/pii/S0260691714002445>.
- [31] H. Casanova, A. Legrand, and Y. Robert. *Parallel algorithms*. Chapman and Hall/CRC, 2008.
- [32] D. Chazan and W. Miranker. Chaotic relaxation. *Linear algebra and its applications*, 2(2): 199–222, 1969.
- [33] M. Chen, S. J. Mountford, and A. Sellen. A study in interactive 3-d rotation using 2-d control devices. In *Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, pages 121–129, 1988.
- [34] J. Clement. Students' preconceptions in introductory mechanics. *American Journal of physics*, 50(1):66–71, 1982.
- [35] J. M. Cohen, S. Tariq, and S. Green. Interactive fluid-particle simulation using translating eulerian grids. In *Proceedings of the 2010 ACM SIGGRAPH symposium on Interactive 3D Graphics and Games*, pages 15–22. ACM, 2010.
- [36] K. Corti. Games-based learning; a serious business application. *Informe de PixelLearning*, 34(6):1–20, 2006.
- [37] H. Courtecuisse, J. Allard, P. Kerfriden, S. P. Bordas, S. Cotin, and C. Duriez. Real-time simulation of contact and cutting of heterogeneous soft-tissues. *Medical image analysis*, 18(2):394–410, 2014.
- [38] R. Couturier and S. Domas. Crac: a grid environment to solve scientific applications with asynchronous iterative algorithms. In *2007 IEEE International Parallel and Distributed Processing Symposium*, pages 1–8, March 2007. doi: 10.1109/IPDPS.2007.370568.
- [39] A. J.-C. de Saint-Venant et al. Théorie du mouvement non-permanent des eaux, avec application aux crues des rivières et à l'introduction des marées dans leur lit. *CR Acad. Sci. Paris*, 73(147-154):5, 1871.
- [40] F. De Vuyst, C. Labourdette, and C. Rey. Gpu-accelerated real-time visualization and interaction for coupled fluid dynamics. *Proceedings CFM 2013*, 2013.

- [41] O. Delalande, N. Férey, G. Grasseau, and M. Baaden. Complex molecular assemblies at hand via interactive simulations. *Journal of Computational Chemistry*, 30(15):2375–2387, 2009. doi: <https://doi.org/10.1002/jcc.21235>. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/jcc.21235>.
- [42] R. DeMaria. Games for health 2006: Dance dance... revolution in fitness. *Retirado de http://www.gamasutra.com/php-bin/news_index.php*, 2006.
- [43] J.-Y. Didier, D. Roussel, M. Mallem, S. Otmane, S. Naudet, Q.-C. Pham, S. Bourgeois, C. Mégard, C. Leroux, and A. Hocquard. AMRA: Augmented Reality Assistance for Train Maintenance Tasks. In *Workshop Industrial Augmented Reality, 4th ACM/IEEE International Symposium on Mixed and Augmented Reality (ISMAR 2005)*, page (Elect. Proc.), Vienna, Austria, Oct. 2005. URL <https://hal.archives-ouvertes.fr/hal-00339457>.
- [44] T. Djukic, V. Mandic, and N. Filipovic. Virtual reality aided visualization of fluid flow simulations with application in medical education and diagnostics. *Computers in biology and medicine*, 43(12):2046–2052, 2013.
- [45] A. Dobrovsky, U. M. Borghoff, and M. Hofmann. Applying and augmenting deep reinforcement learning in serious games through interaction. *Periodica Polytechnica Electrical Engineering and Computer Science*, 61(2):198–208, 2017.
- [46] T. Douillet-Grellier, S. Leclaire, D. Vidal, F. Bertrand, and F. De Vuyst. Comparison of multiphase sph and lbm approaches for the simulation of intermittent flows. *Computational Particle Mechanics*, 6(4):695–720, 2019.
- [47] D. El Baz, A. Frommer, and P. Spiteri. Asynchronous iterations with flexible communication: contracting operators. *Journal of Computational and Applied Mathematics*, 176(1):91–103, 2005.
- [48] N. Elmqvist and J.-D. Fekete. Semantic pointing for object picking in complex 3d environments. In *Proceedings of graphics interface 2008*, pages 243–250. ACM, 2008.
- [49] E. Elnikety, T. Elsayed, and H. E. Ramadan. ihadoop: asynchronous iterations for mapreduce. In *2011 IEEE Third International Conference on Cloud Computing Technology and Science*, pages 81–90. IEEE, 2011.
