

Travail de Maturité en Physique : Guide et Comparaison Phénoménologique entre Simulation de Particules et Simulation Eulérienne

Marco Costa

7 novembre 2023

Table des matières

1	Introduction	3
2	Dérivées partielles et discrétisations	5
2.1	Dérivées	5
2.2	Fonctions de plusieurs variables et dérivées partielles	7
2.3	Opérateur nabla et opérateur laplacien	9
2.3.1	Gradient	10
2.3.2	Divergence	11
2.3.3	Laplacien	13
2.3.4	Rotationnel	14
2.4	Résolution d'équations différentielles	18
3	Méthodologie	21
4	Simulations	22
4.1	Simulation de particules	22
4.1.1	Conditions initiales et paramètres	22
4.1.2	Modèle et équations	22
4.1.3	Discrétisation des équations	25
4.2	Simulation Eulérienne	26
4.2.1	Conditions initiales et paramètres	27
4.2.2	Modèle et équations	27
4.2.3	Discrétisation et résolution des équations	31
4.2.3.1	Calcul de la pression	32
4.2.3.2	Calcul de l'advection	36
4.2.3.3	Calcul de la viscosité (Diffusion)	38
4.2.3.4	Application des calculs	38
5	Résultats	39
5.1	Instabilité de Kelvin-Helmholtz	39
5.1.1	Description du phénomène physique	40
5.1.2	Résultats de la simulation Eulérienne	42
5.1.3	Résultats de la simulation de particules	44
5.2	Rayleigh-Bénard	46

5.2.1	Description du phénomène physique	46
5.2.2	Résultats de la simulation Eulérienne	46
5.2.3	Résultats de la simulation de particules	48
6	Comparaison entre la simulation Eulérienne et la simulation de particules	49
6.1	Efficacité et parallélisation	49
6.2	Gestion de paramètres et flexibilité	50
6.3	Reproduction du phénomène voulu	50
7	Conclusion	50
Références		51
A	Programmes de simulation	54
A.1	Stockage des données	55
A.2	Programmes	56
A.3	Optimisation des simulations	58
B	Utilisation de langages de haut et de bas niveau	58
C	ChatGPT et productivité	59

1 Introduction

L'idée de ce travail de Maturité remonte à l'été 2021 pendant lequel je suis tombé sur une vidéo de la chaîne youtube "ScienceEtonnante" parlant de la thermodynamique de la vie. J'étais intrigué par le concept de "lyfe", dont je me souviens encore la définition : "Structure dissipative capable d'homéostasie, d'autocatalyse et d'apprentissage". Dans la vidéo, David Louapre essaye de reproduire une simulation où une forme de lyfe apparaît. En particulier la première simulation qu'il a réalisée simulait la convection de Rayleigh-Bénard.



FIGURE 1 – Image de la convection de Rayleigh-Bénard extraite de la vidéo de Science Étonnante.



FIGURE 2 – Image du jeu Sand :Box

En plus, durant cette période, j'aimais beaucoup jouer à un jeu mobile de simulation physique appelé "Sand :Box". Dans ce jeu, chaque pixel, représentant un élément manipulable, interagit différemment en fonction de sa nature. Ceci m'avait inspiré pour faire mon propre jeu de simulation visant à simuler des phénomènes nettement plus complexes à l'instar de la corrélation entre haute pression et augmentation de température. Car, en jouant, j'avais justement essayé de reproduire la convection de Rayleigh-Bénard, mais cela s'avéra impossible. Je me suis donc résolu à faire une première simulation ayant pour but de pouvoir reproduire cette instabilité, qui, pour faire court, se produit lorsqu'un fluide, réchauffé par le bas et refroidit par le haut, forme des cellules de convection où le fluide circule. Cependant, dans l'élaboration de ma simulation, je ne m'étais basé sur aucune formule physique, simplement mon imagination. Le résultat s'était révélé peu satisfaisant :

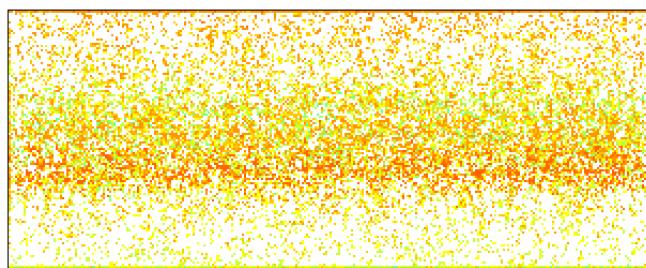


FIGURE 3 – Image du projet ThermicBalls

En arrivant au gymnase, j'en ai parlé avec un élève qui faisait son TM à propos de simulations de particules et suis arrivé à la conclusion que je devais me résoudre à utiliser de vraies équations physiques si je voulais arriver à un résultat satisfaisant. Mais je n'étais plus très motivé, je me suis donc limité à faire des simulations de particules interagissant par des collisions parfaitement élastiques, deux par deux, pendant les cours d'informatique. J'avais en fait mis ce projet sous le tapis en sachant que j'allais le ressortir pour en faire mon Travail de Maturité. Cette fois, j'étais déterminé à réussir à simuler quelque chose s'apparentant à un

fluide !

J'avais donc décidé que le but de ce travail allait être de parvenir à simuler divers phénomènes se produisant dans des milieux fluides avec une simulation de particules. Je savais que j'allais avoir besoin, pour cela, d'une simulation de fluides, de référence, qui fonctionne. J'ai donc, avant que mon TM démarre, commencé à télécharger des simulations de fluides, que je trouvais casuellement, en essayant, ensuite, de bidouiller leur code pour comprendre comment elles fonctionnaient. Mais cela n'a pas été très concluant. J'ai donc changé de stratégie, me suis mis à chercher sur internet "comment programmer une simulation de fluides" et ai trouvé une vidéo¹ qui, justement, expliquait concrètement un algorithme permettant de simuler des fluides incompressibles. En quelques heures et beaucoup de concentration, je me suis alors retrouvé avec une simulation qui fonctionnait.

Mais ce n'était que le début ; certes j'avais fait une simulation de fluides et avais compris à peu près ce que faisaient les différentes étapes de calcul. Mais je n'avais pas la moindre idée des équations qui les régissaient. Ou plutôt, je savais que c'étaient les équations de Navier-Stokes mais ne savais pas du tout comment on pouvait obtenir mon algorithme à partir de celles-ci. En effet, la vidéo ne l'expliquait pas.

Parallèlement à cela, il y eut ce jour qui sera à tout jamais marqué dans l'histoire : le 30 novembre 2022. Oui je parle bien de la date de sortie de chatGPT au grand public. Depuis, il ne s'est pas écoulé un seul jour où il ne m'ait pas aidé dans une tâche ou une autre. Rencontrant déjà depuis plusieurs temps des difficultés à comprendre les équations de Navier-Stokes, au mois de juin, je lui ai donc demandé de trouver des ressources en ligne qui pourraient m'aider. C'est là que j'ai finalement trouvé du pain pour mes dents. Il m'a conseillé le cours "12 steps to Navier-Stokes" de l'université de Boston (disponible sur Youtube) qui dure environ 15 heures et dont j'ai visionné la moitié des vidéos. Cela pourrait paraître peu, mais j'ai énormément appris de ces quelques 16 vidéos. En plus de cela, pour approfondir ma compréhension, je me suis renseigné sur les concepts mathématiques abordés comme par exemple les séries de Fourier ou la résolution d'une équation de Poisson. Les cours de Gilbert Strang sur youtube à ce sujet sont particulièrement intéressants et merveilleusement bien expliqués.

Avec cette compréhension plus approfondie des façons de résoudre des équations aux dérivées partielles, j'ai pu introduire une dimension thermique à mon modèle de fluide, qui était athermal. J'ai également pu déduire comment l'algorithme que j'avais utilisé avait été obtenu à partir des équations de Navier-Stokes, ce que j'explique en détail dans la section 4.2.2.

Pendant ce temps là, la simulation de particules n'a pas évolué ; ses particules sont restées de simples particules interragissant par collisions parfaitement élastique. À l'heure où je finalise mon travail de maturité, je me suis informé à propos de simulations de fluides basées sur des système de particules plus en profondeur. Il se trouve qu'il existe un type de simulations de fluides hybrides qui utilisent à la fois des particules et des concepts s'apparentant plutôt à des fluides. Ce sont les "SPH" pour "Smoothed particle hydrodynamics". Il aurait été souhaitable de faire une simulation les utilisant dans le cadre de ce travail. Cependant la contrainte du temps m'en a empêché, ou plutôt ma mauvaise gestion du temps qui a fait que j'ai du bâcler mon travail en deux semaines. Même si ces dernières ont été considérablement intensives en travail, comme je l'explique dans l'annexe C.

Le but de ce document est d'étayer et d'expliquer les connaissances apprises pendant la réalisation de mon TM de sorte à ce qu'un élève du gymnase, puisse en le lisant, comprendre au mieux comment faire une simulation de Particules et une simulations de fluides "Eulérienne". La première section est donc dédiée aux dérivées partielles et discrétilisations afin que le lecteur, après l'avoir lue, ait les outils nécessaires pour comprendre comment résoudre les différentes équations qui régiront nos modèles, en particulier, les équation de Navier-Stokes.

1. Pour obtenir les lien, veuillez consulter la version électronique de ce document présente sur le répertoire github <https://github.com/Brinax/Fluid-Sim-TM>

En plus de cela, dans ce travail, je vais essayer de reproduire des phénomènes physiques dans les deux simulations et comparer brièvement les résultats.

Pour finir je tient à préciser que ce travail de maturité été rédigé avec la grande aide de ChatGPT. Le texte originel, avant corrections, a cependant été rédigé, selon mes estimations, au moins à 75% par moi-même.

2 Dérivées partielles et discrétisations

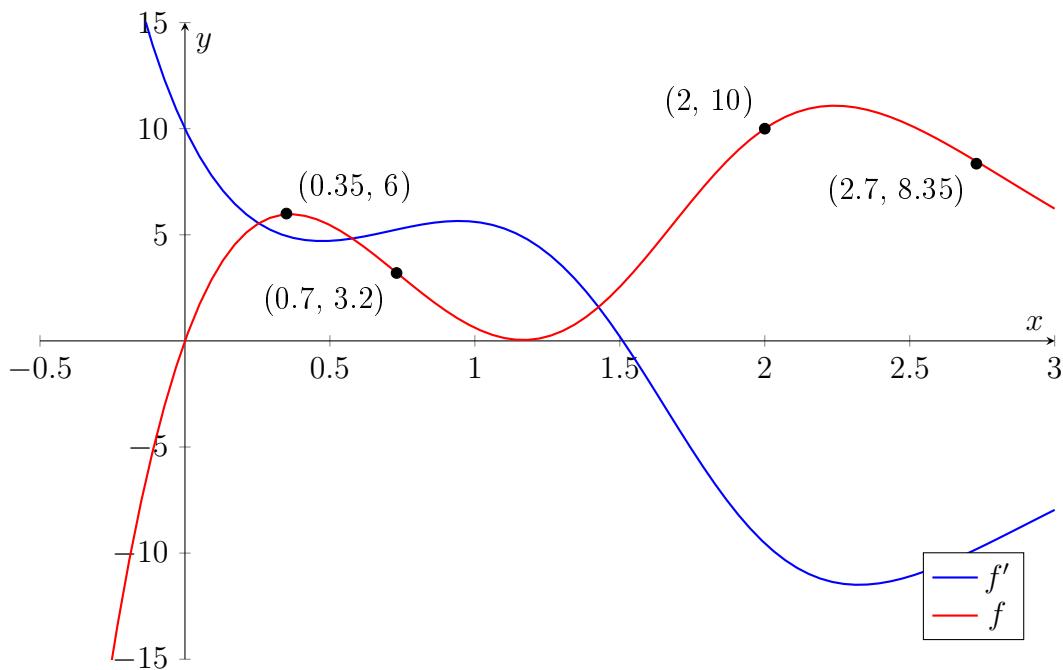
Nous allons commencer en expliquant ce que sont les dérivées partielles. Pour cela rappelons-nous de ce que sont des dérivées...

2.1 Dérivées

Soit $f : \mathbb{R} \rightarrow \mathbb{R}$ une fonction continue

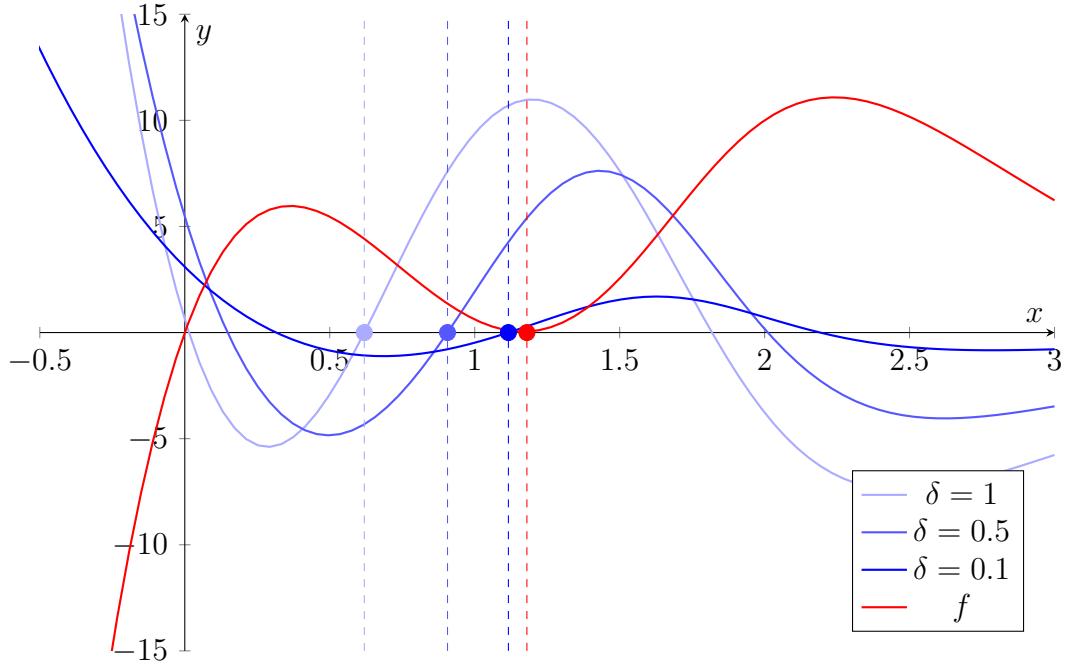
$$x \mapsto f(x)$$

Alors sa dérivée souvent notée f' cherche à décrire l'augmentation où la diminution de la valeur de $f(x)$ en fonction de x . Par exemple si $f(0) = 0$ et $f(2) = 10$ nous dirons que la valeur de x augmente dans l'intervalle $[0, 2]$. Pour représenter cette augmentation nous allons définir $f'(x)$ comme étant $f(x+2) - f(x)$ ainsi, si la fonction "augmente" dans l'intervalle $[x, x+2]$ on aura $f(x) < f(x+2)$ et donc $f'(x) > 0$; notre pseudo-dérivée représentera donc l'augmentation sous la forme d'une valeur positive. À l'inverse si la valeur de f diminue dans l'intervalle $[x, x+2]$ alors f' prendra une valeur négative. Jusque là pas de souci, imaginons donc que le graphe de f aie la forme ci-dessous et observons le comportement de f' :



Dans la partie gauche du graphe jusqu'à $x \approx 0.35$ notre pseudo-dérivée est positive et décrit en effet "l'augmentation" de la valeur de f . Cependant celle-ci se termine rapidement. Autour de $x \approx 0.7$ la valeur de f est clairement en chute et pourtant notre fonction f' est toujours positive! Cela est dû au fait que la fonction f' qu'on a définie décrit l'augmentation ou la

diminution de f en fonction de sa valeur à $x + 2$ et, en effet, même si, à $x \approx 0.7$, f est en "diminution", sa valeur en $x = 2.7$ est plus élevée que celle en $x = 0.7$. Ce que nous aimerais faire c'est capturer l'augmentation ou la diminution de f dans l'intervalle le plus petit possible. Observons donc la fonction f' pour des intervalles δ de plus en plus petits :



Vous pouvez voir dans ce graphique, qu'à mesure que δ diminue, le graphe de f' (en bleu) se rapproche d'une correcte estimation de l'augmentation/diminution de la valeur de f . Nous pouvons, par exemple, voir que le zéro de la fonction f' , s'approche, graduellement, du moment où la fonction f atteint un minimum local², aux alentours de $x \approx 1.2$, qui marque un changement de "diminution" à "augmentation" et qui fait logiquement passer notre dérivée de valeurs négatives à des valeurs positives. (Attention, dans cet exemple f passe par 0 en même temps que la dérivée, mais c'est une pure coïncidence!) Retenons de cet exemple que plus δ est petit plus le moment de bascule entre l'augmentation et la diminution de la fonction f est représenté de façon précise par les croisements de f' avec l'axe x . De plus, nous pouvons voyons, dans le graphique, que plus δ est proche de 0 plus les valeurs de f' sont proches de 0, regardez, par exemple, le graphe de f' lorsque $\delta = 0.1$. De la gauche, elle arrive considérablement plus bas que pour n'importe quel autre δ pour lequel on a affiché f' et, en x positif, ses valeurs ne sortent pas de l'intervalle $[-5, 5]$ là où toutes les autres oui. Cela est dû assez intuitivement au fait que plus on regarde l'augmentation ou la diminution de f dans un intervalle petit moins, en général, cette dernière réalise un changement élevé de ses valeurs. Pour dire cela en d'autres mots, si on prends $f(x)$ et $f(x + 0.0000001)$, en général, la différence va être très petite et on aura donc $f(x) - f(x + \delta)$ très petit. Mathématiquement on peut exprimer une idée similaire en utilisant la notion de limite :

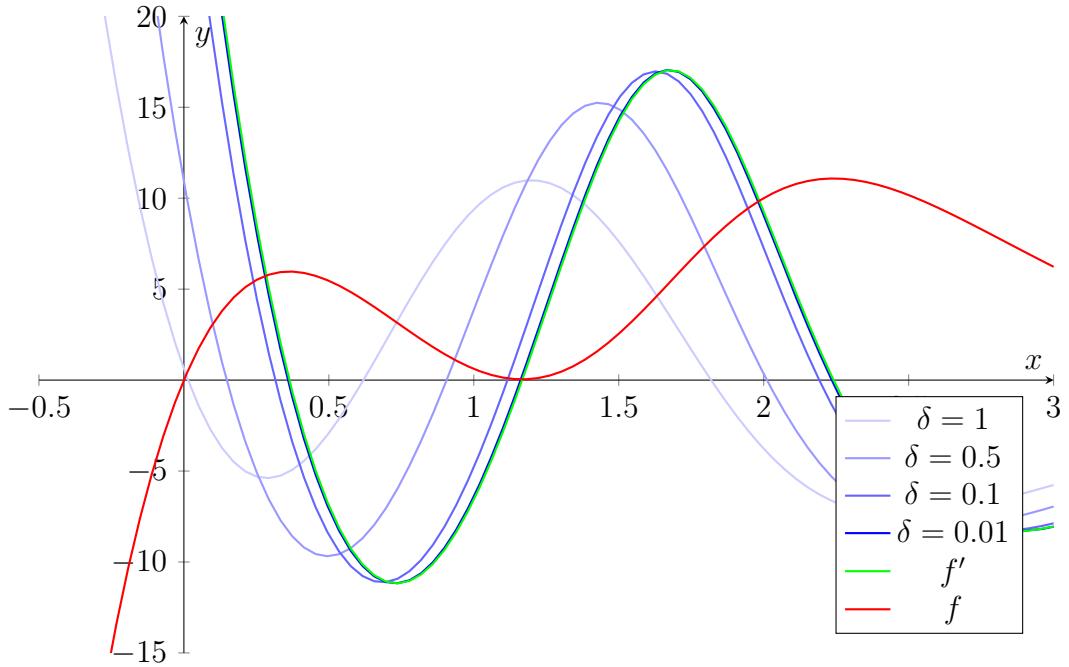
$$\lim_{x \rightarrow 0} f(c + x) = f(c)$$

Cette assertion, qui est vraie pour tout c appartenant au domaine de définition d'une fonction continue, nous dit que $\forall \epsilon > 0 \exists \Delta > 0$ tel que $|\delta| < \Delta \implies |f(c + \delta) - f(c)| < \epsilon$

Autrement dit en un point c quelconque dans le domaine de définition, un Δ assez petit donnera un intervalle où la différence entre $f(c)$ et $f(c + \delta)$ est arbitrairement petite si $|\delta|$ est inférieur à Δ . C'est justement ce qui fait que notre pseudo-dérivée tende vers 0 ! Pour contre-

2. Un minimum local d'une fonction f est un point $(x, f(x))$ tel qu'il existe un intervalle $[x - \Delta, x + \Delta]$ avec $\Delta > 0$ tel que quelque soit $x_0 \in [x - \Delta, x + \Delta]$ on a l'inégalité $f(x) \leq f(x_0)$. Autrement dit, c'est un point autour duquel $f(x)$ augmente dans toutes les directions. Dans le graphique nous voyons que cela implique qu'en avant du point x la fonction soit en diminution et qu'après celui-ci la fonction f augmente.

carrer ce problème modifions notre fonction f' en la multipliant par un coefficient dépendant de δ . Il faut que plus δ soit petit plus le coefficient soit grand. $1/\delta$ est le candidat idéal. Notre fonction devient donc $f'(x) = \frac{f(x+\delta)-f(x)}{\delta}$.