- [50] S. Eroglu, S. Gebhardt, P. Schmitz, D. Rausch, and T. W. Kuhlen. Fluid sketching—immersive sketching based on fluid flow. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 475–482. IEEE, 2018.

- [51] S. S. Esfahani, X. Zhai, M. Chen, A. Amira, F. Bensaali, J. AbiNahed, S. Dakua, G. Younes, A. Baobeid, R. A. Richardson, et al. Lattice-boltzmann interactive blood flow simulation pipeline. *International journal of computer assisted radiology and surgery*, 15(4):629–639, 2020.
- [52] J. Falah, S. Khan, T. Alfalah, S. F. Alfalah, W. Chan, D. K. Harrison, and V. Charissis. Virtual reality medical training system for anatomy education. In *2014 Science and information conference*, pages 752–758. IEEE, 2014.
- [53] B. Frohlich, S. Barrass, B. Zehner, J. Plate, and M. Gobel. Exploring geo-scientific data in virtual environments. In *Proceedings Visualization'99 (Cat. No. 99CB37067)*, pages 169–173. IEEE, 1999.
- [54] M. García, J. Duque, P. Boulanger, and P. Figueroa. Computational steering of cfd simulations using a grid computing environment. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 9(3):235–245, 2015.
- [55] L. Grossman. The army's killer app. *Time*, 165(9):43–44, 2005.
- [56] A. K. Gunstensen, D. H. Rothman, S. Zaleski, and G. Zanetti. Lattice boltzmann model of immiscible fluids. *Physical Review A*, 43(8):4320, 1991.
- [57] R. Haines and D. Kenwright. On the velocity gradient tensor and fluid feature extraction. In *14th Computational Fluid Dynamics Conference*, page 3288, 1999.
- [58] I. A. Halloun and D. Hestenes. Common sense concepts about motion. *American journal of physics*, 53(11):1056–1065, 1985.
- [59] I. A. Halloun and D. Hestenes. The initial knowledge state of college physics students. *American journal of Physics*, 53(11):1043–1055, 1985.
- [60] C. Hand. A survey of 3d interaction techniques. In *Computer graphics forum*, volume 16, pages 269–281. Wiley Online Library, 1997.
- [61] M. R. Hanzaki and P. Boulanger. Proxy haptics for surgical training. In *2020 22nd Symposium on Virtual and Augmented Reality (SVR)*, pages 134–143. IEEE, 2020.
- [62] M. J. Harris. Fast fluid dynamics simulation on the gpu. *SIGGRAPH Courses*, 220(10.1145):1198555–1198790, 2005.

- [63] S. G. Hart. Nasa-task load index (nasa-tlx); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting*, volume 50, pages 904–908. Sage publications Sage CA: Los Angeles, CA, 2006.
- [64] A. R. Harwood, P. Wenisch, and A. J. Revell. A real-time modelling and simulation platform for virtual engineering design and analysis. In *Proceedings of 6th European Conference on Computational Mechanics (ECCM 6) and 7th European Conference on Computational Fluid Dynamics (ECFD 7)*, pages 11–15, 2018.
- [65] X. He, S. Chen, and R. Zhang. A lattice boltzmann scheme for incompressible multiphase flow and its application in simulation of rayleigh–taylor instability. *Journal of computational physics*, 152(2):642–663, 1999.
- [66] R. Holliman and E. Scanlon. *Mediating science learning through information and communications technology*. Routledge, 2013.
- [67] G. Hristov, P. Zahariev, N. Bencheva, and I. Ivanov. Designing the next generation of virtual learning environments—virtual laboratory with remote access to real telecommunication devices. In *2013 24th EAEEIE Annual Conference (EAEEIE 2013)*, pages 139–144. IEEE, 2013.
- [68] H.-M. Huang, U. Rauch, and S.-S. Liaw. Investigating learners’ attitudes toward virtual reality learning environments: Based on a constructivist approach. *Computers & Education*, 55(3):1171–1182, 2010.
- [69] A. K. Hutzschenreuter, P. A. Bosman, I. Blonk-Altena, J. van Aarle, H. La Poutré, et al. Agent-based patient admission scheduling in hospitals. In *AAMAS (Industry Track)*, pages 45–52. Citeseer, 2008.
- [70] M. J. Ibarra, W. Soto, P. Ataucusi, and E. Ataucusi. Mathfraction: Educational serious game for students motivation for math learning. In *2016 XI Latin American Conference on Learning Objects and Technology (LACLO)*, pages 1–9. IEEE, 2016.
- [71] R. Jalam. Apprentissage automatique et catégorisation de textes multilingues. *PhD Thesis, Université Lumière Lyon, 2*, 2003.
- [72] M. Jiang, R. Machiraju, and D. S. Thompson. A novel approach to vortex core region detection. In *VisSym*, pages 217–225, 2002.

- [73] H. M. Johnsen, M. Fossum, P. Vivekananda-Schmidt, A. Fruhling, and Å. Slettebø. Developing a serious game for nurse education. *Journal of gerontological nursing*, 44(1):15–19, 2018.
- [74] C. R. Johnson and S. G. Parker. A computational steering model applied to problems in medicine. In *Supercomputing'94: Proceedings of the 1994 ACM/IEEE Conference on Supercomputing*, pages 540–549. IEEE, 1994.
- [75] S. S. Jones and R. S. Evans. An agent based simulation tool for scheduling emergency department physicians. In *AMIA Annual Symposium Proceedings*, volume 2008, page 338. American Medical Informatics Association, 2008.
- [76] P. Kaiser, P. Vasak, F. Suorineni, and D. Thibodeau. New dimensions in seismic data interpretation with 3-d virtual reality visualization for burst-prone mines. *Controlling seismic risk*, 33, 2005.
- [77] S. Kavanagh, A. Luxton-Reilly, B. Wuensche, and B. Plimmer. A systematic review of virtual reality in education. *Themes in Science and Technology Education*, 10(2):85–119, 2017.
- [78] F. Khatib, F. DiMaio, S. Cooper, M. Kazmierczyk, M. Gilski, S. Krzywda, H. Zabranska, I. Pichova, J. Thompson, Z. Popović, M. Jaskolski, D. Baker, F. C. Group, and F. V. C. Group. Crystal structure of a monomeric retroviral protease solved by protein folding game players. *Nature Structural & Molecular Biology*, 18(10):1175–1177, 2011.
- [79] P. Kipfer and R. Westermann. Realistic and interactive simulation of rivers. In *Graphics interface*, volume 2006, pages 41–48. Citeseer, 2006.
- [80] K. Klinker, V. Fries, M. Wiesche, and H. Krcmar. Catcare: Designing a serious game to foster hand hygiene compliance in health care facilities. In *Designing the Digital Transformation: DESRIST 2017 Research in Progress Proceedings of the 12th International Conference on Design Science Research in Information Systems and Technology. Karlsruhe, Germany. 30 May-1 Jun.*, pages 20–28. Karlsruher Institut für Technologie (KIT), 2017.
- [81] D. Kushner. Virtual reality's moment. *Ieee Spectrum*, 51(1):34–37, 2014.
- [82] P. Lallemand and L.-S. Luo. Theory of the lattice boltzmann method: Dispersion, dissipation, isotropy, galilean invariance, and stability. *Physical review E*, 61(6):6546, 2000.
- [83] R. S. Laramée, H. Hauser, H. Doleisch, B. Vrolijk, F. H. Post, and D. Weiskopf. The state of the art in flow visualization: Dense and texture-based techniques. In *Computer Graphics Forum*, volume 23, pages 203–221. Wiley Online Library, 2004.

- [84] J. Lee, M. Kim, and D. Y. Lee. A one-dimensional fluid simulation method of branching narrow vessel for real-time angiography simulation. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1828–1831. IEEE, 2017.
- [85] S.-I. E. Lei, Y.-C. Chen, H.-T. Chen, and C.-F. Chang. Interactive physics-based ink splattering art creation. In *Computer Graphics Forum*, volume 32, pages 147–156. Wiley Online Library, 2013.
- [86] F. T. Leighton. *Introduction to parallel algorithms and architectures: Arrays· trees· hypercubes*. Elsevier, 2014.
- [87] T. Lenaerts and P. Dutré. Mixing fluids and granular materials. In *Computer Graphics Forum*, volume 28, pages 213–218. Wiley Online Library, 2009.
- [88] T. Lenaerts, B. Adams, and P. Dutré. Porous flow in particle-based fluid simulations. *ACM Transactions on Graphics (TOG)*, 27(3):1–8, 2008.
- [89] H. Li and J. Pan. Research in wellbore display platform based on virtual reality. *Applied Mechanics and Materials*, 347-350, 03 2013. doi: 10.2991/iccsee.2013.481.