Maintenant, les croisements de f' avec l'axe x continuent d'approximer les points de bascule augmentation/diminution de la fonction f et en plus de cela notre "dérivée" a arrêté de "s'aplatir" elle semble même tendre vers une certaine fonction...

$$f'(x) = \lim_{\delta \rightarrow 0} \frac{f(x + \delta) - f(x)}{\delta} \quad (1)$$

Voilà la définition de la dérivée telle qu'elle peut être trouvée dans certains livres de maths ! Elle est simplement la limite de notre fonction lorsque δ tends vers 0. Cette définition à un tas d'aspects intéressants comme par exemple le fait qu'elle donne la pente de la tangente à f en $x = c$ mais nous n'allons pas rentrer dans les détails ici, au lieu de cela imaginons ce que pourrait représenter f ...

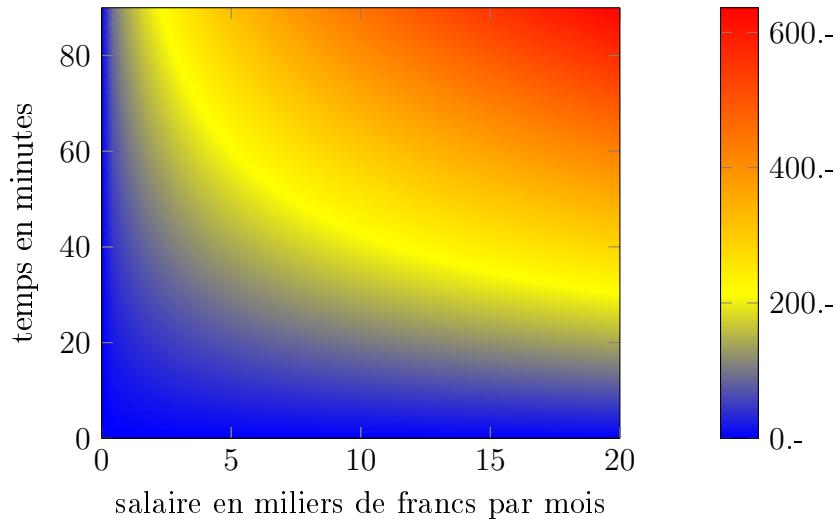
2.2 Fonctions de plusieurs variables et dérivées partielles

Dans un contexte de simulations physiques on a souvent besoin d'avoir la valeur d'une fonction en ...fonction de plusieurs variables. Pour comprendre la trivialité de cette notion et ne pas l'associer à des notions physiques qui pourraient paraître compliquées, considérons un cas imaginaire.

Imaginons que nous voulions avoir un fonction f décrivant la somme que dépensent, en moyenne, des personnes à chaque fois qu'elles vont dans un certain commerce. Si toutes les personnes dépensaient le même montant m à chaque fois, comme par exemple dans un buffet à volonté, on pourrait définir $f \equiv m$. Cependant, supposons que nous voulions décrire f d'un supermarché, et que le montant moyen que dépensent les gens dépende du temps t en minutes qu'elles passent dans le magasin. Alors on pourrait définir $f(t) = 5t$ en supposant que les clients dépensent, en moyenne, 5.- par minute à chaque fois qu'il font des courses dans ce supermarché. Maintenant, supposons que le montant qu'ils dépensent dépende également du salaire mensuel s du client. Alors la fonction pourrait devenir par exemple $f(s, t) = \sqrt{s/10} \cdot 5t$ ce qui serait une fonction de plusieurs variables.

Maintenant, nous allons essayer de comprendre ce que cette fonction nous apprends sur les dépenses de nos clients imaginaires...

Dépenses des clients calculés selon la fonction f



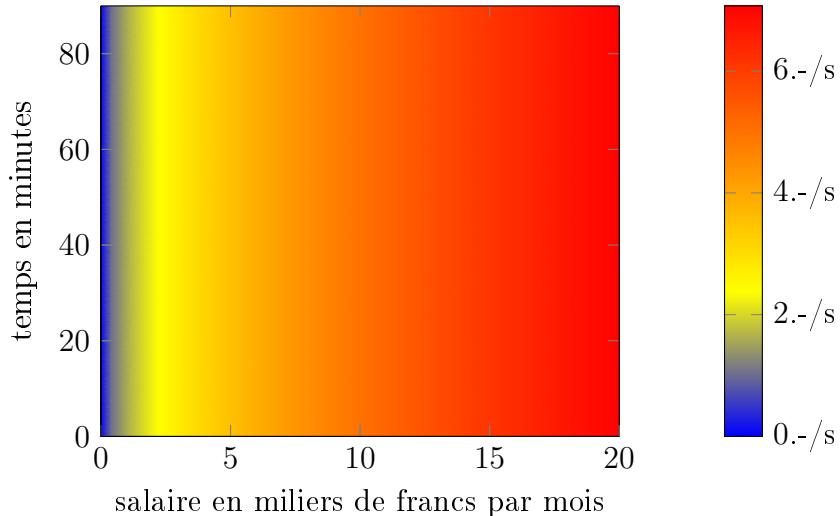
Puisque c'est une fonction de deux variables continues on a besoin d'une troisième dimension pour représenter le graphe de f . Ici la troisième dimension est la dimension "colorée" qui représente les valeurs de f en fonction des coordonnées (s, t) . Sur ce graphique, la ligne jaune indique environ le seuil autour duquel la barre des 200.— de dépenses par personne est atteint. Sur la partie haute du graphique on peut voir que le seuil est atteint pour une personne ayant un salaire d'environ 2'500.— par mois si elle passe au moins 80 minutes dans le supermarché. De l'autre côté, nous voyons que les personnes ayant un salaire de 20'000.— par mois dépassent le seuil en seulement trente minutes. Les parties bleues sur le bord gauche et le bord bas indiquent que les clients n'ayant pas de salaire ne dépensent pas et que les clients qui ne passent que très peu de temps dans le magasin non plus. Jusque là rien de bien compliqué.

Supposons que nous voulons faire un stand au milieu du supermarché pour divertir les clients et ainsi les faire rester plus longtemps. Nous voulons savoir, pour un certain client, combien cela augmenterait ses dépenses dans le magasin en considérant que la fonction décrivant ses dépenses reste la même. Il nous faudrait donc une fonction décrivant la variation de f selon le temps passé dans le magasin. C'est la dérivée partielle de f par rapport au temps, que l'on note $\frac{\partial f}{\partial t}$. Sa définition est en fait très similaire à celle de la dérivée :

$$\frac{\partial f}{\partial t}(s, t) = \lim_{\Delta_t \rightarrow 0} \frac{f(s, t + \Delta_t) - f(s, t)}{\Delta_t} \quad (2)$$

Elle décrit l'augmentation de la valeur de f lorsque on augmente le temps t . Exactement comme pour une dérivée normale, où on prendrait $f(s, t + \Delta t) - f(s, t)$, qui ferait correspondre le signe de la fonction à une augmentation/diminution, que l'on divisera par Δ et dont on observerait le comportement lorsque Δ tends vers 0. Bien que ce dernier soit infinitésimal, il laisse une marque indélébile à $\frac{\partial f}{\partial t}$ au niveau des unités. En effet si Δ_t représente un temps en minutes, aussi petit soit-il, l'unité de f sera du "par minute".

Dérivée partielle de f par rapport à la variable t



Voilà le graphe de $\frac{\partial f}{\partial t}$. Nous ne sommes donc pas surpris de voir que, désormais, les couleurs représentent une somme d'argent par seconde. La dérivée partielle par rapport au temps exprime donc une vitesse de dépense. Plus généralement, une dérivée par rapport à une variable temporelle exprime, presque tout le temps si ce n'est toujours, une certaine forme de vitesse. En regardant le graphe, on voit très clairement une constance verticale de la couleur, ce qui indique que la vitesse ne dépend pas du temps que les clients passent dans le magasin. En revanche plus les clients gagnent bien leur vie plus ils dépensent vite. Mais l'augmentation de la vitesse de dépense est notable uniquement entre les salaires allant de 0 à 10 mille francs par mois. Cela vient du fait que la variable s soit sous la racine dans la fonction f . On peut donc conclure que, pour notre stand de divertissement, le public cible sont les plus de 10.

2.3 Opérateur nabla et opérateur laplacien

Les dérivées partielles c'est très joli mais lorsqu'on atterrit sur les équations de Navier Stokes on voit ça :

$$\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla) \vec{u} = -\nabla p + \nu \Delta \vec{u} + \vec{f}$$

Et on se dit que ces deux là on sûrement confondu une équation avec un exercice de trigonométrie vu les nombreux triangles dessinés dessus... Sauf que ce ne sont pas des triangles à proprement parler, Δ (delta) est l'opérateur laplacien et ∇ (nabla) est l'opérateur nabla. Ce dernier est utilisé pour décrire rien de moins que le gradient, la divergence, le rotationnel et, en fait, aussi Δ , le laplacien.[23] Voyons donc ce qu'il a sous le capot...

$$\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right) \quad (3)$$

Nabla est un vecteur dont les coefficients sont des dérivées partielles par rapport aux coordonnées x , y et z . En deux dimensions la définition est simplement la même mais sans la troisième coordonnée :

$$\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right) \quad (4)$$

Remarquons qu'il manque les fonctions au numérateur des dérivées partielles. En fait ∇ est ce qu'on appelle dans le jargon "un opérateur agissant sur des fonctions différentiables [...]" construit à partir des dérivées partielles [de la fonction]"[25] De cela il faut retenir que ∇ est utilisée sur une fonction f pour en obtenir une autre en fonction de ses dérivées partielles.

Selon l'opération qu'on utilise entre ∇ et f on obtient différents résultats. Mais avant de les énumérer, il faut impérativement savoir faire la différence entre une fonction scalaire et une fonction vectorielle. Car elles interagissent de façon différente avec ∇ . Une fonction scalaire f prend des valeurs scalaires, qui seront en général des réels $f(x) \in \mathbb{R}$, alors qu'une fonction vectorielle \vec{f} renvoie un vecteur, donc en général $\vec{f}(x) \in V$ avec V un \mathbb{R} -espace vectoriel.

Une fois cela clarifié voici la liste des opérations fondamentales que ∇ permet de représenter :

∇f donne le gradient d'une fonction scalaire f .

$\nabla \cdot \vec{f}$ donne la divergence d'une fonction vectorielle \vec{f} .

$\nabla^2 f$ donne le laplacien d'une fonction scalaire f ou vectorielle \vec{f} .

$\nabla \wedge \vec{f}$ donne le rotationnel d'une fonction vectorielle \vec{f} .

Cependant vous me direz : "c'est bien beau, mais c'est quoi le gradient ? ou la divergence ?" Voyons donc cela plus en détail...

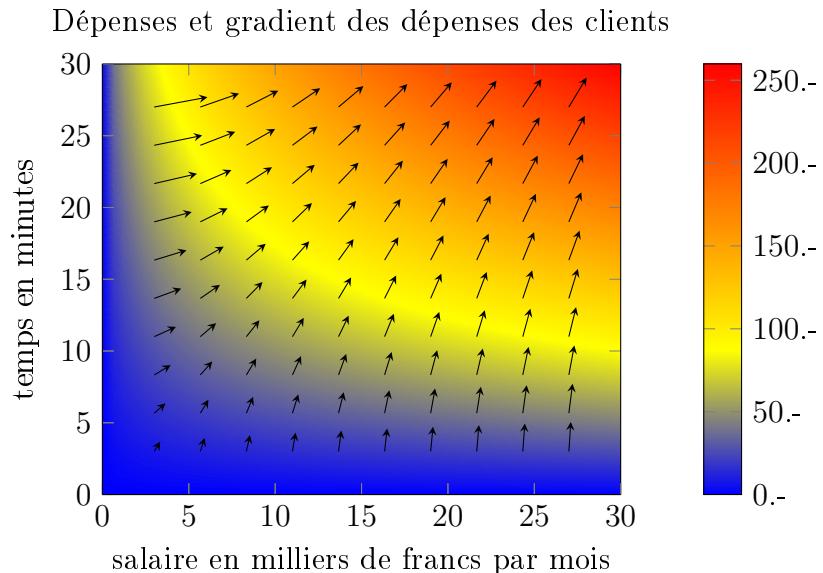
2.3.1 Gradient

Selon Wikipedia :"Le gradient d'une fonction est son taux de variation selon la position"[18]
Pour voir cela développons la forme :

$$\text{grad}(f) = \nabla f = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right) f = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \right) \quad (5)$$

Nous voyons que le gradient effectue la dérivée partielle de f par rapport à chaque axe et cela nous donne un champ vectoriel. Ainsi, pour chaque position, le gradient d'une fonction renvoie un vecteur.

Mais, c'est quoi exactement le taux de variation selon la position ? Pour mieux comprendre cela, regardons le champ vectoriel du gradient de la fonction que l'on avait faite pour la section sur les dérivées :



Nous voyons que le gradient pointe vers là où les valeurs de la fonction augmentent le plus rapidement. Pour comprendre pourquoi cela arrive reprenons (5) et considérons par exemple un point (s, t) où une petite augmentation de s provoquerait une forte élévation de $f(s, t)$ tandis qu'un changement de t ne changerait pas trop $f(s, t)$, comme il est par exemple possible de voir aux coordonnées (5, 25) du graphique. $\nabla f(s, t)$ peut être calculé à partir des dérivées partielles

en utilisant la définition de nabla en deux dimensions(5) en remplaçant x par s et y par t étant donné que ce sont les axes de notre schéma :

$$\nabla f(s, t) = \left(\frac{\partial}{\partial s}, \frac{\partial}{\partial t} \right) f(s, t) = \left(\frac{\partial f(s, t)}{\partial s}, \frac{\partial f(s, t)}{\partial t} \right) \quad (6)$$

Selon la définition d'une dérivée partielle, la coordonnée horizontale serait $\lim_{\Delta_s \rightarrow 0} \frac{f(s + \Delta_s, t) - f(s, t)}{\Delta_s}$ que nous estimons assez grande car nous avons dit qu'au point (s, t) même une augmentation Δ_s petite provoque une augmentation considérable de f . Mathématiquement on écrit cela comme $0 << f(s + \Delta_s, t) - f(s, t)$. Notre approximation de $\frac{\partial f}{\partial s}(s, t)$ donnerait donc :

$$0 << \frac{f(s + \Delta_s, t) - f(s, t)}{\Delta_s} \approx \lim_{\Delta_s \rightarrow 0} \frac{f(s + \Delta_s, t) - f(s, t)}{\Delta_s} = \frac{\partial f}{\partial s}(s, t)$$

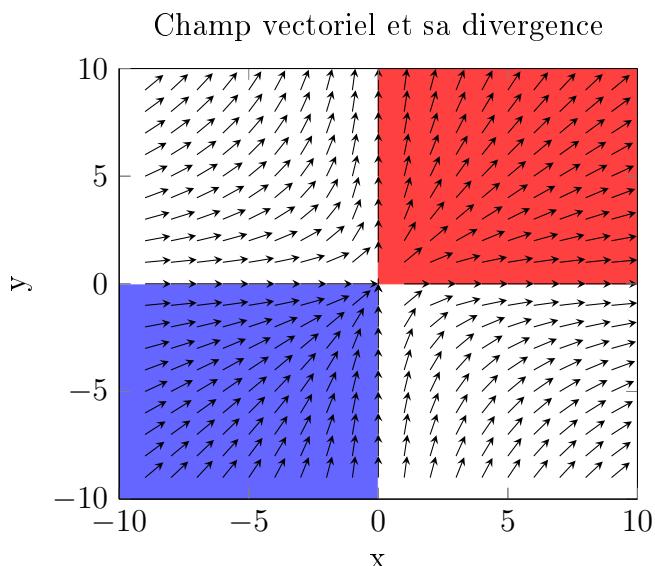
Par un raisonnement similaire nous estimerions $\frac{\partial f}{\partial t}(s, t) \approx 0$ car nous avons dit que $f(s, t)$ ne varie pas beaucoup en fonction de t .

Nous nous retrouvons donc avec un vecteur de la forme $(1, 0)$ qui pointe vers la droite, comme dans le graphe. Alors que justement, respectivement à notre point, la valeur de f augmente dans cette direction. C'est comme ça que l'on peut comprendre de façon assez intuitive pourquoi le gradient pointe toujours vers là où la valeur de f augmente.

Pour aller plus loin dans cette idée de "trouver" la direction du gradient dans des points du graphe, on pourrait aussi, selon le point sur lequel on veut connaître la direction du gradient, changer de référentiel en choisissant l'axe x dans la direction de la plus forte augmentation depuis le point voulu et l'axe y perpendiculaire à l'axe x . De cette façon on aura de nouveau une estimation ayant la même direction que le vecteur $(1, 0)$ ce qui généralise notre explication. Sauf si, en le point choisi, la valeur augmente ou diminue uniformément dans toutes les directions. Dans ce cas, il serait impossible de "trouver" la direction de la plus forte augmentation. Dans une telle situation le gradient serait en fait nul. D'ailleurs ce cas particulier nous amène à la prochaine notion...