- [90] H. Li and J. H. Pan. Research in wellbore display platform based on virtual reality. In *Applied Mechanics and Materials*, volume 347, pages 2910–2914. Trans Tech Publ, 2013.
- [91] Y. Li, R. Shock, R. Zhang, and H. Chen. Numerical study of flow past an impulsively started cylinder by the lattice-boltzmann method. *Journal of Fluid Mechanics*, 519:273–300, 2004.
- [92] Z. Lv, A. Tek, F. Da Silva, C. Empereur-mot, M. Chavent, and M. Baaden. Game on, science - how video game technology may help biologists tackle visualization challenges. *PLOS ONE*, 8(3):1–13, 03 2013. doi: 10.1371/journal.pone.0057990. URL <https://doi.org/10.1371/journal.pone.0057990>.
- [93] P. Ma, Y. Tian, Z. Pan, B. Ren, and D. Manocha. Fluid directed rigid body control using deep reinforcement learning. *ACM Trans. Graph.*, 37(4), July 2018. ISSN 0730-0301. doi: 10.1145/3197517.3201334. URL <https://doi.org/10.1145/3197517.3201334>.
- [94] J. D. Mackinlay, G. G. Robertson, and S. K. Card. Rapid controlled movement through virtual 3d workspaces (videotape). In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '91, page 455–456, New York, NY, USA, 1991. Association for Computing Machinery. ISBN 0897913833. doi: 10.1145/108844.109000. URL <https://doi.org/10.1145/108844.109000>.

- [95] F. Magoulès and G. G. Benissan. Jack2: An mpi-based communication library with non-blocking synchronization for asynchronous iterations. *Advances in Engineering Software*, 119:116–133, 2018.
- [96] F. Magoulès and G. Gbikpi-Benissan. JACK: an asynchronous communication kernel library for iterative algorithms. *The Journal of Supercomputing*, 73(8):3468–3487, 2016. In press.
- [97] F. Magoulès and G. Gbikpi-Benissan. Distributed convergence detection based on global residual error under asynchronous iterations. *IEEE Transactions on Parallel and Distributed Systems*, 29(4):819–829, 2017.
- [98] F. Magoulès, F.-X. Roux, and G. Houzeaux. *Parallel scientific computing*. Wiley Online Library, 2016.
- [99] T. W. Malone. Toward a theory of intrinsically motivating instruction. *Cognitive Science*, 5(4):333–369, 1981. ISSN 0364-0213. doi: [https://doi.org/10.1016/S0364-0213\(81\)80017-1](https://doi.org/10.1016/S0364-0213(81)80017-1). URL <https://www.sciencedirect.com/science/article/pii/S0364021381800171>.
- [100] T. W. Malone. Toward a theory of intrinsically motivating instruction. *Cognitive science*, 5(4):333–369, 1981.
- [101] D. R. Michael and S. L. Chen. *Serious games: Games that educate, train, and inform*. Muska & Lipman/Premier-Trade, 2005.
- [102] T. Möller and B. Trumbore. Fast, minimum storage ray-triangle intersection. *Journal of graphics tools*, 2(1):21–28, 1997.
- [103] J. J. Monaghan. Smoothed particle hydrodynamics. *Annual review of astronomy and astrophysics*, 30(1):543–574, 1992.
- [104] M. Müller, D. Charypar, and M. H. Gross. Particle-based fluid simulation for interactive applications. In *Symposium on Computer animation*, pages 154–159, 2003.
- [105] M. Müller, R. Keiser, A. Nealen, M. Pauly, M. Gross, and M. Alexa. Point based animation of elastic, plastic and melting objects. In *Proceedings of the 2004 ACM SIG-GRAPH/Eurographics symposium on Computer animation*, pages 141–151, 2004.
- [106] M. Müller, S. Schirm, and M. Teschner. Interactive blood simulation for virtual surgery based on smoothed particle hydrodynamics. *Technology and Health Care*, 12(1):25–31, 2004.

- [107] M. Muratet, P. Torguet, J.-P. Jessel, and F. Viallet. Towards a serious game to help students learn computer programming. *International Journal of Computer Games Technology*, 2009, 2009.
- [108] M. Muratet, E. Delozanne, P. Torguet, and F. Viallet. Serious game and students' learning motivation: effect of context using prog&play. In *international conference on Intelligent Tutoring Systems*, pages 123–128. Springer, 2012.