2.3.2 Divergence

La divergence d'un champ vectoriel en un point donne une idée de la façon dont les vecteurs du champ "s'éloignent" de façon générale de ce point. En d'autres termes, elle mesure l'expansion ou la contraction du champ autour d'un point donné. Voici un exemple de graphe décrivant la divergence par des couleurs du champ vectoriel :

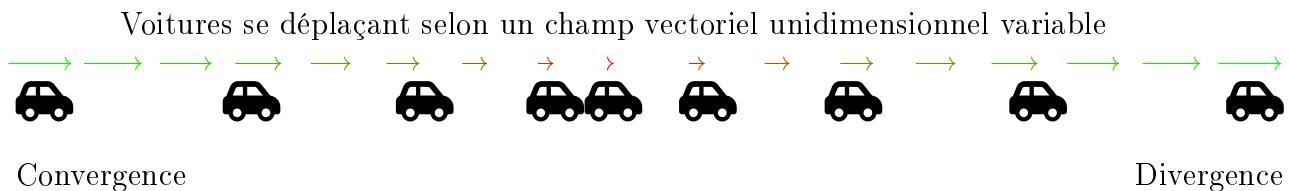


La divergence en un point donné nous indique si les vecteurs divergent du point (divergence positive) ou s'en rapprochent (divergence négative). Dans ce graphe la divergence est soit positive (rouge) soit négative (bleu) soit nulle (blanche). Nous voyons aisément que dans la partie rouge du graphe les vecteurs divergent et que dans la partie bleue ils convergent. Quand aux parties blanches, il est assez difficile, intuitivement, de savoir si ses vecteurs divergent ou convergent. Ce qu'on voit c'est que les vecteurs dans ces régions ont l'air de "tourner". La divergence est en fait nulle dans ces parties. Ce qui ne constraint pourtant pas le champ vectoriel ,dans ces régions, à être uniforme... Cela peut sembler assez contre intuitif, car on a bien l'impression qu'il y a doit y avoir certains points légèrement convergents ou divergents, mais ce n'est pas le cas. Voici donc l'équation qui nous permet de calculer la divergence :

$$\operatorname{div}(\vec{f}) = \nabla \cdot \vec{f} = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right) \cdot \begin{pmatrix} \vec{f}_x \\ \vec{f}_y \\ \vec{f}_z \end{pmatrix} = \frac{\partial \vec{f}_x}{\partial x} + \frac{\partial \vec{f}_y}{\partial y} + \frac{\partial \vec{f}_z}{\partial z} \quad (7)$$

La divergence est la somme des dérivées partielles des composantes d'une fonction vectorielle par rapport aux axes respectifs. Dans l'équation, on voit cela par exemple avec la composante x de \vec{f} notée \vec{f}_x cette dernière est différentiée par rapport à l'axe x ce qui donne $\frac{\partial \vec{f}_x}{\partial x}$. Il faut bien comprendre que le x au numérateur n'a pas le même sens que le x au dénominateur. En effet, le x au dénominateur indique la variable de \vec{f} selon laquelle on différencie \vec{f} alors que le x en index de la fonction \vec{f} , au numérateur, indique la composante x du vecteur \vec{f} . Lorsque on voudra avoir la valeur de ce terme en un point (x, y, z) , on notera $\frac{\partial \vec{f}_x(x,y,z)}{\partial x}$ et le x au dénominateur fera explicitement référence au x parmi, (x, y, z) , les variables de la fonction \vec{f} .

Maintenant que nous avons compris la notation de l'équation essayons de comprendre son fond. Pour cela considérons un cas unidimensionnel où le champ vectoriel représente la vitesse d'une file voitures :



Dans la partie gauche du schéma, on a bien l'impression que le champ vectoriel est convergent car nous voyons que les voitures, se déplaçant selon le champ vectoriel, sont des plus en plus rapprochées à mesure qu'elles avancent. Tandis que dès que la vitesse réaugmente les voitures gagnent de la distance entre elles. Cet exemple est assez intuitif et nous permet de comprendre pourquoi, en une dimension, la divergence de \vec{f} est $\frac{\partial \vec{f}_x}{\partial x}$. En effet, lorsque les vecteurs $\vec{f}_x(x)$ diminuent à mesure que x augmente, des voitures ayant pour vitesses $\vec{f}(x)$ où x est leur position, se rapprochent entre elles, donc "convergent". À l'inverse lorsque $\frac{\partial \vec{f}_x(x)}{\partial x}$ est positive les voitures s'éloignent entre elles, divergent, et la divergence est également positive.

On peut aussi voir cela d'une manière légèrement différente et considérer que la divergence indique à quel point "l'espace libre" autour d'une voiture augmente ou diminue. Cette interprétation nous permet d'étendre le concept au cas bidimensionnel et tridimensionnel $\operatorname{div}(\vec{f}) = \frac{\partial \vec{f}_x}{\partial x} + \frac{\partial \vec{f}_y}{\partial y} (+ \frac{\partial \vec{f}_z}{\partial z})$. On se donne une intuition du fait que chaque dérivée partielle dans chaque dimension, indique une vitesse de croissance ou de diminution, du volume, ou aire, disponible dans la dimension donnée autour d'une particule, et que donc l'augmentation totale d'aire/volume est simplement la somme des augmentations dans chaque dimension.

De là on comprend tout naturellement que pour qu'un fluide soit incompressible il faut que sa divergence soit nulle, car dans un fluide incompressible chaque "particule" occupe toujours une même volume. Si la divergence n'était pas nulle à certains endroits, il y aurait des particules qui gagnent du volume ou qui en perdent, le fluide serait donc "comprimé" ou "dilaté" en ces points.

2.3.3 Laplacien

Maintenant que nous avons compris la notion de gradient, qui associe un champ vectoriel à une fonction scalaire, et la notion de divergence, qui associe une fonction scalaire à un champ vectoriel. Nous pourrions donc essayer de calculer la divergence d'un gradient, non ? Ou bien le gradient de la divergence d'un gradient, ou la divergence d'un gradient de la divergence d'un gradient de la divergence d'un gradient de la divergence de... etc...

Théoriquement cela est possible, mais dans la pratique il n'y a généralement pas d'intérêt à faire la divergence/gradient d'une fonction plus de deux ou trois fois de suite. En particulier lorsqu'on fait la divergence d'un gradient cela s'appelle le laplacien d'une fonction scalaire que l'on note :

$$\Delta = \nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \quad (8)$$

Lorsqu'on le multiplie par une fonction scalaire f cela revient donc à appliquer deux fois nabla :

$$\Delta f = \nabla(\nabla f) = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right) \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \\ \frac{\partial f}{\partial z} \end{pmatrix} = \frac{\partial f}{\partial x^2} + \frac{\partial f}{\partial y^2} + \frac{\partial f}{\partial z^2} \quad (9)$$

La notation $\frac{\partial f}{\partial x^2}$ pourrait sembler déroutante mais c'est en fait simplement la deuxième dérivée partielle de f par rapport à la variable x . Une notation équivalente mais très peu élégante serait $\frac{\partial^2 f}{\partial x^2}$.

Nous pourrions de nouveau tenter de donner une explication intuitive de ce que représente le laplacien, mais cela devient considérablement plus abstrait. En fait, le plus simple c'est de comprendre cas par cas ce qu'il exprime en fonction de ce que f représente. On verra dans la section de l'interprétation des équation de Navier-Stokes qu'il est utilisé pour calculer le terme de viscosité.

Justement ce dernier mérite un léger approfondissement de la notion de Laplacien. Car dans les équation de Navier-Stokes, ce dernier n'est pas appliqué à un champ scalaire, mais bien à un champ vectoriel ; celui de la vitesse. Lorsqu'on effectue $\Delta \vec{f}$ on pourrait se dire que, le laplacien, au lieu d'être la divergence du gradient, est, dans ce cas, simplement le gradient de la divergence noté $\nabla(\nabla \vec{f})$. Cependant cette interprétation est fausse !

Le laplacien d'un champ vectoriel appelé Laplacien vectoriel est calculé sur une fonction vectorielle $\vec{f} = (\vec{f}_x, \vec{f}_y, \vec{f}_z)$ comme suit :

$$\Delta \vec{f} = (\Delta \vec{f}_x, \Delta \vec{f}_y, \Delta \vec{f}_z) \quad (10)$$

C'est donc le vecteur formé du laplacien de chaque composante de \vec{f} .

Finalement, puisque nous n'avons pas pu trouver d'interprétation intuitive au Laplacien, comme consolation, regardons ce que nous dit Wikipedia son propos :

"Intuitivement, il combine et relie la description statique d'un champ (décrit par son gradient) aux effets dynamiques (la divergence) de ce champ dans l'espace et le temps." [26]

2.3.4 Rotationnel

Avant d'en finir avec les opérateurs différentiels définis à l'aide de ∇ , on ne pouvait pas ignorer le rotationnel qui est une opération vectorielle qui s'applique à une fonction vectorielle \vec{f} dans un espace tridimensionnel. Cette opération est notée $\nabla \wedge \vec{f}$ et est donc définie comme le produit scalaire entre \vec{f} et ∇ :

$$\nabla \wedge \vec{f} = \begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ f_x & f_y & f_z \end{vmatrix} = \left(\frac{\partial f_z}{\partial y} - \frac{\partial f_y}{\partial z}, \frac{\partial f_x}{\partial z} - \frac{\partial f_z}{\partial x}, \frac{\partial f_y}{\partial x} - \frac{\partial f_x}{\partial y} \right) \quad (11)$$

Le rotationnel mesure la tendance de la fonction vectorielle à "tourner" autour d'un point dans l'espace. Plus précisément, le rotationnel donne un vecteur qui a pour direction l'axe de rotation maximal et dont la magnitude est proportionnelle à la vitesse angulaire, c'est à dire une certaine quantité de radians par seconde. En fait il est assez similaire au gradient ; là où le gradient s'oriente vers la direction d'augmentation maximale d'une fonction scalaire f , le rotationnel pointe dans la "direction de rotation" maximale d'une fonction vectorielle \vec{f} . Mais qu'est-ce qu'on entend par "direction de rotation" ? Pour comprendre cela rappelons-nous des propriétés d'une rotation en trois dimensions. Un objet en rotation a toujours un axe "autour" duquel il tourne, appelé axe de rotation. Une propriété de l'axe de rotation d'un objet est que la partie de l'objet qui est "sur l'axe" ne se déplace, en fait, pas, ou en tout cas, pas par rotation autour de l'axe choisi. Pour clarifier cela observons la rotation d'une roue :



FIGURE 4 – Roue de vélo en rotation

La flèche verticale rouge représente l'axe de rotation. C'est une flèche, et non pas une simple droite, afin de représenter le sens de la rotation de la roue (déterminé selon la règle de la main droite). Ensuite, les flèches faisant le tour du pneu représentent la trajectoire qu'aurait un "bout" de la roue, en un certain intervalle de temps, comme nous l'avons dit plus haut, la partie de la roue proche de l'axe de rotation se déplace de façon beaucoup plus faible que le reste (Petites flèches rouges). Ainsi, plus ces flèches sont proches du centre de rotation, plus elles sont courtes (vert > orange > rouge sur le schéma), ce qui indique qu'elles ont une vitesse moindre. Cependant, elles se déplacent toutes à la même vitesse angulaire, étant donné qu'à priori la roue est rigide, et donc, qu'en tournant, tous ses composants réalisent le même nombre de tours par seconde ; le contraire serait préoccupant.

Maintenant revenons à notre histoire de rotationnel, si le champ vectoriel décrit par \vec{f} représentait la vitesse de chaque point de la roue. Alors son rotationnel devrait, en chaque point,

représenter le même vecteur "de rotation", étant donné que toutes les parties de la roue ont la même vitesse angulaire et axe de rotation.

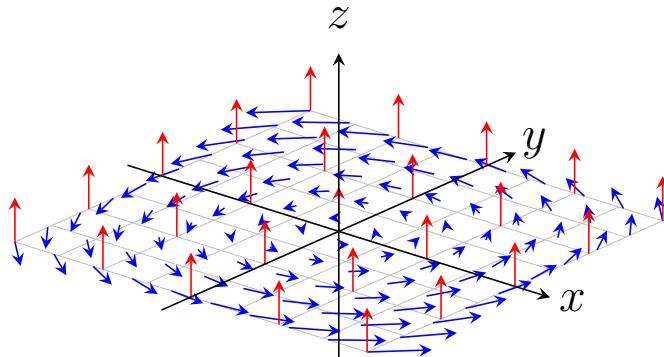


FIGURE 5 – Champ vectoriel (bleu) et son rotationnel (rouge)

Dans ce graphique le champ de vecteurs en bleu représente la vitesse de la roue à un point de l'espace donné. Nous n'affichons qu'un plan bidimensionnel car sinon le schéma deviendrait bien trop difficile à lire. Mais il faut bien imaginer que ce champ pourrait s'étendre également sur l'axe z . En rouge, c'est le champ vectoriel du rotationnel, qui, comme nous l'avons expliqué, pointe uniformément vers le haut. Cependant si notre champ vectoriel \vec{f} était tourné à 90° autour de l'axe y alors le champ de son rotationnel le serait également.

Maintenant que nous avons une intuition de ce que représente le rotationnel, essayons de comprendre intuitivement comment ∇ permet d'obtenir le champ rotationnel de \vec{f} . Pour cela considérons un cas simple dans un plan bidimensionnel où nous voyons clairement une rotation et dans lequel nous pouvons faire une estimation des différentes dérivées spatiales de \vec{f} :

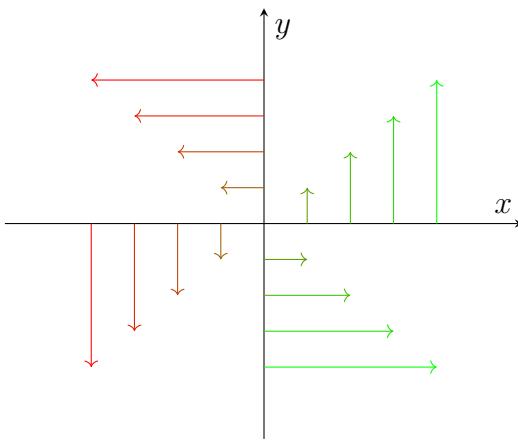


FIGURE 6 – Champ vectoriel tournant en sens antihoraire

Voici donc un champ vectoriel très simple où uniquement les vecteurs sur l'axe x et y sont représentés, cela nous permettra d'estimer le rotationnel de \vec{f} au point $(0, 0)$. Avant cela, remarquons que le champ vectoriel tourne en sens anti-horaire autour de l'axe z qui n'est pas représenté sur le schéma mais qui "sort" de la feuille. Son rotationnel à l'origine devrait donc être de la forme $(0, 0, \lambda)$ avec $\lambda > 0$ (en suivant la règle de la main droite).

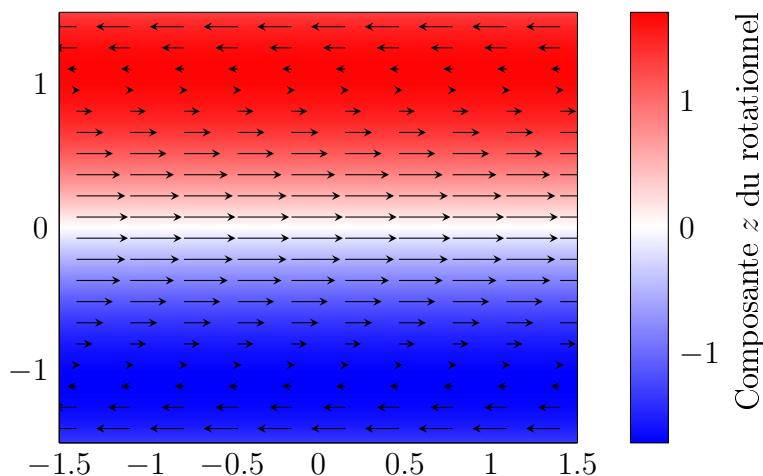
Passons à l'analyse des dérivées partielles, nous voyons que sur l'axe x les vecteurs ont la même direction que l'axe y , cela veut dire qu'ils sont de la forme $(0, \lambda)$. De plus, leur composante λ augmente clairement à mesure que x augmente, autrement dit on voit que $\frac{\partial \vec{f}_y(0,0)}{\partial x} >> 0$.

À l'inverse, sur l'axe y , les vecteurs ont la même direction que l'axe x . Ils sont donc de la forme $(\lambda, 0)$ et λ diminue à mesure que y augmente. On voit clairement cela dans le bas du graphique ($y < 0$) où le vecteur représenté pointe vers les coordonnées positives, on a donc $\lambda \gg 0$, alors que dans le haut du graphique ($y > 0$) les vecteurs pointent dans le sens opposé à l'axe x , on a donc $\lambda \ll 0$. Ainsi $\frac{\partial \vec{f}_x(0,0)}{\partial y} \ll 0$ car $f_x(0, y)$ diminue à mesure que y augmente.

En résumé, nous estimons donc $\frac{\partial f_y(0,0)}{\partial x} \gg 0$ et $\frac{\partial f_x(0,0)}{\partial y} \ll 0$ or la composante z de $\nabla \wedge \vec{f}(0,0)$ est justement $\frac{\partial f_y(0,0)}{\partial x} - \frac{\partial f_x(0,0)}{\partial y}$ qui selon nos estimations serait positive. C'est justement ce que l'on voulait montrer!... ou presque.

Puisque nous avons montré que l'on peut estimer la composante z du rotationnel comme étant positive, il ne nous reste plus qu'à montrer que les composantes x et y de $\nabla \wedge \vec{f}$ sont nulles. C'est là que l'on va rencontrer une différence majeure entre ce que notre intuition nous dit à priori et ce que le rotationnel d'un champ vectoriel va indiquer. En effet, en voyant 6, puisque les vecteurs vitesse ne sortent pas du plan XY (leur composante z est nulle), on a bien l'impression que le champ vectoriel ne pourrait tourner ni autour de l'axe x ni autour de l'axe y . En fait, nous ferions là une erreur d'interprétation qui est celle de considérer le champ vectoriel comme un "objet" rigide qui, pour tourner sur l'axe x doit forcément avoir des valeurs non-nulles en z . Car si toutes les valeurs en z sont nulles, le déplacement qui se créerait en chaque point, en considérant le champ vectoriel comme un champ de vitesses, resterait sur le plan XY ce qui semble être incompatible avec une rotation autour de x ou y .

Cependant, rappelons-nous que le rotationnel ainsi que toutes les opérations différentielles agissent de façon locale. Elles "approximent", sur un point, une valeur qui est d'habitude calculée sur un intervalle discret comme par exemple une augmentation. À cela on peut rajouter que dans un objet en rotation il n'y a pas tout l'objet qui se déplace par rotation. En effet, comme on l'a dit plus tôt, la partie d'un objet qui est exactement sur son axe de rotation ne se déplace pas par rotation autour de cet axe. On pourrait donc imaginer que chaque vecteur est le centre d'une rotation parallèle à l'axe x qu'il ne partage pas avec ses vecteurs adjacents sur le plan XY qui ne sont donc pas "obligés" d'avoir une composante en z pour suivre sa rotation. Tout cela est un peu compliqué et abstrait, mais l'essentiel est de comprendre, qu'en fait, le rotationnel de notre champ vectoriel 6 (tout comme celui de 5) pourrait avoir une composante en y et en z . Cette composante serait cependant effectivement uniquement déterminée par les vecteurs qui ne se trouvent pas sur le plan XY . Pour mieux comprendre ce phénomène on peut le reproduire en 2 dimensions :



Ici, au lieu d'avoir un plan XY où les composantes z des vecteurs sont nulles, on a des lignes horizontales où la composante y est nulle. On est donc tentés croire que le rotationnel en

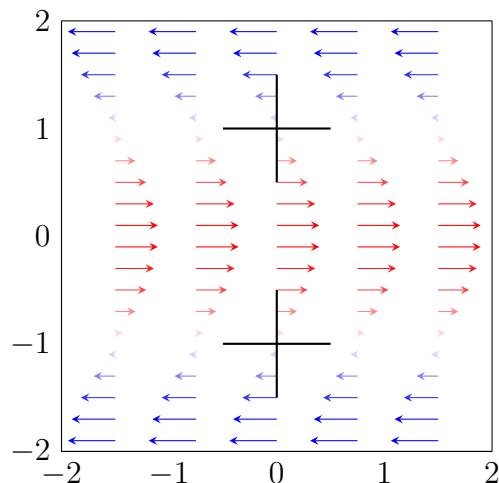
z est forcément nul et pourtant ce n'est pas le cas. En effet, la vitesse ne change pas au long de l'axe x , cependant le long de l'axe z oui. Plus spécifiquement, la composante x des vecteurs augmente jusqu'à $y = 0$ puis diminue avec $y > 0$. Mathématiquement on peut résumer cela en écrivant $\frac{\partial \vec{f}}{\partial x} = 0$, $\forall y < 0 \frac{\partial \vec{f}_x}{\partial y}(x, y) > 0$ et $\forall y > 0 \frac{\partial \vec{f}_x}{\partial y}(x, y) < 0$. Calculons donc la composante z du rotationnel :

$$(\nabla \wedge \vec{f})_z = \frac{\partial \vec{f}_y}{\partial x} - \frac{\partial \vec{f}_x}{\partial y} = 0 - \frac{\partial \vec{f}_x}{\partial y} = -\frac{\partial \vec{f}_x}{\partial y} \quad (12)$$

Selon nos estimations, puisque c'est l'opposé de la dérivée de la composante x de \vec{f} par rapport à z , elle sera négative (au lieu de positive) avec $y < 0$ et positive (au lieu de négative) sur $y > 0$ et c'est justement ce que l'on voit sur le schéma.

Notre champ vectoriel à donc un rotationnel non-nul par rapport à l'axe z sans que les vecteurs aient de composante y ! Nous pouvons également remarquer, comme on l'avait dit plus tôt pour le cas bidimensionnel, que le rotationnel de chaque "ligne" horizontale dépend uniquement des vecteurs en dehors de cette ligne. Car dans la formule du rotationnel on a $\frac{\partial \vec{f}_y}{\partial x} = 0$ ce qui laisse la détermination de la valeur du rotationnel en z à $-\frac{\partial \vec{f}_x}{\partial y}$ qui est une dérivée partielle par rapport à l'axe y et non pas x .

À ce stade vous me direz probablement : "Alors, j'ai très bien compris que le rotationnel donne une rotation selon certains axes même dans des cas où on ne dirait pas qu'il y en ait. Mais alors faut-il abandonner l'idée du rotationnel décrivant des rotations ?" Et je vous répondrais que non, il y a, en fait, une technique pratiquement infaillible qui vous permettra d'estimer les composantes du rotationnel d'un champ vectoriel. Cette technique assez connue, que j'appelle la "technique du moulin à eau", est très simple et consiste à estimer dans quelle direction tournerait un très petit moulin à eau positionné à un point donné et dans l'axe de rotation voulu, en considérant que le champ vectoriel décrit l'écoulement d'un fluide. S'il tournerait en sens anti-horaire (vu depuis les coordonnées positives de l'axe voulu) alors la composante serait positive, s'il tournerait en sens horaire la composante serait négative et s'il ne tournerait simplement pas, la composante serait nulle. Illustrons cela dans notre exemple :



Ici, les couleurs indiquent la norme des vecteurs. Nous voyons clairement que le "moulin" du haut devrait tourner en sens anti-horaire, étant donné que sa pale du haut est poussée sur la gauche et celle du bas par la droite, ce qui est le signe d'une valeur positive de la composante z du rotationnel, tandis que le "moulin" en bas devrait tourner en sens horaire. Cela représente donc bien les valeurs du rotationnel.

Si on avait un champ vectoriel tridimensionnel, il faudrait répéter l'opération sur chaque axe

pour se donner une idée du rotationnel.

Pour finir, rappelons-nous donc que le rotationnel d'un champ vectoriel peut être différent de zéro même si le champ vectoriel n'a pas l'air de "tourner" vraiment, cela va être notamment important lors de l'interprétation des convections de Kelvin-Helmholtz.

2.4 Résolution d'équations différentielles

Nous savons maintenant à peu près reconnaître les différents termes des équations de Navier-Stokes, cependant maintenant vient la question de la résolution de ces dernières. Comment utiliser les équations de Navier-Stokes pour calculer le champ vectoriel de la vitesse en fonction du temps ? Et bien nous allons voir cela en détail dans la section 4.2.2, mais pour pouvoir faire cela, il faut d'abord avoir quelques bases en approximation de dérivées et en résolutions d'équations différentielles. Je vais donc ici rapidement vous présenter différentes approximations qui découlent de la définition d'une dérivée. Mais qui peuvent également être obtenues en manipulant les développements de Taylor d'une fonction deux fois dérivable f aux coordonnées $x + \delta$ et $x - \delta$. Passer par le développement de Taylor à l'avantage de fournir des informations concernant la précision de l'approximation.

Différence avant et arrière Lorsque on fait une approximation par différence avant, on utilise simplement la définition de la dérivée qui est $\lim_{\delta \rightarrow 0} \frac{f(x+\delta) - f(x)}{\delta}$ en choisissant un petit $\delta > 0$ ce qui nous donne $\frac{f(x+\delta) - f(x)}{\delta}$. Elle s'appelle différence "avant" car elle utilise des valeurs de la fonction qui sont "plus en avant" ($x + \delta > x$). À l'inverse, lorsqu'on choisit un $\delta < 0$, on parle de différence arrière. Cependant, au lieu de choisir un $\delta < 0$ on choisit généralement δ positif et on insère dans la définition de dérivée $-\delta$ ce qui donne $\frac{f(x-\delta) - f(x)}{-\delta} = \frac{f(x) - f(x-\delta)}{\delta}$. La notion de différence avant et arrière est très importante car selon les cas de figure $f(x + \delta)$ pourrait être inconnue alors que $f(x - \delta)$ pourrait être connue. Par exemple si x est une variable temporelle et qu'on cherche à calculer la valeur future $f(x + \delta)$.

Différence centrale Un des désavantages des différences avant et arrière est qu'elles introduisent une asymétrie. Alors pour résoudre le problème, la différence centrale a été inventée qui est simplement la moyenne des deux valeurs (avec la formule $M = \frac{A+B}{2}$) qui donne donc :

$$\frac{\frac{f(x+\delta) - f(x)}{\delta} + \frac{f(x) - f(x-\delta)}{\delta}}{2} = \frac{f(x + \delta) - f(x) + f(x) - f(x - \delta)}{2\delta} = \frac{f(x + \delta) - f(x - \delta)}{2\delta} \quad (13)$$

Cette formule, en plus d'être symétrique et de nous enlever la question du choix entre différence avant et arrière, a pour avantage de tendre plus vite vers la vraie valeur de f' par rapport aux deux premiers types de différences. On parle d'approximation d'ordre 2 au lieu d'ordre 1.

Différence d'ordre 2 centrale Ici pour "différence d'ordre 2" on entend une approximation de la deuxième dérivée f'' de f . Elle peut être calculée en faisant une différence centrale sur f' qui donne $f'(x) \approx \frac{f'(x+\delta) - f'(x-\delta)}{2\delta}$. Les valeurs de f' en $x + \delta$ et $x - \delta$ peuvent s'approximer respectivement par une différence arrière et une différence avant ce qui nous donne :

$$f''(x) \approx \frac{f'(x + \delta) - f'(x - \delta)}{2\delta} \quad (14)$$

$$f'(x + \delta) \approx \frac{f(x + \delta) - f(x + \delta - \delta)}{\delta} = \frac{f(x + \delta) - f(x)}{\delta} \quad (15)$$

$$f'(x - \delta) \approx \frac{f(x - \delta + \delta) - f(x - \delta)}{\delta} = \frac{f(x) - f(x - \delta)}{\delta} \quad (16)$$

En remplaçant les valeurs approximées des dérivées dans la première équation, on obtient :

$$f''(x) \approx \frac{\frac{f(x+\delta)-f(x)}{\delta} - \frac{f(x)-f(x-\delta)}{\delta}}{2\delta} = \frac{f(x+\delta) - f(x) - f(x) + f(x-\delta)}{2\delta^2} \quad (17)$$

Ainsi on devrait avoir :

$$f''(x) \approx \frac{f(x+\delta) - 2f(x) + f(x-\delta)}{2\delta^2} \quad (18)$$

Cela serait bien pratique... sauf que cette formule est bonne à jeter car son erreur³ est trop grande. C'est ici qu'il est nécessaire d'observer les termes dans la formule de Taylor pour faire les bonnes manipulations, mais à nouveau, je vais laisser le lecteur faire les calculs avec Taylor de son côté, car la complexité de cette dernière va au delà de ce que je veux admettre dans cette explication. Contentons-nous de remarquer que lorsqu'on fait les différences arrières et avant de f' l'erreur est d'ordre 1. Cela veut dire en quelque sorte qu'elle diminue comme $\delta^1 = \delta$ lorsque δ tends vers 0. Cependant, dans la formule 14, les dérivées de f' sont divisées par 2δ . Les erreurs de ces dernières, au lieu de rester d'ordre 1, sont amplifiées et deviennent de la forme $\delta/\delta = 1$, étant donné qu'elles sont divisées par δ , et évidemment 1 ne tends pas vers 0 lorsque δ tends vers 0.

Pour résoudre ce problème, il suffit de faire les différences centrales de $f'(x+\delta)$ et de $f'(x-\delta)$, qui sont cette fois des approximations d'ordre 2, et de les injecter dans la formule (14). Je vous épargne les calculs mais cela donne :

$$\frac{f(x+2\delta) - 2f(x) + f(x-2\delta)}{4\delta^2} \quad (19)$$

Qui peut se réécrire en remplaçant δ par $\frac{\delta}{2}$:

$$\frac{f(x+\delta) - 2f(x) + f(x-\delta)}{\delta^2} \quad (20)$$

Et mon petit doigt (aka Taylor) me dit que c'est une approximation d'ordre 2.

De tout cela, en plus de la formule, retenez que lorsque l'on fait des approximations, il est important de limiter l'erreur de l'approximation.

Maintenant que nous avons décrit différentes approximations de dérivées, il faut encore parler de deux méthodes de résolution d'équations différentielles. La première, et la plus simple, est la méthode explicite, tandis que la deuxième, plus difficile à calculer mais souvent plus précise, est la méthode implicite. Utilisons les deux méthodes sur un cas simple décrivant une quantité de prédateurs (loups) et de proies (moutons) qui interagissent entre eux au fil du temps. Nous considérons donc une fonction $L(t)$ décrivant le nombre de Loups en fonction du temps t en années et une autre fonction $M(t)$ décrivant le nombre de moutons. Voici les équations différentielles qui les associent[15] :

$$\frac{\partial M}{\partial t} = M(t)\left(2 - \frac{1}{2}L(t)\right) \quad (21)$$

$$\frac{\partial L}{\partial t} = L(t)\left(\frac{1}{4}M(t) - 1\right) \quad (22)$$

L'augmentation de la population des moutons, à chaque instant, est égale à sa population par 2, ce qui représente le taux de reproduction des moutons entre eux, moins sa population multipliée

3. L'erreur de l'approximation d'une fonction est la différence entre la valeur réelle et la valeur approximée de la fonction.

par la moitié de celle des loups divisée par deux, ce qui représente le fait que les moutons sont mangés par les loups. De l'autre côté, la population des loups augmente comme sa population multipliée par le quart de celle des moutons, ce qui représente le taux de reproduction des loups, qui est possible s'il arrivent à se nourrir de moutons, moins une fois sa population, ce qui représente le taux de mortalité des loups.

Maintenant si nous voulions discréteriser cette équation, il nous faudrait une population initiale de loups et de moutons. En effet, les équations différentielles décrivent uniquement le taux de changement des différentes fonctions, mais n'impliquent généralement pas des valeurs spécifiques de celles-ci. Considérons donc $M(0) = 40$ et $L(0) = 2$. Ce qui nous permettra d'approximer le nombre de loups l'année suivante ($t = 1$).

Méthode explicite Dans la méthode explicite on va considérer les équations au temps $t = 0$ et on va utiliser des différences avant pour approximer les valeurs de $M'(t) = \frac{\partial M(t)}{\partial t}$ et de $L'(t) = \frac{\partial L(t)}{\partial t}$:

$$M'(t) = \frac{M(t+1) - M(t)}{1} = M(t)(2 - \frac{1}{2}L(t)) \quad (23)$$

$$L'(t) = \frac{L(t+1) - L(t)}{1} = L(t)(\frac{1}{4}M(t) - 1) \quad (24)$$

En isolant $M(t+1)$ et $L(t+1)$ on a :

$$M(t+1) = M(t)(2 - \frac{1}{2}L(t)) + M(t) \quad (25)$$

$$L(t+1) = L(t)(\frac{1}{4}M(t) - 1) + L(t) \quad (26)$$

Pour $t = 0$ on a donc :

$$M(1) = 40(2 - 1) + 40 = 80 \quad (27)$$

$$L(1) = 2(10 - 1) + 2 = 20 \quad (28)$$

Nous avons réussi à prédire le nombre de loups et de moutons l'année suivante, il y aura 80 moutons et 20 loups. Remarquons que c'était très facile.

Méthode implicite Pour la méthode implicite nous utilisons des différences arrières au lieu de différences avant et nous allons écrire les équations pour $t = 1$:

$$M'(t) = \frac{M(t) - M(t-1)}{1} = M(t)(2 - \frac{1}{2}L(t)) \quad (29)$$

$$L'(t) = \frac{L(t) - L(t-1)}{1} = L(t)(\frac{1}{4}M(t) - 1) \quad (30)$$

Le problème est que ces deux équations ne décrivent pas des droites, qui se croiseraient en un seul point qui déterminerait les valeurs de $M(t)$ et de $L(t)$, mais, bel et bien, des paraboles :

$$0 = M(t) - \frac{1}{2}M(t)L(t) + M(t-1) \quad (31)$$

$$0 = \frac{1}{4}L(t)M(t) - 2L(t) + L(t-1) \quad (32)$$

Elle se croisent en deux points $(-43.37, 0.16)$ et $(7.38, 12.84)$, en considérant $M(t) = x$ et $L(t) = y$. Puisque à priori notre troupeau de moutons ne peut pas avoir un nombre négatif

d'individus le point que l'on va garder sera celui où $M(1) = 7.38$ et $L(1) = 12.8$. Remarquons que cette fois le résultat à été considérablement plus difficile à calculer, car on a du trouver l'intersection de deux paraboles et ensuite "choisir" une des deux intersections selon les pertinences de celles-ci.

Nous avons rapidement vu que la méthode explicite est plus simple que la méthode implicite, mais comment savoir si la fin justifie les moyens ? Comment faire pour encore mieux approximer les valeurs ? Et bien pour la deuxième question, il suffit de calculer par intervalles de temps plus petits qu'un an, par exemple 0.01 an, et de répéter la méthode choisie (explicite/implicite) cent fois pour obtenir une meilleure estimation de $M(1)$ et de $L(1)$. Dans notre exemple, les vraies valeurs sont $M(1) \approx 0.022$ et $L(1) \approx 16.57\dots$ Il faut donc croire, qu'au bout d'un an, il y aura 16.57 loups pour seulement 0.022 moutons...

À la place de constater à quel point les équations différentielles choisies sont peu réalistes concentrons nous sur la précisions des deux méthodes, en un seul "pas" de 1 elle sont toutes deux assez imprécises ; l'explicite estimait $M(1) = 80$ et $L(1) = 20$ tandis que l'implicite estimait $M(1) = 7.38$ et $L(1) = 12$ des valeurs proches de la réalité sont de 0.34 moutons et 16.27 loups. La méthode explicite a donc légèrement mieux estimé les loups alors que la méthode implicite est clairement plus proche sur l'estimation des moutons mais nous ne pouvons pas vraiment départager...

En résumé, si on note $C(t)$ les variables que l'on connaît au temps t alors la méthode explicite utilise une certaine fonction F pour estimer $C(t + \delta)$ à partie de $C(t)$ ce qui donne $C(t + \delta) = F(C(t))$, alors que la méthode implicite utilise une F pour laquelle il faut résoudre $F(C(t), C(t + \delta)) = 0$ pour trouver les valeurs de $C(t + \delta)$ [17]

En pratique, c'est en général la méthode explicite qui est utilisée, cependant dans certains cas elle diverge même avec des δ très petits ce qui augmente considérablement le temps de calcul. Dans ces situations utiliser la méthode implicite peut être une solution.

Je termine en concluant que discréteriser des équations n'est vraiment pas une mince affaire, il faut faire attentions à différent paramètre et les équations deviennent rapidement très complexes. C'est pour cela que je n'ai pas moi-même résolu entièrement les équations de Navier-Stokes ; je me suis aidé de sources externes.

3 Méthodologie

La méthode que j'ai suivi dans ce travail à été de programmer deux simulations. La première est basée sur des particules et la deuxième sur une description Eulérienne. Je me suis ensuite informé sur différents phénomènes physiques qui se produisent dans des fluides et ai tenté de les reproduire avec des conditions analogues dans les deux simulations. J'ai finalement comparé les différents résultats pour voir si les phénomènes apparaissaient. Le tout a été fait avec des programmes pythons décrits dans l'annexe A.

De façon plus concrète, j'ai cherché les différentes équations physiques correspondant au modèle choisi sur internet (par ex. formules de collision élastique pour la simulation de particules). Ensuite, j'ai pu écrire un programme qui calcule, en utilisant les équations physiques, l'évolution du système à partir de valeurs initiales. Une fois ces étapes faites pour la simulation de particules et pour la simulation de fluides, j'ai pu définir des conditions et des contraintes similaires dans les deux simulations, comme la température aux bords ou bien la vitesse. Les programmes on calculé l'évolution des différents modèles et j'ai finalement affiché les données obtenues de la façon la plus pertinente que possible pour les comparer.

4 Simulations

4.1 Simulation de particules

4.1.1 Conditions initiales et paramètres

Au début de la simulation une matrice de particules est générée avec des vitesses, des masses et des positions prédéfinies. La longueur et la largeur du rectangle dans lequel les particules se déplaceront sont aussi choisies à l'avance. Ce rectangle pourra, si cela est préférable pour mettre en évidence certains phénomènes être cyclique sur son axe x . Si c'est le cas on peut alors dire que le modèle décrira des particules se déplaçant sur la surface d'un cylindre.