- [109] M. Muratet, T. Carron, I. N'hari, B. Hasenknopf, B. Grosjean, L. Sixdenier, and R. Vuilleumier. Usages de jeux sérieux pour motiver les étudiants: initiatives d'enseignants et impulsions institutionnelles. *Les Annales de QPES*, 1(3), 2021.
- [110] X. Nie, X. Shan, and H. Chen. A lattice-boltzmann/finite-difference hybrid simulation of transonic flow. In *47th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, page 139, 2009.
- [111] G. M. Nielson and D. R. Olsen Jr. Direct manipulation techniques for 3d objects using 2d locator devices. In *Proceedings of the 1986 workshop on Interactive 3D graphics*, pages 175–182, 1987.
- [112] P. Nolin, A. Stipanovic, M. Henry, Y. Lachapelle, D. Lussier-Desrochers, P. Allain, et al. Clinicavr: Classroom-cpt: A virtual reality tool for assessing attention and inhibition in children and adolescents. *Computers in Human Behavior*, 59:327–333, 2016.
- [113] A. E. Olszewski and T. A. Wolbrink. Serious gaming in medical education: a proposed structured framework for game development. *Simulation in Healthcare*, 12(4):240–253, 2017.
- [114] R. L. Page. Brief history of flight simulation. *SimTecT 2000 Proceedings*, pages 11–17, 2000.
- [115] Z. Peng, Y. Xu, M. Yan, and W. Yin. Arock: an algorithmic framework for asynchronous parallel coordinate updates. *SIAM Journal on Scientific Computing*, 38(5):A2851–A2879, 2016.
- [116] B. Z. Perez, M. M. Marin, and E. I. Perez. Developing a virtual environment for safety training. In *Electronics, Robotics and Automotive Mechanics Conference (CERMA 2007)*, pages 545–550, 2007. doi: 10.1109/CERMA.2007.4367743.

- [117] B. Z. Perez, M. M. Marin, and E. I. Pérez. Developing a virtual environment for safety training. In *Electronics, Robotics and Automotive Mechanics Conference (CERMA 2007)*, pages 545–550. IEEE, 2007.
- [118] A. Perez-Valle and D. Sagasti. A novel approach for tourism and education through virtual vitoria-gasteiz in the 16 th century. In *2012 18th International Conference on Virtual Systems and Multimedia*, pages 615–618. IEEE, 2012.
- [119] K. Perkins, W. Adams, M. Dubson, N. Finkelstein, S. Reid, C. Wieman, and R. LeMaster. Phet: Interactive simulations for teaching and learning physics. *The physics teacher*, 44(1): 18–23, 2006.
- [120] F. H. Post, B. Vrolijk, H. Hauser, R. S. Laramée, and H. Doleisch. Feature extraction and visualization of flow fields. *Eurographics 2002 State-of-the-Art Reports*, 1:69–100, 2002.
- [121] A. Rahouti, R. Lovreglio, S. Datoussaïd, and T. Descamps. Prototyping and validating a non-immersive virtual reality serious game for healthcare fire safety training. *Fire Technology*, pages 1–38, 2021.
- [122] S. Reed, S. Hsi, O. Kreylos, M. Yikilmaz, L. Kellogg, S. Schladow, H. Segale, and L. Chan. Augmented reality turns a sandbox into a geoscience lesson. *Eos*, 97:18–22, 2016.
- [123] S. E. Reed, O. Kreylos, S. Hsi, L. H. Kellogg, G. Schladow, M. B. Yikilmaz, H. Segale, J. Silverman, S. Yalowitz, and E. Sato. Shaping watersheds exhibit: An interactive, augmented reality sandbox for advancing earth science education. In *AGU Fall Meeting Abstracts*, volume 2014, pages ED34A–01, 2014.
- [124] J. L. Reisman, P. L. Davidson, and J. Y. Han. A screen-space formulation for 2d and 3d direct manipulation. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology*, pages 69–78, 2009.
- [125] G. Riva, G. Castelnuovo, and F. Mantovani. Transformation of flow in rehabilitation: the role of advanced communication technologies. *Behavior research methods*, 38(2):237–244, 2006.
- [126] M. K. Robel. The difference between military & civilian wargames. *Dirty Little Secrets*, 2004.
- [127] M. T. Roberts, A. Mohraz, K. T. Christensen, and J. A. Lewis. Direct flow visualization of colloidal gels in microfluidic channels. *Langmuir*, 23(17):8726–8731, 2007.

- [128] T. R. Rohleder, P. Lewkonja, D. P. Bischak, P. Duffy, and R. Hendijani. Using simulation modeling to improve patient flow at an outpatient orthopedic clinic. *Health care management science*, 14(2):135–145, 2011.
- [129] S. Sagheb, A. Bahreman, R. LiKamWa, and B. Lahey. Swish: Shifting weight-based interfaces for simulated hydrodynamics in mixed-reality fluid vessels. In *Proceedings of the Thirteenth International Conference on Tangible, Embedded, and Embodied Interaction*, pages 213–221, 2019.
- [130] I. Santos, L. Soares, F. Carvalho, and A. Raposo. A collaborative virtual reality oil and gas workflow. *International Journal of Virtual Reality*, 11:1–13, 01 2012. doi: 10.20870/IJVR.2012.11.1.2832.
- [131] M. Scaife and Y. Rogers. External cognition: how do graphical representations work? *International journal of human-computer studies*, 45(2):185–213, 1996.
- [132] M. Schirski, A. Gerndt, T. van Reimersdahl, T. Kuhlen, P. Adomeit, O. Lang, S. Pischinger, and C. Bischof. Vista flowlib - framework for interactive visualization and exploration of unsteady flows in virtual environments. In *Proceedings of the Workshop on Virtual Environments 2003, EGVE '03*, page 77–85, New York, NY, USA, 2003. Association for Computing Machinery. ISBN 1581136862.
- [133] X. Shan and H. Chen. Lattice boltzmann model for simulating flows with multiple phases and components. *Physical review E*, 47(3):1815, 1993.
- [134] S. Sharma and S. Otunba. Collaborative virtual environment to study aircraft evacuation for training and education. In *2012 International Conference on Collaboration Technologies and Systems (CTS)*, pages 569–574. IEEE, 2012.
- [135] K. Shoemake. Arcball: A user interface for specifying three-dimensional orientation using a mouse. In *Graphics interface*, volume 92, pages 151–156, 1992.
- [136] B. Solenthaler, J. Schläfli, and R. Pajarola. A unified particle model for fluid–solid interactions. *Computer Animation and Virtual Worlds*, 18(1):69–82, 2007.
- [137] A. H. Squillacote, J. Ahrens, C. Law, B. Geveci, K. Moreland, and B. King. *The paraview guide*, volume 366. Kitware Clifton Park, NY, 2007.
- [138] J. Stam. Stable fluids. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '99*, page 121–128, USA, 1999. ACM Press/Addison-Wesley Publishing Co. ISBN 0201485605. doi: 10.1145/311535.311548.

- [139] J. E. Stone, J. Gullingsrud, and K. Schulten. A system for interactive molecular dynamics simulation. In *Proceedings of the 2001 Symposium on Interactive 3D Graphics*, I3D '01, page 191–194, New York, NY, USA, 2001. Association for Computing Machinery. ISBN 1581132921. doi: 10.1145/364338.364398. URL <https://doi.org/10.1145/364338.364398>.
- [140] S. Succi. *The lattice Boltzmann equation: for fluid dynamics and beyond*. Oxford university press, 2001.
- [141] P. H. Suruagy Perrusi, P. Baksic, and H. Courtecuisse. Interactive Finite Element model of needle insertion and laceration. In *Eurographics 2021 - The 42nd Annual Conference of the European Association for Computer Graphics*, Vienne / Virtual, Austria, May 2021. European Association for Computer Graphics. URL <https://hal.inria.fr/hal-03187709>.
- [142] T. Susi, M. Johannesson, and P. Backlund. Serious games: An overview. 2007.
- [143] D. S. Svyetlichnyy. Development of precipitation model with the use of the lattice boltzmann method and its application for the rolling process. *Computer Methods in Applied Mechanics and Engineering*, 370:113246, 2020.
- [144] M. R. Swift, W. Osborn, and J. Yeomans. Lattice boltzmann simulation of nonideal fluids. *Physical review letters*, 75(5):830, 1995.
- [145] J. Tan and X. Yang. Physically-based fluid animation: A survey. *Science in China Series F: Information Sciences*, 52(5):723–740, 2009.
- [146] L. Terenzi and P. Zaal. Rotational and translational velocity and acceleration thresholds for the onset of cybersickness in virtual reality. In *AIAA Scitech 2020 Forum*, page 0171, 2020.