4.1.2 Modèle et équations

Nous travaillons dans un modèle où les particules interagissent entre elles par des choc élastiques instantanés. Les équations qui régissent notre système de particules sont donc la conservation de l'énergie cinétique et la conservation de la quantité de mouvement. En considérant n masses m_i ($i = 1, 2, \dots, n$) ayant respectivement des vitesses $\mathbf{v}_i(t)$ à un temps t arbitraire et en posant E_c et \vec{p} l'énergie cinétique et la quantité de mouvement constantes totales de notre système, on a la conservation de l'énergie cinétique

$$E_c = \sum_{i=0}^n \frac{1}{2} m_i \mathbf{v}_i(t)^2 \quad (1)$$

et la conservation de la quantité de mouvement.

$$\vec{p} = \sum_{i=0}^n m_i \mathbf{v}_i(t) \quad (2)$$

Les collisions que nous considérons se produisent lorsque la distance entre deux particules est égale à la somme de leurs rayons et que les deux particules se rapprochent. Puisque les collisions sont instantanées on considère négligeable la probabilité que plus de deux particules entrent en collision au même instant. Lorsque une collision se produit entre deux particules on considère donc que les énergies cinétiques et les quantités de mouvement des autres particules sont constantes pendant un instant. Ainsi pour les vitesses initiales $v(t_0)$ et finales $v(t_1)$ lors d'une collision entre deux masses m_1, m_2 on peut poser :

$$E_c = \frac{1}{2} m_1 \mathbf{v}_1(t)^2 + \frac{1}{2} m_2 \mathbf{v}_2(t)^2 + \sum_{i=3}^n \frac{1}{2} m_i \mathbf{v}_i(t)^2 \quad (3)$$

Puisque $\sum_{i=3}^n \frac{1}{2} m_i \mathbf{v}_i(t)^2$ est désormais un terme constant on peut définir $C := E_c - \sum_{i=3}^n \frac{1}{2} m_i \mathbf{v}_i(t)^2$
On a donc

$$C = \frac{1}{2} m_1 \mathbf{v}_1(t_0)^2 + \frac{1}{2} m_2 \mathbf{v}_2(t_0)^2 \quad (4)$$

et

$$C = \frac{1}{2} m_1 \mathbf{v}_1(t_1)^2 + \frac{1}{2} m_2 \mathbf{v}_2(t_1)^2 \quad (5)$$

Ce qui implique naturellement que la somme des énergies cinétiques de nos deux particules sont constantes avant et après la collision. On peut faire de même pour les quantités de mouvement et obtenir que la somme des quantités de mouvement et la somme d'énergies cinétiques de nos deux particules restent constantes.

$$\frac{1}{2} m_1 \mathbf{v}_1(t_0)^2 + \frac{1}{2} m_2 \mathbf{v}_2(t_0)^2 = \frac{1}{2} m_1 \mathbf{v}_1(t_1)^2 + \frac{1}{2} m_2 \mathbf{v}_2(t_1)^2 \quad (6)$$

$$m_1 \mathbf{v}_1(t_0) + m_2 \mathbf{v}_2(t_0) = m_1 \mathbf{v}_1(t_1) + m_2 \mathbf{v}_2(t_1) \quad (7)$$

Pour simplifier les calculs des vitesses finales on va, pour l'instant, étudier le cas d'une collision à une seule dimension et nous généraliserons ensuite le résultat. Continuons donc en regroupant les termes facteurs de m_1 et m_2

$$m_1(\mathbf{v}_1(t_0)^2 - \mathbf{v}_1(t_1)^2) = m_2(\mathbf{v}_2(t_1)^2 - \mathbf{v}_2(t_0)^2) \quad (8)$$

$$m_1(\mathbf{v}_1(t_0) - \mathbf{v}_1(t_1)) = m_2(\mathbf{v}_2(t_1) - \mathbf{v}_2(t_0)) \quad (9)$$

Puisque il y a une collision les vitesses initiales et finales des particules sont différentes et donc les termes de (9) sont non nuls, on peut donc diviser (8) par (9) et cela nous donne

$$\mathbf{v}_1(t_0) + \mathbf{v}_1(t_1) = \mathbf{v}_2(t_1) + \mathbf{v}_2(t_0) \quad (10)$$

$$m_1(\mathbf{v}_1(t_0) - \mathbf{v}_1(t_1)) = m_2(\mathbf{v}_2(t_1) - \mathbf{v}_2(t_0)) \quad (11)$$

Ensuite isolons $\mathbf{v}_1(t_1)$ dans la première équation et $\mathbf{v}_2(t_1)$ dans la deuxième

$$\mathbf{v}_1(t_1) = \mathbf{v}_2(t_1) + \mathbf{v}_2(t_0) - \mathbf{v}_1(t_0) \quad (12)$$

$$\mathbf{v}_2(t_1) = \frac{m_1(\mathbf{v}_1(t_0) - \mathbf{v}_1(t_1))}{m_2} + \mathbf{v}_2(t_0) \quad (13)$$

Finalement en remplaçant $\mathbf{v}_2(t_1)$ dans(12)

$$\mathbf{v}_1(t_1) = \frac{(m_1 - m_2)\mathbf{v}_1(t_0) + 2m_2\mathbf{v}_2(t_0)}{m_1 + m_2} \quad (14)$$

Ceci nous permet de déterminer la vitesse après collision de la première particule. Puisque cette équation est obtenue à partir de deux équations symétriques (8) (9), on pourrait échanger m_1 et \mathbf{v}_1 avec m_2 et \mathbf{v}_2 et obtenir

$$\mathbf{v}_2(t_1) = \frac{(m_2 - m_1)\mathbf{v}_2(t_0) + 2m_1\mathbf{v}_1(t_0)}{m_2 + m_1} \quad (15)$$

Ainsi on sait calculer les changements de vitesses causés par les collisions de nos particules en une dimension !

Pour étendre ces résultats au cas général des collisions à deux dimensions commençons par illustrer le problème :

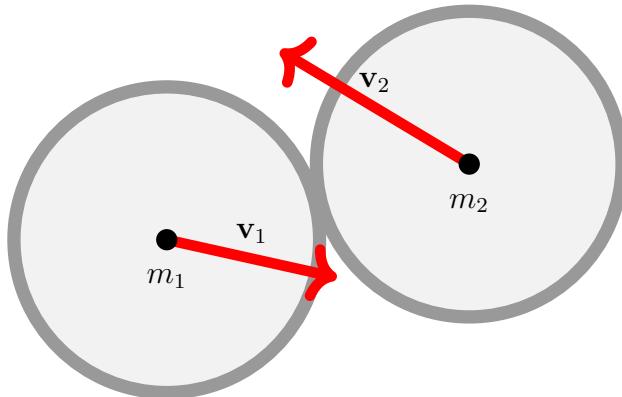


FIGURE 7 – Illustration d'une collision entre deux particules ayant des masses m_1 et m_2 et des vitesses \mathbf{v}_1 et \mathbf{v}_2

Pour nous ramener au cas d'une collision en deux dimensions il faut spécifier l'axe de collisions car jusqu'à maintenant les positions des balles n'entraînait pas dans le calcul des vitesses

finales alors qu'en deux dimensions il est évident que l'angle entre les particules à l'instant de l'impact a une importance. En effet, ce que nous n'avons pas mentionné encore ce sont les vecteurs \vec{f}_1 et \vec{f}_2 des forces qui empêchent les masses m_1 et m_2 de se traverser, ces derniers sont centrés sur le point d'impact et sont perpendiculaires à la tangente des deux cercles passant par le point d'impact. Voici ces vecteurs illustrés dans notre schéma précédent :

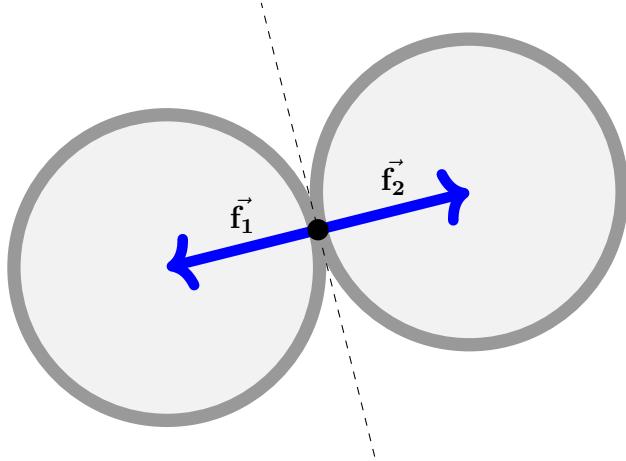


FIGURE 8 – Illustration des forces \vec{f}_1 et \vec{f}_2 agissant pendant la collision entre deux particules

En choisissant un axe x colinéaire aux vecteurs des forces on peut décomposer \mathbf{v}_1 et \mathbf{v}_2 en leurs composantes u et v projections des vitesses sur l'axe x et l'axe y respectivement :

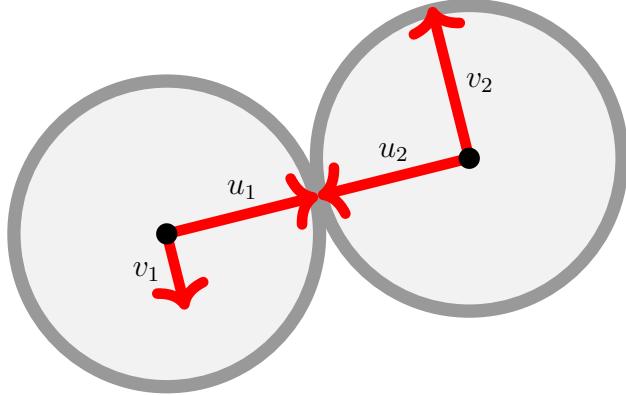


FIGURE 9 – Illustration d'une collision entre deux particules ayant des masses m_1 et m_2 et des vitesses \mathbf{v}_1 et \mathbf{v}_2 décomposées en v_1 , v_2 , u_1 et u_2

Les forces vont agir uniquement sur les vitesses en x puisque les forces sont alignées avec l'axe x. Les vitesses en y restent constantes comme les vitesses de toutes les balles non-impliquées dans la collision. On peut réécrire les équations (6) et (7) pour les composantes u des vitesses ce qui donne lieu à leurs équations de conservation :

$$u_1(t_1) = \frac{(m_1 - m_2)u_1(t_0) + 2m_2u_2(t_0)}{m_1 + m_2} \quad (16)$$

et

$$u_2(t_1) = \frac{(m_2 - m_1)u_2(t_0) + 2m_1u_1(t_0)}{m_2 + m_1} \quad (17)$$

Ces équations nous permettront de calculer les vitesses des particules après les collisions.

Pour finir, il faut également noter que les particules se déplacent dans un rectangle, ainsi, si les particules se trouvent à l'extérieur du rectangle après un déplacement, soit elles collisionnent avec le bord du rectangle qu'elles ont dépassé, leur vitesse est alors inversée par une force horizontale au bord, ce qui permet de conserver l'énergie cinétique du système mais pas sa quantité de mouvement.

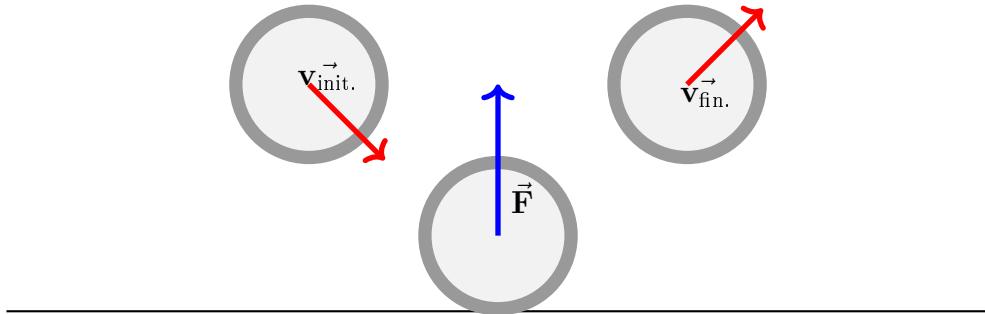


FIGURE 10 – Illustration du comportement d'une particule lorsqu'elle entre en collision avec une des parois du rectangle.

Soit le rectangle est considéré comme étant la surface d'un cylindre et la position en x est définie comme étant modulo la longueur du rectangle. Pour clarifier cela illustrons les axes x et y sur un cylindre :

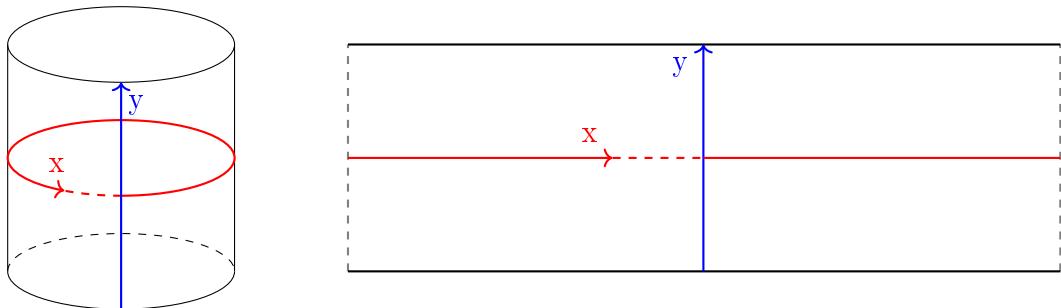


FIGURE 11 – Illustration du caractère cyclique du plan d'un cylindre

On voit à gauche le cylindre et à droite son plan mis à plat. Il est facile de voir que l'axe x est cyclique. Dans cette configuration on peut considérer que la largeur du rectangle de simulation est le périmètre du cylindre et que sa position en x est "modulo" le périmètre.⁴ Le but ici est de diminuer l'interaction des particules avec des bords, puisque ces dernières sont toujours les mêmes et donc peu intéressantes. Considérer la topologie d'un cylindre permet aux phénomènes, qu'on a choisis, de se développer plus aisément, surtout pour l'instabilité de Kelvin-Helmholtz qui a besoin de faire interagir deux longues couches de fluides ayant des vitesses différentes.

4.1.3 Discrétisation des équations

Pour pouvoir calculer numériquement le déplacement de nos particules, la chose la plus importante est de connaître les moments des collisions entre nos particules. Car ensuite l'horaire de la position de chaque particules ne dépend plus que de la vitesse. On pourrait trouver les moments et positions de collisions précis de façon analytique, mais cela serait, à priori, très intensif calculatoirement et compliqué à programmer. Au lieu de faire cela nous allons discréteriser le temps en un ensemble $t = 0, 1, \dots, n$ de n "étapes" temporelles, pour lesquelles on va successivement calculer les positions des particules. Le cycle commence avec les vitesses et

4. En fait on peut aussi représenter la position sur le plan du cylindre par (θ, y) où θ est un angle.

les positions des particules initiales et démarre en les déplaçant une première fois. Ensuite, on vérifie si certaines particules sont en collision et changeons leurs vitesses en conséquence. Une fois ce travail fait, les particules pourront être déplacées à nouveau en recommençant le cycle. Celui-ci est donc de le suivant :

Déplacement des particules → Collisions → Déplacement des particules → etc.

Chaque paire de déplacement-collision nous permet d'avancer au temps $t+1$. Puisque les vitesses des particules sont déterminées par les collisions dont on connaît déjà les solutions. Il faut simplement qu'on mette en évidence les calculs des positions successives des particules. Les équations différentielles des vitesses nous permettant de les trouver sont[21] :

$$\begin{aligned}\frac{\partial x}{\partial t} &= u \\ \frac{\partial y}{\partial t} &= v\end{aligned}$$

Avec u et v la vitesse horizontale et verticale, respectivement, d'un particule donnée et (x, y) sa position. En considérant $(x(t), y(t))$ la position de la particule à un temps t , on peut écrire la définition de la dérivée pour un Δt arbitrairement petit :

$$\begin{aligned}u &\approx \frac{x(t + \Delta t) - x(t)}{\Delta t} \\ v &\approx \frac{y(t + \Delta t) - y(t)}{\Delta t}\end{aligned}$$

En isolant les positions x et y au temps $t + \Delta t$, on a :

$$\begin{aligned}x(t + \Delta t) &\approx u \cdot \Delta t + x(t) \\ y(t + \Delta t) &\approx v \cdot \Delta t + y(t)\end{aligned}$$

Ce qui nous permet de calculer les nouvelles positions de chaque particule après un temps arbitraire, $\Delta t = 1$ dans notre cas, à partir de leur position au temps t .

4.2 Simulation Eulérienne

En mécanique des fluides, on modélise la matière par des fonctions continues[20]. Dans notre simulation de fluides notre objectif est donc de calculer les propriétés d'un fluide continu à travers le temps à partir d'un état initial défini. Un fluide continu est défini comme un milieu dont les propriétés, telles que la vitesse et la température, varient de manière continue dans l'espace et dans le temps. Grâce à cette définition il est possible de définir le comportement du fluide par des dérivées et un état initial de notre système.

Concrètement, notre fonction du fluide décrira la vitesse ponctuelle en fonction des coordonnées spatiales et du temps. La dérivée de cette fonction par rapport au temps donnera l'accélération du fluide. En ce qui concerne les dérivées spatiales, elles conduiront à la définition de la divergence de la vitesse et à son laplacien, pour plus d'explications à propos de ces notions, il y a l'annexe 2.

À la différence de notre simulation basée sur des particules, le modèle de notre simulation de fluides n'est pas composée d'un ensemble de masses ayant des position, formes et vitesses

spécifiques, il est composé d'une grille bidimensionnelle de volumes de contrôle⁵ entre lesquels les entrées et les sorties de fluides sont "contrôlées" pour satisfaire les différentes loi de conservation. [5]. Nous allons considérer des valeurs du fluide comme la vitesse, la pression, la température ou la densité uniquement sur chaque case de la grille et considérer que ces valeurs à une position spécifique n'a plus vraiment de sens une fois l'espace discrétisé de cette façon, mais elles peuvent être estimées selon les valeurs des cases les plus proches. La discréttisation des équations de navier-stokes permettra d'assurer que lorsque le pas de temps et les tailles des cases de la grille tendent vers zéro les valeurs représentées par les volumes de contrôle tendent vers les valeurs exactes du fluide continu.

4.2.1 Conditions initiales et paramètres

Les conditions initiales sont une certaine largeur et longueur qui définiront un rectangle de fluide. Ce dernier sera limité à cet espace. Cependant, comme pour la simulation de particules, ce rectangle représentera en général la surface d'un cylindre. Le système de coordonnées décrivant le fluide sera donc cyclique sur l'axe x . Le fluide aura des vitesses sur les bords du rectangle (ou du cylindre) prédéfinies et consantes. En plus de considérer un fluide, dans le rectangle il y a également un colorant arbitraire ayant une densité initiale sur l'ensemble du fluide et une densité constante sur les bords qui pourra éventuellement se diffuser. Justement la constante de diffusion est un autre paramètre ainsi que la viscosité du fluide.

4.2.2 Modèle et équations

Comme nous l'avons déjà expliqué le modèle considère un fluide dans un rectangle. Ce dernier à pour principal et unique propriété : sa vitesse dans l'espace (le plan dans notre cas). Elle est définie par la fonction $\mathbf{v}(x, y)$ avec (x, y) un point dans notre rectangle. Il faut maintenant écrire des équations afin que \mathbf{v} adopte le comportement d'un fluide. Nous choisissons pour l'instant de rendre le fluide incompressible et d'ignorer sa température ainsi que sa densité(c. à. d. de les considérer constantes et uniformes). Un tel fluide peut être décrit par les équations de Navier-Stokes que nous allons interpréter et dont nous allons préciser les unités de chaque terme pour que le lecteur puisse se faire une idée de ce qu'ils représentent :

1. l'équation de bilan de la masse :

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \quad [\text{Équation de bilan de la masse}] \quad (1)$$

Cette équation nous dit que, en un certain volume, la variation de densité dans le temps ($\text{kg m}^{-3} \text{s}^{-1}$) additionnée à la divergence dans l'espace de la quantité de mouvement ($(\text{kg/m}^3)(\text{m/s})/\text{m} = \text{kg m}^{-3} \text{s}^{-1}$) est nulle. Hors la divergence dans l'espace de la quantité de mouvement est simplement l'opposé de la "quantité" en (kg) qui, comme nous l'avons expliqué dans la section dédiée, "s'accumule" par seconde par mètre cube. Nous pouvons déplacer le deuxième terme de l'autre côté de l'équation :

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho \mathbf{v}) \quad (2)$$

Ici il est explicite que la variation de masse par mètre cube est égale à "l'accumulation" de masse due à son déplacement en suivant le champ de vitesse. Dans notre fluide incompressible on a une densité uniforme constante donc $\frac{\partial \rho}{\partial t} = 0$. On peut donc écrire :

$$\nabla \cdot (\rho \mathbf{v}) = 0 \quad (3)$$

5. En deux dimensions les volumes de contrôle sont bidimensionnels ce qui nous empêche théoriquement de les appeler "volumes" cependant on peut considérer que chaque "volume" a une profondeur arbitrairement petite qui laisse les propriétés du fluide inchangées

Qui se simplifie pour une densité constante ρ :

$$\nabla \cdot (\mathbf{v}) = 0 \quad [\text{Définition d'un fluide incompressible}] \quad (4)$$

Cette dernière équation est la définition d'un fluide incompressible à savoir que la divergence de vitesse est nulle. C'est à dire en quelque sorte qu'il n'y a aucun point où le fluide "s'accumule" ou bien "diverge". Une autre façon d'interpréter cela est que en prenant une quelconque limite spatiale définissant un volume en son sein, il y aura à chaque instant autant de fluide qui entre et qui sort de celui-ci.

Un exemple connu est celui du cas particulier où le champ est un champ magnétique. Il est incompressible et c'est pour cela que sur une surface de Gauss il y a autant de lignes de champ magnétique qui entrent que de lignes qui sortent[7].

Si on avait pu faire entrer plus de fluide que celui qui sort on aura en fait dû "comprimer" le fluide dedans d'où l'appellation fluide "incompressible".