- [147] N. Thürey, C. Körner, and U. Rüde. Interactive free surface fluids with the lattice boltzmann method. *Technical Report05-4. University of Erlangen-Nuremberg, Germany*, 2005.
- [148] N. Umetani and B. Bickel. Learning three-dimensional flow for interactive aerodynamic design. *ACM Transactions on Graphics (TOG)*, 37(4):1–10, 2018.
- [149] G. Vassilacopoulos. A simulation model for bed allocation to hospital inpatient departments. *Simulation*, 45(5):233–241, 1985.
- [150] D. Vergara, J. Extremera, M. P. Rubio, and L. P. Dávila. Meaningful learning through virtual reality learning environments: A case study in materials engineering. *Applied Sciences*, 9(21):4625, 2019.

- [151] J. M. Vézien, B. Ménélas, J. Nelson, L. Picinali, P. Bourdot, M. Ammi, B. F. Katz, J. M. Burkhardt, L. Pastur, and F. Lusseyran. Multisensory vr exploration for computer fluid dynamics in the corsaire project. *Virtual Real.*, 13(4):257–271, 2009. ISSN 1359-4338.
- [152] M. Wang, N. Férey, P. Bourdot, and F. Magoulès. Interactive 3d fluid simulation: steering the simulation in progress using lattice boltzmann method. In *2019 18th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*, pages 72–75. IEEE, 2019.
- [153] M. Wang, N. Férey, P. Bourdot, and F. Magoulès. Using asynchronous simulation approach for interactive simulation. In *2019 18th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*, pages 84–87, 2019. doi: 10.1109/DCABES48411.2019.00028.
- [154] M. Wang, N. Férey, F. Magoulès, and P. Bourdot. Interactive Simulation for easy Decision-making in Fluid Dynamics. In H. Theisel and M. Wimmer, editors, *Eurographics 2021 - Short Papers*. The Eurographics Association, 2021. ISBN 978-3-03868-133-5. doi: 10.2312/egs.20211022.
- [155] D. Weiskopf and G. Erlebacher. Overview of flow visualization., 2005.
- [156] D. A. Wolf-Gladrow. *Lattice-gas cellular automata and lattice Boltzmann models: an introduction*. Springer, 2004.
- [157] T. L. Woods, S. Reed, S. Hsi, J. A. Woods, and M. R. Woods. Pilot study using the augmented reality sandbox to teach topographic maps and surficial processes in introductory geology labs. *Journal of Geoscience Education*, 64(3):199–214, 2016.
- [158] P. Wouters, E. D. Van der Spek, and H. Van Oostendorp. Current practices in serious game research: A review from a learning outcomes perspective. *Games-based learning advancements for multi-sensory human computer interfaces: techniques and effective practices*, pages 232–250, 2009.
- [159] C. Xie. Interactive heat transfer simulations for everyone. *The Physics Teacher*, 50(4):237–240, 2012.
- [160] X. Yan, Y.-T. Jiang, C.-F. Li, R. R. Martin, and S.-M. Hu. Multiphase sph simulation for interactive fluids and solids. *ACM Transactions on Graphics (TOG)*, 35(4):1–11, 2016.

- [161] J. Zarins and M. Weiland. Progressive load balancing of asynchronous algorithms. In *Proceedings of the Seventh Workshop on Irregular Applications: Architectures and Algorithms, IA3'17*, pages 5:1–5:9, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-5136-2.
- [162] Y. Zhang, X. Ban, F. Du, and W. Di. Fluidsnet: End-to-end learning for lagrangian fluid simulation. *Expert Systems with Applications*, 152:113410, 2020.
- [163] B. Zigon, L. Zhu, and F. Song. Interactive 3d simulation for fluid–structure interactions using dual coupled gpus. *The Journal of Supercomputing*, 74(1):37–64, 2018.
- [164] Q. Zou, G. Gbikpi-Benissan, and F. Magoulès. Asynchronous parareal algorithm applied to european option pricing. In *2017 16th International Symposium on Distributed Computing and Applications to Business, Engineering and Science (DCABES)*, pages 37–40. IEEE, 2017.
- [165] Q. Zou, G. Gbikpi-Benissan, and F. Magoulès. Asynchronous communications library for the parallel-in-time solution of black-scholes equation. In *2017 16th International Symposium on Distributed Computing and Applications to Business, Engineering and Science (DCABES)*, pages 45–48. IEEE, 2017.