Remarquons donc que dans un fluide à densité constante uniforme l'incompressibilité découle de l'équation de bilan de masse, il est également trivial que si un fluide a une densité uniforme, dans l'espace, à chaque instant de façon indépendante, alors, pour pouvoir conserver sa masse, la densité doit également être constante dans le temps. Cependant si la densité n'est pas uniforme dans l'espace, elle n'est plus nécessairement constante ($\frac{\partial \rho}{\partial t} \neq 0$ en général) et il n'est pas implicite que le fluide soit incompressible(4). Il n'est cependant pas impossible que ça soit le cas. C'est à dire qu'un fluide peut être incompressible en ayant une densité hétérogène et variable en chaque point de l'espace, comme par exemple lorsque l'on mélange deux liquides de densités différentes (par ex. de l'huile et de l'eau).

2. l'équation de bilan de la quantité de mouvement :

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{v} + \mathbf{g} \quad (5)$$

Qui, pour un fluide à densité constante $\rho = 1 \text{ kg m}^{-3}$ s'écrit :

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\nabla p + \nu \nabla^2 \mathbf{v} + \mathbf{g} \quad (6)$$

Cette équation comporte un certain nombre de termes, le plus important est $\frac{\partial \mathbf{v}}{\partial t}$ qui décrit le changement de vitesse, c'est à dire l'accélération (m s^{-2}). Elle est additionnée à $(\mathbf{v} \cdot \nabla) \mathbf{v}$ qui est un terme un peu plus compliqué. Ce dernier n'est pas, comme on pourrait le croire, équivalent à $\mathbf{v} \nabla \mathbf{v}$ ou à $(\nabla \cdot \mathbf{v}) \mathbf{v}$. En fait il est plutôt équivalent à $\nabla(\mathbf{v} \otimes \mathbf{v})$. Ici \otimes est le produit tensoriel. Il donne en retour une vitesse au carré ($\text{m}^2 \text{s}^{-2}$, dont sa divergence est donc une accélération (m s^{-2}). Nous n'allons pas nous arrêter pour comprendre exactement comment il est calculé. Mais considérons que ce dernier rend l'interprétation de $\nabla(\mathbf{v} \otimes \mathbf{v})$ semblable à celle de $\nabla \cdot (\rho \mathbf{v})$. Le deuxième indiquait, l'opposé de l'accumulation, donc "l'enlèvement" de ρ ammené par \mathbf{v} . Le premier représente donc "l'enlèvement" de \mathbf{v} par... \mathbf{v} la vitesse elle-même !

Une façon d'interpréter $\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v}$ est donc de dire que c'est la variation de la vitesse à un point donné à laquelle on enlève sa variation due simplement au champ de vitesses, donc au déplacement du fluide. Ainsi de l'autre côté de l'équation nous aurons les "forces" qui s'appliquent sur un point du fluide.

De façon plus générale pour toute quantité ϕ , scalaire ou vectorielle, $(\mathbf{V} \cdot \nabla) \phi$ décrit sa diminution à un point donné dû au déplacement de ϕ selon la vitesse \mathbf{v} . On appelle ceci "l'advection"[9] de la quantité ϕ .

$-\nabla p + \nu \nabla^2 \mathbf{v} + \mathbf{g}$ le plus simple des trois termes est \mathbf{g} qui est simplement la résultante des forces arbitraires appliquées en ce point. C'est souvent la gravité, mais ça pourrait être simplement un "jet" artificiel, la force d'un moteur ou n'importe quel autre chose dans ce style.

Ensuite il y a $-\nabla p$ qui est le gradient de pression, qui pointe donc vers la plus basse pression au lieu de la plus haute pression à cause de son signe négatif. Son unité est celle de la pression $\text{kgm/s}^2\text{m}^2 = \text{kg/s}^2\text{m}$. Cependant rappelons nous que dans l'équation (5) ce terme était divisé par la densité en kg m^{-3} ce qui donne donc m^2s^{-2} puisque on fait son gradient on divise par une distance et on obtient donc bien une accélération m s^{-2} . Au delà de l'interprétation des unités, il semble logique que la pression applique une force depuis les hautes pressions vers les plus basses pressions. On sait donc maintenant interpréter le terme de pression, mais en fait qu'elle est la pression de notre fluide ?

Jusqu'à maintenant nous n'en avons pas parlé et c'est dû au fait que on considère que le fluide est "incompressible". Dans ce contexte ce terme peut être mal interprété, en effet il ne signifie pas que le fluide n'a pas de pression, mais plutôt que son volume ne varie pas en fonction de sa pression ; qu'on ne peut pas le "compresser". Justement, ce qui va permettre au fluide de ne pas être comprimé, sous l'effet d'une force, c'est sa pression interne qui va "renvoyer" la force de pression à travers le fluide.

Pour se faire une meilleure idée du comportement de la pression on peut penser à un ballon de baudruche rempli d'eau, qui, quand il percute le sol après une chute sur plusieurs mètres, s'aplatit contre celui-ci et, éventuellement, explose. Lors de l'impact, le sol applique une force en direction de la balle sur toute la surface de contact ; une pression. Cette force, qui fait drastiquement augmenter la pression de l'eau dans le ballon, commence à faire effet par la partie qui a touché le sol en premier. Puisque la pression est, au début, plus forte sur le bas du ballon que sur le haut, le gradient de la pression applique une force normale au fluide, ce qui le fait décélérer. Cependant, sur les côtes, le fluide ne ressent aucune pression venant de l'extérieur, à part, peut-être, celle causée par la tension de la baudruche qui est négligeable avant l'impact⁶. Les composantes x et z , dans le plan du sol, du gradient⁷ de la pression vont donc vers l'extérieur ce qui cause "l'aplatissement" du ballon. Ensuite, pendant que le ballon se déforme, la tension de la baudruche augmente, ce qui applique une pression au fluide depuis les côtés où ce dernier s'est étendu. Cela va donc finir par rééquilibrer le gradient des pressions et limiter "l'aplatissement" du ballon. Si cela n'arrive pas, la baudruche atteint sa tension maximale et se casse ; le ballon explose.

Durant tout ce processus c'est la pression interne de l'eau qui l'a rendue "incompressible". C'est elle qui équilibre les forces pour que sur chaque point elles n'induisent pas une accélération qui causerait une vitesse dont la divergence n'est pas nulle. En faisant cela elle transmet donc la "pression". Dans ce cas, depuis la partie basse du ballon vers ses côtes. Notons que la pression dans les fluides compressibles transmettrait également la pression d'une façon semblable. Cependant elle ne pourrait pas être aussi simplement déterminée par les forces agissant sur le fluide car ce dernier aurait la capacité de se "comprimer" et donc d'admettre une divergence des forces.

Un autre exemple, peut-être plus intuitif, est celui d'un barrage hydroélectrique où l'eau est immobile et soumise à la force de pesanteur. Pour éviter que cette dernière ne se

6. En fait, à priori, avant l'impact, la tension de la baudruche induit une pression depuis toute sa surface vers l'intérieur du ballon. Du point de vue d'un point à l'intérieur du ballon, la pression amenée par celle-ci est donc la même dans chaque direction. Le gradient de cette pression est donc nul

7. on parle ici de $-\nabla p$ donc l'opposé du gradient de pression qui pointe vers la plus grande "diminution" de la pression

comprime, il faut que le gradient de la pression applique une force sur chaque volume d'eau de sorte à ce que celui-ci reste immobile, étant donné que les seules deux forces agissant sur le fluide sont la gravité et la pression. En estimant la pression à la surface comme étant nulle, on peut trouver la pression à mesure que la profondeur augmente ; il suffit de la faire augmenter de façon à que son gradient corresponde l'opposé de la force de pesanteur.

Cette caractéristique de la pression peut être difficile à comprendre. Mais elle est essentielle pour pouvoir résoudre les équations de navier-stokes d'un fluide incompressible. Il faut bien comprendre qu'elle est déterminée entièrement par sa nature à rendre le fluide incompressible.

Finalement on a $\nu\nabla^2\mathbf{v}$ qui est certainement le plus compliqué des trois termes. Il décrit la force induite par la viscosité du fluide (ν). On peut également écrire ce terme comme $\nu\Delta\mathbf{v}$ étant donné que comme nous avons vu dans la section dédiée, le laplacien est ∇^2 . Dans ce cas cependant, nous avons le laplacien d'un champ vectoriel et non pas d'un scalaire, comme nous l'avions dit ce dernier se calcule comme le laplacien de chaque composante de \mathbf{v} :

$$\nu\Delta\mathbf{v} = \nu(\Delta\mathbf{v}_x, \Delta\mathbf{v}_y, \Delta\mathbf{v}_z) \quad (7)$$

Une interprétation de par exemple la composante x de ce dernier est qu'elle décrit la divergence du gradient de la composante horizontale de \mathbf{v} c'est à dire la divergence du champ allant vers la plus grande augmentation de v_x possible. Donc si par exemple dans un certain point le gradient de v_x diverge, cela veut dire que elle "part" du point dans toutes les directions et que donc v_x augmente dans toutes les directions autour du point choisi. À l'inverse si v_x diminuait dans toutes les directions à partir du point choisi, le gradient pointerait donc vers le point choisi et la divergence serait négative.

En résumé, si autour d'un point v_x diminue alors la composante x de $\Delta\mathbf{v}$ sera négative, et si v_x augmente elle sera positive. Donc la "force" en x donnée par ce terme visqueux dépend de si le fluide autour du point va plus ou moins vite en x , si il va plus vite la force est positive sinon elle est négative.

On peut interpréter les 2 autres composantes de façon similaire et donc globalement la force visqueuse en un point va avoir un vecteur décrivant la vitesse "en plus" que le fluide tends à avoir autour de lui. Ainsi, sous son effet, chaque point va graduellement accélérer ou décélérer pour avoir la même vitesse que le fluide environnant.

Justement, lorsqu'on imagine un fluide très visqueux comme par exemple du miel. On voit que sa vitesse est assez homogène, on aurait par exemple du mal imaginer une "tornade" de miel. Alors que dans des fluides peu visqueux comme l'air ou l'eau on peut observer des tornades dans l'air ou des vortex lorsqu'on tire la chasse d'eau par exemple. Ces tourbillons impliquent que dans le fluide il y ait de grands changements de vitesses selon la position. Ce qui explique que pour qu'ils se créent il faut que la viscosité du fluide soit basse. En pratique, si je remplaçais l'eau de ma chasse d'eau par du miel. Il n'y aurait probablement aucun tourbillon qui se créerait lorsque je la tire, en fait de par sa viscosité le miel collerait au surfaces en essayant d'imiter leur vitesse nulle et mettrait donc longtemps à descendre dans les tuyaux.⁸

On a donc compris ce que signifiait le terme $\nu\nabla^2\mathbf{v}$, ν étant simplement un scalaire indiquant à quel point le fluide est visqueux. Au niveau des unités, $\nabla^2\mathbf{v}$ donne une vitesse

8. En allant à l'extrême, il y a des fluides comme le "poix" qui sont tellement visqueux qu'ils peuvent sembler solides. Il y a à ce propos une expérience de l'université de Queensland qui fait couler du poix depuis un entonnoir goutte par goutte depuis 1927 afin de démontrer son caractère fluide, ce qui lui a valu le record de la plus longue expérience de laboratoire fonctionnant en continu au monde.[16] Elle peut être observée en direct, la dixième goutte devrait tomber dans les prochaines années...

par mètre carré (m/s)/ m^2 = ms . ν est un viscosité cinématique dont l'unité est $\text{m}^2 \text{s}^{-1}$. La multiplication des deux donne donc bien m s^{-2} . [14]

De tout cela on retient donc que : $\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\nabla p + \nu \nabla^2 \mathbf{v} + \mathbf{g}$ Indique que le changement de vitesse du fluide à un point donné en ignorant celui dû au simple déplacement de celui-ci et égal à celui induit par la somme de la force de pression, la force de viscosité et la force de pesanteur.

Ces deux équations sont celles qui, en les résolvant, permettront de connaître l'évolution dans le temps de notre fluide. Cependant une solution analytique "exacte" de ces équations n'est possible que pour des cas où le champ des vitesses initiales est très simple. Il va donc falloir approximer la solution...

4.2.3 Discrétisation et résolution des équations

Pour simuler numériquement un tel fluide il faut tout d'abord discréteriser le temps, au lieu de considérer notre fluide à $t \in \mathbb{R}$ on va considérer $t = 0, 1, 2, 3, 4, \dots, n$ pour une simulation de n étapes. Ensuite, il faut discréteriser l'espace dans lequel on travaille, ou, plus précisément, le plan dans lequel on travaille. C'est à dire considérer que le fluide est défini sur un nombre fini de points dans le plan et non pas sur une continuité de points infinis. Pour effectuer cela, on définit ce qu'on appelle une grille de volumes de référence, qui dans notre cas sera une grille bidimensionnelle où chaque case est une "aire" de référence⁹, et dont on considère les vitesses $\mathbf{v}_{x,y}$ à des points situés au milieu de chaque case de la grille :

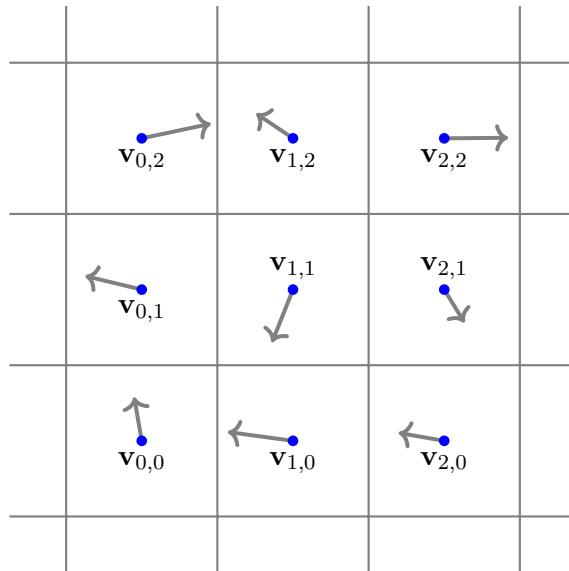


FIGURE 12 – Illustration d'une grille de volumes de référence ayant des vitesses en leur centre.

Cette méthode est appelée "description eulérienne" car elle va évaluer le champ des vitesses du fluide à des points fixes. La description Lagrangienne en comparaison aurait voulu qu'on suive les vitesses de points "fluides" se déplaçant avec celui-ci[11][12].

La première fonction fondamentale de cette grille est qu'elle nous permet de définir la vitesse initiale à partir d'un fichier de données où un nombre correspond à chaque composante de la vitesse de chaque case. La deuxième fonction encore plus essentielle est qu'elle nous sert à calculer le champ de vitesse à un temps $t + 1$

9. On peut quand même considérer que ce sont des "volume" en considérant que les cases ont une hauteur infinitésimale dt , cependant concrètement cela ne changera rien mathématiquement

4.2.3.1 Calcul de la pression

Puisque comme nous l'avons expliqué, la propriété qui rend le fluide incompressible est la pression commençons par considérer $\nabla \mathbf{v}$ qui nous donne la divergence de la fonction vectorielle de vitesses \mathbf{v} :

$$\nabla \mathbf{v} = \frac{\partial \mathbf{v}_x}{\partial x} + \frac{\partial \mathbf{v}_y}{\partial y} \quad (8)$$

Nous devons nous assurer que la divergence du champ vectoriel des vitesses soit nul. Cela est fait en utilisant le théorème fondamental du calcul vectoriel[29] qui nous dit que tout champ vectoriel de \mathbb{R} dont la dérivée est continue (on suppose que c'était le cas dans notre champ de vitesses avant de l'avoir discrétilisé), et qui est défini sur un domaine Ω compact et connexe (Ce qui est le cas car on travaille dans un rectangle, qui est un objet connexe, de \mathbb{R}^2 fermé et borné et donc compact [28]) peut être décomposé en un champ solénoidal, c'est à dire dont la divergence est nulle, et en un champ irrotationnel. Cette décomposition est appelée la décomposition de Helmholtz d'un champ vectoriel, en voici une illustration :

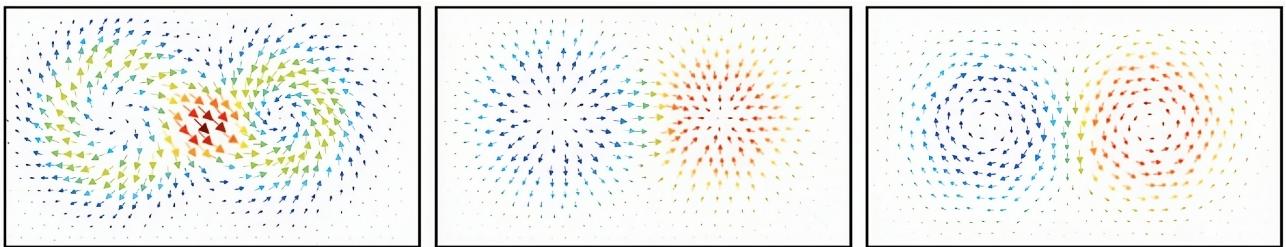


FIGURE 13 – Décomposition d'un champ vectoriel (gauche) en un champ irrotationnel (milieu) et solénoidal (droite)

Il se trouve qu'un champ irrotationnel est en général le gradient d'un certain champ scalaire qui dans notre cas particulier sera justement p car c'est la pression du fluide en chaque point qui est chargée de rendre ce dernier incompressible. Pour clarifier cela, voyons un exemple de comment agirait ce dernier dans un point où la divergence serait non-nulle :

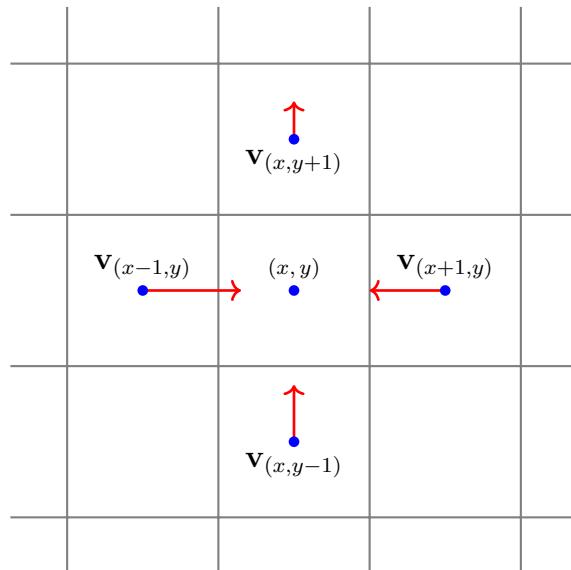


FIGURE 14 – Illustration de la divergence en un point (x, y) .

Nous voyons dans ce schéma que la divergence en (x, y) n'est clairement pas nulle car il y a plus de vitesse entrante que sortante. Choisissons donc une valeur de $p(x, y) > 0$ et appliquons le

gradient de la pression ∇p sur les cases adjacentes pour ainsi "annuler" la divergence. Puisqu'on considère que la pression des cases environnantes est nulle, il est logique que le gradient de p noté \vec{p} agisse, avec une intensité proportionnelle à $p(x, y)$, sur les cases adjacentes comme suit :

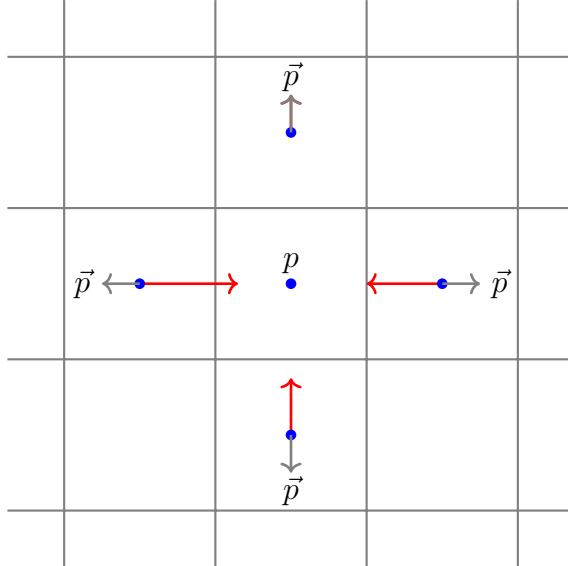


FIGURE 15 – Application d'une pression $p(x, y)$ pour annuler la divergence.

En appliquant la pression de cette façon à la vitesse, on peut atteindre une divergence nulle à la case (x, y) . Pour comprendre pourquoi, contentons-nous de remarquer que $\nabla(\mathbf{v}(x, y) - \vec{p}(x, y)) = \nabla\mathbf{v}(x, y) - \nabla\vec{p}(x, y)$ et que la manière dont les vecteurs de \vec{p} sont disposés montrent une divergence positive. Ainsi avec un $p(x, y)$ bien choisi on devrait avoir $\nabla\mathbf{v}(x, y) - \nabla\vec{p}(x, y) = 0$. On pourrait donc dire qu'on a trouvé la pression à cet endroit. Cependant, il ne faut pas oublier que l'on est dans une grille et qu'il va falloir trouver une pression sur chaque case individuellement. Si on déterminait ainsi la vitesse de chaque case, puisqu'à chaque fois qu'on détermine la pression sur une case le gradient de pression des cases adjacentes change, notre calcul risquerait de donner une divergence nulle uniquement sur la dernière case dont on a calculé la "pression".

Au lieu de faire cela soyons un peu plus rigoureux, et notons la pression $p_{(x,y)}$ à chaque point de notre grille, supposons que la divergence de notre vitesse $\mathbf{v}(t)$ ne soit pas nulle. Nous calculerons alors p de sorte à ce qu'après son application, notre vitesse finale $\mathbf{v}(t + 1)$ ait de nouveau une divergence nulle. Considérons donc la façon dont p agit sur \mathbf{v} dans l'équation de conservation de la quantité de mouvement en ignorant tous les autres termes ; Pour l'instant nous allons calculer l'évolution de \mathbf{v} sous le seul effet de la pression :

$$\frac{\partial \mathbf{v}}{\partial t} = -\nabla p \quad (9)$$

Notons $\mathbf{v}' := \mathbf{v}(t + 1)$. En faisant une différence avant dans le temps sur \mathbf{v} et une différence centrale dans l'espace sur ∇p , on obtient :

$$\mathbf{v}'_{x,y} \approx \mathbf{v}_{x,y} - \left(\frac{p_{(x+1,y)} - p_{(x-1,y)}}{2}, \frac{p_{(x,y+1)} - p_{(x,y-1)}}{2} \right) \quad (10)$$

On remarque que de cette façon la pression p conserve la quantité de mouvement étant donné qu'elle soustrait de la vitesse aux cases dont les coordonnées sont supérieures à elle d'un côté

mais en ajoute tout autant de l'autre côté. De plus on veut que la divergence de notre nouvelle vitesse \mathbf{v}' soit nulle :

$$\nabla \mathbf{v}' = 0 \Leftrightarrow \frac{\mathbf{v}'_{x+1} - \mathbf{v}'_{x-1} + \mathbf{v}'_{y+1} - \mathbf{v}'_{y-1}}{2} \approx 0$$

En précisant que :

\mathbf{v}'_{x+1} est la composante x de \mathbf{v}' en $(x+1, y)$ \mathbf{v}'_{x-1} est la composante x de \mathbf{v}' en $(x-1, y)$,

\mathbf{v}'_{y+1} est la composante y de \mathbf{v}' en $(x, y+1)$ \mathbf{v}'_{y-1} est la composante y de \mathbf{v}' en $(x, y-1)$

En insérant l'équation (10), on peut réécrire (en utilisant une notation similaire pour \mathbf{v}) :

$$0 \approx \frac{(\mathbf{v}_{x+1} - \frac{p_{x+2}-p_x}{2}) - (\mathbf{v}_{x-1} - \frac{p_x-p_{x-2}}{2})}{2} + \frac{(\mathbf{v}_{y+1} - \frac{p_{y+2}-p_y}{2}) - (\mathbf{v}_{y-1} - \frac{p_y-p_{y-2}}{2})}{2}$$

Réarrangeant les termes, on obtient :

$$0 \approx \frac{\mathbf{v}_{x+1} - \mathbf{v}_{x-1} + \mathbf{v}_{y+1} - \mathbf{v}_{y-1}}{2} - \frac{p_{x+2} - 2p_{x,y} + p_{x-2} + p_{y+2} - 2p_{x,y} + p_{y-2}}{4} \approx \nabla \mathbf{v} - \nabla^2 p$$

En effet, la discrétisation par une différence centrale de deuxième ordre de $\nabla^2 u$ s'écrit :

$$\nabla^2 u \approx \frac{u_{x+h} - 2u_{x,y} + u_{x-h} + u_{y+h} - 2u_{x,y} + u_{y-h}}{h^2}$$

Ici on avait donc $h = 2$. On donc a obtenu l'équation suivante :

$$\nabla^2 p = \nabla \mathbf{v}$$

Pour résumer, on montré que :

$$\begin{cases} \mathbf{v}' = \mathbf{v} - \nabla p \\ \nabla \mathbf{v}' = 0 \end{cases} \implies \nabla^2 p = \nabla \mathbf{v}$$

Ce qui peut se simplifier en :

$$\nabla(\mathbf{v} - \nabla p) = 0 \implies \nabla \mathbf{v} - \nabla^2 p = 0$$

On aurait pu également arriver à la même conclusion en remarquant que :

$$\nabla(\mathbf{v} - \nabla p) = \nabla \mathbf{v} - \nabla^2 p$$

Mais manipuler des opérateurs différentiels est assez délicat, passer par les discrétisation nous à permis de mieux comprendre ce que l'on faisait.

Quoi qu'il en soit, étant donné que $\nabla \mathbf{v}$ est connu, on peut maintenant calculer notre fonction p par une discrétisation centrée d'ordre 2 :

$$p_{x+1} - 2p_{x,y} + p_{x-1} + p_{y+1} - 2p_{x,y} + p_{y-1} \approx \nabla \mathbf{v}_{x,y} \quad (11)$$

On réarrange cette équation pour obtenir :

$$p_{x+1} + p_{x-1} + p_{y+1} + p_{y-1} - 4p_{x,y} \approx \nabla \mathbf{v}_{x,y} \quad (12)$$

Et on isole $p_{x,y}$:

$$p_{x,y} \approx \frac{(p_{x+1,y} + p_{x-1,y} + p_{x,y-1} + p_{x,y+1}) - \nabla \mathbf{v}_{x,y}}{4} \quad (13)$$

Cependant, me direz-vous, $p_{x+1,y}, p_{x-1,y}, p_{x,y-1}, p_{x,y+1}$ ne sont pas connus. Cela est vrai, nous devons donc résoudre un système d'équations qui contient XY inconnues en considérant que

notre grille est de dimension X verticale et Y horizontale. On peut représenter ce système d'équations par une matrice de $A \in M_{XY,XY}(\mathbb{R})$ tel que :

$$Ap = \nabla \mathbf{v}$$

Ou sous une forme développée :

$$\begin{pmatrix} -4 & 1 & 0 & \cdots & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & -4 & 1 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & -4 & \cdots & 0 & 0 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & \cdots & -4 & 1 & 0 & \cdots & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & \cdots & 1 & -4 & 1 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 1 & -4 & \cdots & 0 & 0 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & -4 & 1 & 0 & \cdots & 1 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 1 & -4 & 1 & \cdots & 0 & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 1 & \cdots & 0 & 1 & -4 & \cdots & 0 & 0 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & -4 & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 1 & -4 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & \cdots & 0 & 1 & -4 \end{pmatrix} \begin{pmatrix} p_{1,1} \\ p_{2,1} \\ p_{3,1} \\ \vdots \\ p_{1,2} \\ p_{2,2} \\ p_{3,2} \\ \vdots \\ p_{X-2,Y-1} \\ p_{X-1,Y-1} \\ p_{X,Y-1} \\ \vdots \\ p_{X,Y} \\ p_{X,Y} \\ p_{X,Y} \end{pmatrix} = \begin{pmatrix} \nabla \mathbf{v}_{1,1} \\ \nabla \mathbf{v}_{2,1} \\ \nabla \mathbf{v}_{3,1} \\ \vdots \\ \nabla \mathbf{v}_{1,2} \\ \nabla \mathbf{v}_{2,2} \\ \nabla \mathbf{v}_{3,2} \\ \vdots \\ \nabla \mathbf{v}_{X-2,Y-1} \\ \nabla \mathbf{v}_{X-1,Y-1} \\ \nabla \mathbf{v}_{X,Y-1} \\ \vdots \\ \nabla \mathbf{v}_{X,Y} \\ \nabla \mathbf{v}_{X,Y} \\ \nabla \mathbf{v}_{X,Y} \end{pmatrix}$$

À première vue la matrice A semble un peu compliquée ; laissez-moi vous la présenter :

1. Sa diagonale vaut -4 et représente le coefficient de $p_{x,y}$ dans l'équation (12)
2. les 1 à droite et à gauche adjacents à la diagonale représentent les coefficients de $p_{x-1,y}$ et de $p_{x+1,y}$ dans l'équation (1)
3. les 1 encore plus à droite ou plus à gauche représentent les coefficients de $p_{x,y+1}$ et de $p_{x,y-1}$ dans l'équation (1)
4. tous les autres coefficients sont des zéros

Les coefficients de p sont simplement $p_{(1,1)}, p_{(2,1)}, \dots, p_{(1,2)}, p_{(2,2)}, \dots, p_{(X-1,Y)}, p_{(X,Y)}$ de façon à ce que tous les $p_{(x,y)}$ y soient inclus. Cette matrice est essentiellement remplie de zéros. On peut utiliser la méthode de Jacobi[22] pour approximer les valeurs de $p_{x,y}$ ¹⁰. Cette méthode consiste à définir une valeur de départ P_0 :

$$P_0 = \begin{pmatrix} P_{1,1}^{(0)} \\ \vdots \\ P_{X,1}^{(0)} \\ \vdots \\ P_{X,Y}^{(0)} \end{pmatrix}$$

On considère $M := -4I_{XY}$ la matrice ayant -4 sur sa diagonale, son inverse est donc $M^{-1} = -\frac{1}{4}I_{XY}$. Ensuite on définit $B := M^{-1}(M - A) = \frac{1}{4}(A - M)$ qui est donc la matrice A avec des 0 sur la diagonale dont tous les coefficients sont divisés par 4. Puis, la méthode de Jacobi nous dit de calculer la suite P_k pour $k = 1, 2, \dots$ par la récurrence suivante :

$$X_{k+1} = BX_k + M^{-1}\Delta \mathbf{v}$$

10. Pour pouvoir utiliser la méthode il faut que A satisfasse certaines conditions, en l'occurrence notre matrice devrait les satisfaire. Le lecteur qui veut approfondir cette méthode est incité à visiter la page wikipedia de la méthode de Jacobi pour en apprendre plus à ce sujet.

En fait, elle revient à calculer chaque nouveau coefficient de P_k comme étant :

$$P_{x,y}^{(t+1)} \approx \frac{(P_{x+1,y}^{(t)} + P_{x-1,y}^{(t)} + P_{x,y-1}^{(t)} + P_{x,y+1}^{(t)}) - \nabla \mathbf{v}_{x,y}}{4} \quad (14)$$

Qui est en fait simplement l'équation ?? en considérant la partie gauche de l'équation comme étant "nouvelle".

Jacobi nous dit que cette suite tends vers le vecteur décrivant les pressions p . Ce qui implique que l'on peut choisir un nombre k assez haut, que l'on peut choisir arbitrairement, pour obtenir X_k "proche" de p . Sinon on peut aussi choisir un $\delta > 0$ arbitrairement petit et on calcule la suite jusqu'à avoir un k assez grand tel que :

$$|AX_k - \nabla \mathbf{v}| < \delta$$

C'est une méthode plus "sûre" car de cette façon il n'y a pas de doute sur le fait qu'on a enlevé la divergence de \mathbf{v} et on peut choisir le niveau de précision désiré. Une fois que l'on a approximé p , plus ou moins finement, on calcule la nouvelle valeur de \mathbf{v} :

$$\mathbf{v}' = \mathbf{v} - \nabla p$$

Que l'on calcule par une discréétisation que l'on a déjà décrit :

$$\mathbf{v}'_{x,y} \approx \mathbf{v}_{x,y} - \left(\frac{p_{(x+1,y)} - p_{(x-1,y)}}{2}, \frac{p_{(x,y+1)} - p_{(x,y-1)}}{2} \right)$$

A partir de là on a un fluide incompressible, youpi !

Rappelons que on n'a pour l'instant décrit l'évolution de \mathbf{v} qu'en fonction de la pression. Pour calculer la vraie valeur de $\mathbf{v}(t+1)$ il faut coupler le résultat à ceux des autres composants....

4.2.3.2 Calcul de l'advection

Pour calculer $(\mathbf{v} \cdot \nabla) \mathbf{v}$ en chaque point, écrivons le d'abord sous forme vectorielle :

$$(\mathbf{v} \cdot \nabla) \mathbf{v} = \begin{pmatrix} \mathbf{v}_x \frac{\partial \mathbf{v}_x}{\partial x} + \mathbf{v}_y \frac{\partial \mathbf{v}_x}{\partial y} \\ \mathbf{v}_x \frac{\partial \mathbf{v}_y}{\partial x} + \mathbf{v}_y \frac{\partial \mathbf{v}_y}{\partial y} \end{pmatrix}$$

Pour simplifier les calculs nous prendrons un différent repère pour chaque point (x, y) de sorte à ce que la composante y de la vitesse (\mathbf{x}, \mathbf{y}) soit nulle. On aura donc :

$$(\mathbf{v} \cdot \nabla) \mathbf{v} = \begin{pmatrix} \mathbf{v}_x \frac{\partial \mathbf{v}_x}{\partial x} \\ \mathbf{v}_x \frac{\partial \mathbf{v}_y}{\partial x} \end{pmatrix}$$

Pour préserver le caractère temporel du pas de notre simulation on va le noter $\Delta t = 1$. Faisons donc une différence arrière d'une distance $\Delta t \mathbf{v}_x$:

$$(\mathbf{v} \cdot \nabla) \mathbf{v} \approx \begin{pmatrix} v_{(x,y)} \frac{v_{(x,y)} - v_{(x-\Delta t \mathbf{v}_x, y)}}{\Delta t v_{(x,y)}} \\ v_{(x,y)} \frac{u_{(x,y)} - u_{(x-\Delta t \mathbf{v}_x, y)}}{\Delta t v_{(x,y)}} \end{pmatrix} = \begin{pmatrix} \frac{v_{x,y} - v_{x-\Delta t \mathbf{v}_x, y}}{\Delta t} \\ \frac{u_{x,y} - u_{x-\Delta t \mathbf{v}_x, y}}{\Delta t} \end{pmatrix} = \frac{\mathbf{v}_{(x,y)} - \mathbf{v}_{(x-\Delta t \mathbf{v}_x, y)}}{\Delta t}$$

On incorpore cette dernière dans l'égalité :

$$\frac{\partial \mathbf{v}}{\partial t} = -(\mathbf{v} \cdot \nabla) \mathbf{v}$$

En discréteisant également la dérivé de la vitesse $\frac{\mathbf{v}_{t+1} - \mathbf{v}_t}{\Delta t}$ cela nous donne :

$$\frac{\mathbf{v}_{t+1} - \mathbf{v}_t}{\Delta t} = -\frac{\mathbf{v}_{(x,y)} - \mathbf{v}_{(x-\nabla t \mathbf{v}_x, y)}}{\Delta t}$$

On isole \mathbf{v}_{t+1} :

$$\mathbf{v}_{t+1} = \mathbf{v}_{(x-\nabla t \mathbf{v}_x, y)}$$

On se souvient que pour obtenir cela on a dû aligner l'axe des x avec la vitesse du point (x, y) . Ainsi \mathbf{v}_{t+1} est la vitesse \mathbf{t} prise en une position "dérrière" (x, y) selon sa vitesse et le pas de temps ($\nabla t \mathbf{v}_x = \mathbf{v}_x$). Illustrons cela avec un schéma :

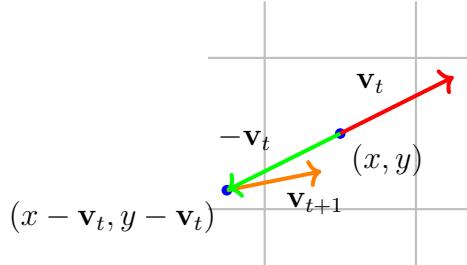


FIGURE 16 – Illustration de l'advection de la vitesse d'un point (x, y) . La vitesse \mathbf{v}_{t+1} est obtenue en se déplaçant en "arrière" d'un vecteur $-\mathbf{v}_t$ à partir de (x, y) .

Cependant, nous connaissons les vitesses (x, y) uniquement lorsque x et y sont des entiers, on va donc approximer la valeur de la vitesse en $(x - \mathbf{v}_t, y - \mathbf{v}_t)$ par une interpolation linéaire des quatre valeurs $v_{(x_1, y_1)}, v_{(x_2, y_1)}, v_{(x_1, y_2)}, v_{(x_2, y_2)}$ de vitesses connues qui l'entourent. En prenant $0 \leq a, b \leq 1$ tel que $(x, y) = (x_1 + a(x_2 - x_1), y_1 + b(y_2 - y_1))$ on calcule :

$$\begin{aligned} v_1 &= v_{(x_1, y_1)} + b(v_{(x_1, y_2)} - v_{(x_1, y_1)}) \\ v_2 &= v_{(x_2, y_1)} + b(v_{(x_2, y_2)} - v_{(x_2, y_1)}) \\ v_{(x,y)} &\approx v_1 + a(v_2 - v_1) \end{aligned}$$

Nous pouvons illustrer ces calculs par un schéma :

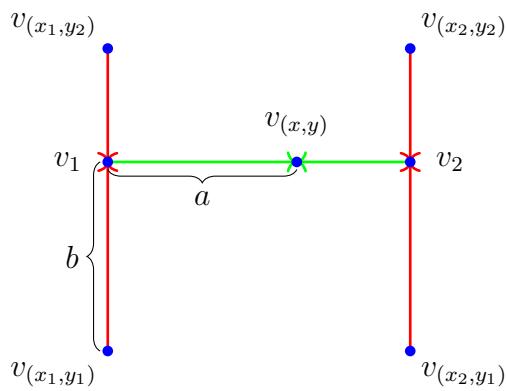


FIGURE 17 – Interpolation linéaire pour estimer $v_{(x,y)}$

Intuitivement, dans ce schéma on voit qu'on "mélange" les points l'un sur l'autre 2 à 2 pour arriver à la même hauteur que $\mathbf{v}_{(x,y)}$ (flèches rouges) puis on "mélange" linéairement de nouveau les deux points obtenus (flèches vertes) pour obtenir une valeur estimée au point désiré. Avec cette technique il est assez intuitif que lorsque les dimensions de chaque case de la grille, tendent vers 0 notre valeur approximée tends vers la valeur réelle de $v_{(x,y)}$, étant donné que la

grille deviendra de plus en plus dense.

Nous savons donc désormais calculer $\mathbf{v}(t + 1)$ après l'application de l'advection de la vitesse. Notons que l'algorithme qu'on a trouvé pour "déplacer" la vitesse fonctionne aussi pour l'advection de n'importe quelle autre quantité se déplaçant avec le fluide. On l'utilisera notamment pour déplacer la densité d'un colorant et aussi pour le déplacement de la température.

4.2.3.3 Calcul de la viscosité (Diffusion)

Il ne reste plus qu'à calculer $\nu \nabla^2 \mathbf{v}$ qui décrit la diffusion de la vitesse par viscosité cinématique $\nu > 0$. Posons :

$$\frac{\partial \mathbf{v}}{\partial t} = \nu \nabla^2 \mathbf{v}$$

Nous pouvons appliquer à cela une discréétisation implicite (en considérant $\Delta t = \Delta x = \Delta y = 1$) :

$$\mathbf{v}_{t+1} - \mathbf{v}_t = \nu(\mathbf{v}_{(x+1,y)} - 2\mathbf{v}_{(x,y)} + \mathbf{v}_{(x-1,y)} + \mathbf{v}_{(x,y+1)} - 2\mathbf{v}_{(x,y)} + \mathbf{v}_{(x,y-1)}) \quad (15)$$

($\mathbf{v}_{(x,y)}$ sont donc évalués au temps $t + 1$) On isole \mathbf{v}_{t+1} :

$$\mathbf{v}_{t+1} = \frac{\mathbf{v}_t + \nu(\mathbf{v}_{(x+1,y)} + \mathbf{v}_{(x-1,y)} + \mathbf{v}_{(x,y+1)} + \mathbf{v}_{(x,y-1)})}{4\nu + 1} \quad (16)$$

Cela nous donne un système d'équations qu'on peut résoudre en utilisant la méthode de Jacobi¹¹ que l'on a déjà décrite. Le calcul pour la récurrence consiste de nouveau à choisir des valeurs de \mathbf{v} et à calculer $\mathbf{v}^{(k+1)}$ selon l'équation (16) que l'on vient d'obtenir.

Ce que fait la viscosité comme nous en avons déjà parlé c'est qu'elle "diffuse" la vitesse de façon à la rendre homogène. On peut utiliser ce mécanisme de diffusion aussi pour diffuser la densité d'un colorant dans le fluide en remplaçant dans les équations la vitesse par la densité. En fait selon certaines conditions l'équation qu'on vient de résoudre est semblable à celle de la chaleur[13] :

$$\frac{\partial T}{\partial t} = D \nabla^2 T \quad (17)$$

Où la fonction scalaire T décrit la température à un point donné et D est le "coefficent de diffusion thermique" qui détermine la vitesse à laquelle la température est conduite dans le fluide.

4.2.3.4 Application des calculs

On a donc résolu les équations de Navier Stokes pour des fluides incompressibles :

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\nabla p + \nu \nabla^2 \mathbf{v} + \mathbf{g}$$

En effet, on a décrit comment chaque terme influençait le calcul \mathbf{v}_{t+1} . Dans le programme de la simulation, chaque calcul de \mathbf{v}_{t+1} est fait selon chaque terme de façon indépendante. Donc on va en réalité faire un "séquence" d'étapes dans cet ordre[8] :

Diffusion(D) → Application de la Pression(P) → Advection(A) → Application de la Pression(P)
 Premièrement, précisons que chaque "étape" va appliquer un changement à la grille des vitesses enregistrées dans notre programme. Il faut bien comprendre que chaque "étape" prends le résultat de celle précédente pour en donner un nouveau.

11. Une des conditions suffisantes pour pouvoir utiliser la méthode de Jacobi est que la matrice A soit à diagonale strictement dominante. Ici, les coefficients sur la diagonale de la matrice correspondante, qui s'obtient en isolant \mathbf{v} dans l'équation (16), sont $4\nu + 1$ et les coefficients des autres termes sont 0 sauf ceux des quatres vecteurs vitesses adjacents dont les coefficient sont $-\nu$. La matrice est donc à diagonale strictement dominante car $4\nu + 1 > 4|-\nu|$.

Ensuite, on remarque qu'on applique la pression deux fois au lieu d'une. En fait, puisque le rôle de la pression est d'annuler la divergence du champ de vitesse on va l'appliquer à chaque fois que le champ vectoriel de la vitesse \mathbf{v} est modifié. On a le droit de faire cela autant de fois qu'on veut, étant donné que la pression est déterminée de façon à annuler la divergence.

Finalement notons que la diffusion est appliquée avant l'advection, ça n'a en fait pas vraiment d'importance, on aurait pu très bien les inverser ; puisque de toute façon on va répéter cette séquence un grand nombre de fois on ne va pas vraiment reconnaître la différence entre : D-P-A-P-D-P-A-P-D-P-.. et A-P-D-P-A-P-D-P-... La seule chose qui va changer légèrement c'est qu'on va considérer que le champ vectoriel \mathbf{v} à chaque temps t est celui auquel on vient d'appliquer l'advection au lieu de la diffusion. Cette imprécision tends vers 0 avec le pas de temps Δt .

Ainsi on ne résout pas "réellement" toute l'équation de conservation de la quantité de mouvement en un bloc mais en parties séparées. On aurait pu très bien calculer tout les termes de l'équation à chaque étape et mettre à jour \mathbf{v} en une seule fois en additionnant/soustrayant chaque terme. Mais, premièrement, cela demanderait plus d'espace mémoire et deuxièmement, à mon avis, cela n'améliorerait pas de beaucoup la précision.

Pour finir, comme on l'a dit pendant l'explication des différents calculs, il est facilement possible de considérer la température du fluide et un colorant ayant une "densité" arbitraire. On calculera, pour maintenir une certaine cohérence, leurs advection et diffusion en même temps que l'advection et la diffusion du champ des vitesses. En plus de cela, en temps normal, dans un fluide la masse volumique est impactée par la différence de température, ainsi une augmentation/diminution hétérogène de celle-ci devrait provoquer des courants. Nous allons cependant considérer ces dernier négligeables pour ne considérer que l'impact sur la "flottabilité" du fluide induite par la force d'archimède selon la masse volumique. Comme décrit par l'approximation de Boussinesq[10]. On va, en fait, "ajouter" une force verticale proportionnelle à la température du fluide ce qui forme donc une nouvelle "étape" dans le calcul de $\mathbf{v}_{(t+1)}$ après laquelle on va appliquer de nouveau la pression.

5 Résultats

Nous parlerons maintenant des phénomènes physiques que j'ai tenté simulé et analyserons les différents résultats obtenus.

5.1 Instabilité de Kelvin-Helmholtz



FIGURE 18 – Illustrations de l'instabilité de Kelvin-Helmholtz[4]

Le premier phénomène que j'ai voulu simuler à été l'instabilité de Kelvin-Helmholtz. Ce n'était pas la première que je voulais simuler, mais c'était la plus simple.

5.1.1 Description du phénomène physique

L'instabilité de Kelvin-Helmholtz (KH) se crée lorsque deux fluides sont superposés et se déplacent à une vitesse sensiblement différente sur leur surface de contact. Elle a été étudiée au *XIX* siècle par les physiciens Lord Kelvin et Hermann von Helmholtz. Ils ont remarqué que la différence de vitesse entre deux fluides crée un flot turbulent. [19] Ce flot turbulent se produit pour n'importe quelle différence de vitesse à moins que la tension de surface ou que la force gravité, plus importante sur un fluide que sur l'autre, les en empêche.

Pour générer le phénomène numériquement, il faut en général considérer deux masses de fluides ayant des vitesses opposées auquelles on rajoute du "bruit"¹² pour casser leur symétrie. Au lieu de mettre du bruit, pour mieux comprendre comment se crée le phénomène, simulons en 2 dimensions deux fluides superposés ayant des vitesses horizontales sensiblement différentes mais uniformes sur chaque fluide et ayant des vitesses verticales nulles. À l'exception d'un point à l'interface entre les deux fluides.

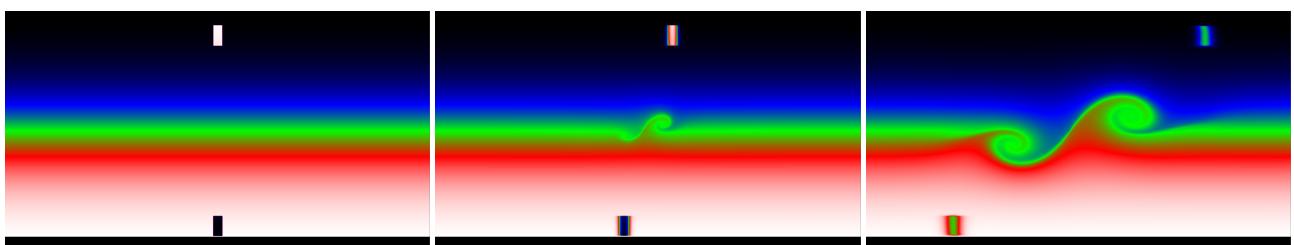


FIGURE 19 – Illustration de l'instabilité créée à l'interface des deux fluides avec une forte vitesse initiale verticale au centre

Dans ces figures les couleurs décrivent la densité d'un colorant (noir < bleu < vert < rouge < blanc). Deux rectangles sont visibles sur le haut et le bas du graphique, ils ont une densité qui contraste avec celle environnante, ils servent de repère pour se faire une idée du déplacement des deux couches de fluide. Nous voyons qu'au début il y a un gradient de densité car en bas la densité est maximale et elle diminue graduellement jusqu'en haut du graphique où elle est nulle. Il est un peu difficile de comprendre pourquoi ces cellules de convection se créent. Premièrement on peut remarquer que dans un fluide incompressible puisque la divergence de la vitesse est nulle, en considérant un fluide au repos, si on ajoute une vitesse locale, il y a automatiquement une forme de convection circulaire qui se crée pour "compenser" cette vitesse.

12. Ici le "bruit" est un champ vectoriel dont les composantes sont aléatoirement générées.

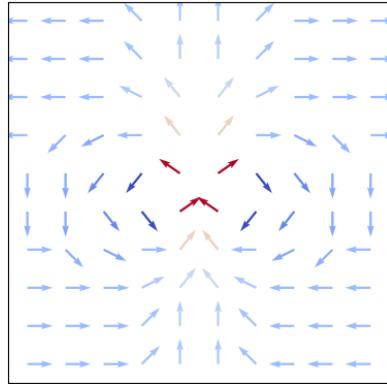


FIGURE 20 – Illustration du champ vectoriel d'un fluide après avoir ajouté une vitesse verticale en son centre

Dans les fluides compressibles l'effet n'est pas aussi fort mais il devrait à mon avis rester quand même non négligeable à cause de la pression interne. Dès lors la vraie question n'est pas tellement de savoir pourquoi il y a des convections circulaires, mais de comprendre pourquoi elles deviennent de plus en plus grandes. En effet, dans l'image 19 on voit que la forme de la convection reste à peu près la même pendant qu'elle s'agrandit.

Pour comprendre ce qui produit cela observons le champ vectoriel sur l'instabilité :

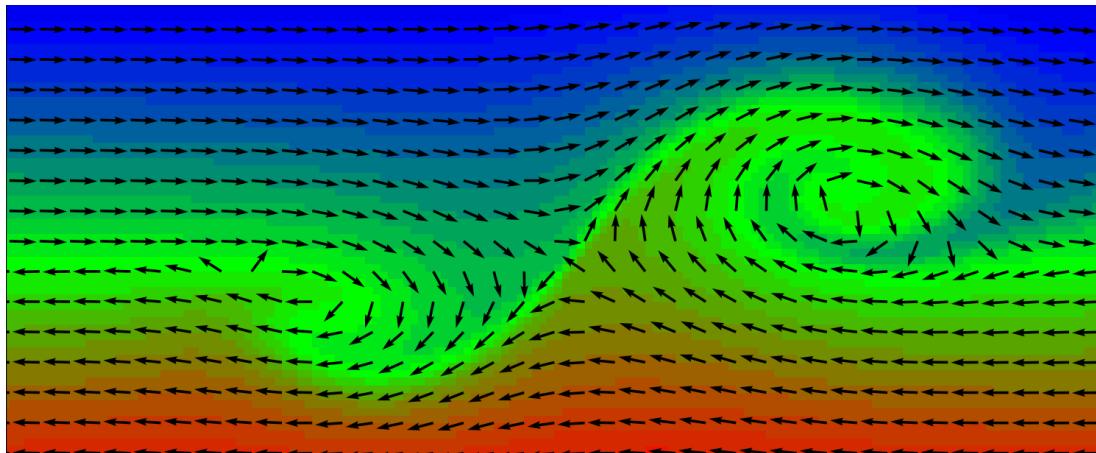


FIGURE 21 – Illustration du champ vectoriel de l'instabilité de Kelvin-Helmholtz

Grâce à ce graphique nous comprenons qu'au milieu du phénomène les deux courants opposés se rencontrent de façon frontale. Ce qui alimente les deux convections circulaires, car les deux flux symétriques et opposées sont obligées de "sortir" par les côtés. Voici une illustration simplifiée de ce qui arrive :

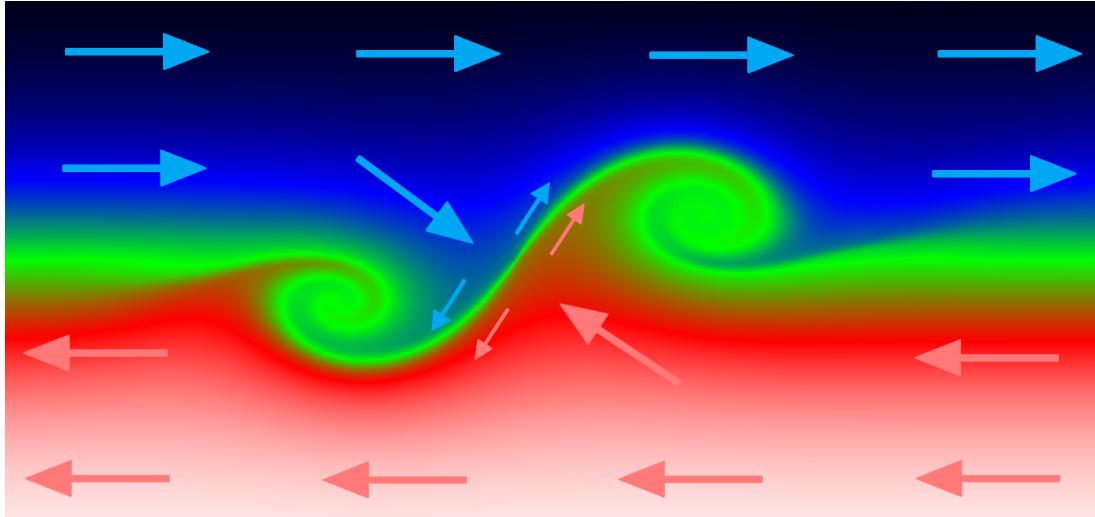


FIGURE 22 – Illustration du champ vectoriel simplifié causant l’instabilité de Kelvin-Helmholtz

Ainsi, nous comprenons un peu mieux comment cette instabilité fonctionne. Une dernière chose à noter à son propos est qu’elle a tendance à grandir graduellement jusqu’à atteindre les mêmes dimensions du fluide dans lequel elle est. Une fois cette taille atteinte elle se désintègre. Maintenant voyons si nous avons réussi à la simuler...

5.1.2 Résultats de la simulation Eulérienne

Pour simuler l’instabilité de Kelvin-Helmholtz dans la simulation eulérienne, on définit des vitesses positives dans la moitié supérieure du carré de simulation (30×100) et des vitesses négatives dans la moitié inférieure. Les valeurs ne sont pas exactes, en effet le module des vitesses de chaque case est aléatoirement choisi dans l’intervalle $[4, 5]$. On fait exprès cela pour créer l’instabilité, sinon, comme nous l’avons dit plus tôt, le modèle resterait parfaitement symétrique.

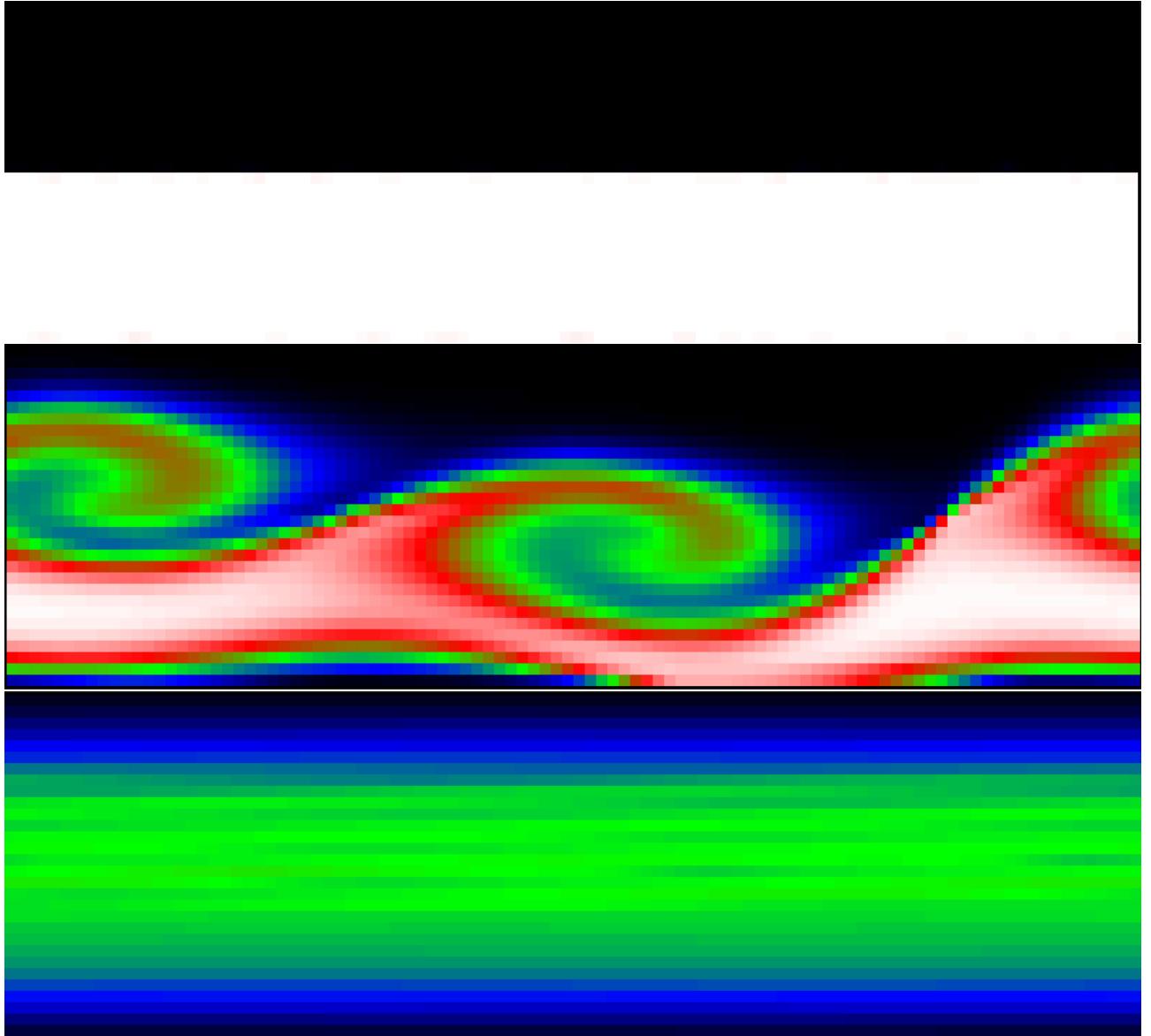


FIGURE 23 – Évolution de la densité (blanche>rouge>verte>bleue>noire) au fil du temps

Nous voyons exactement ce que l'on a décrit plus haut. Le nuage de densité, qui peut être vu comme une sorte de colorant, est au départ ($t = 0$) uniquement sur la moitié inférieure du rectangle de simulation. Puis à ($t \approx 28$) on observe que le colorant a pris une forme de vague, et finalement la simulation se termine avec le nuage de colorant qui se désintègre. Il n'est plus qu'un mélange lisse.

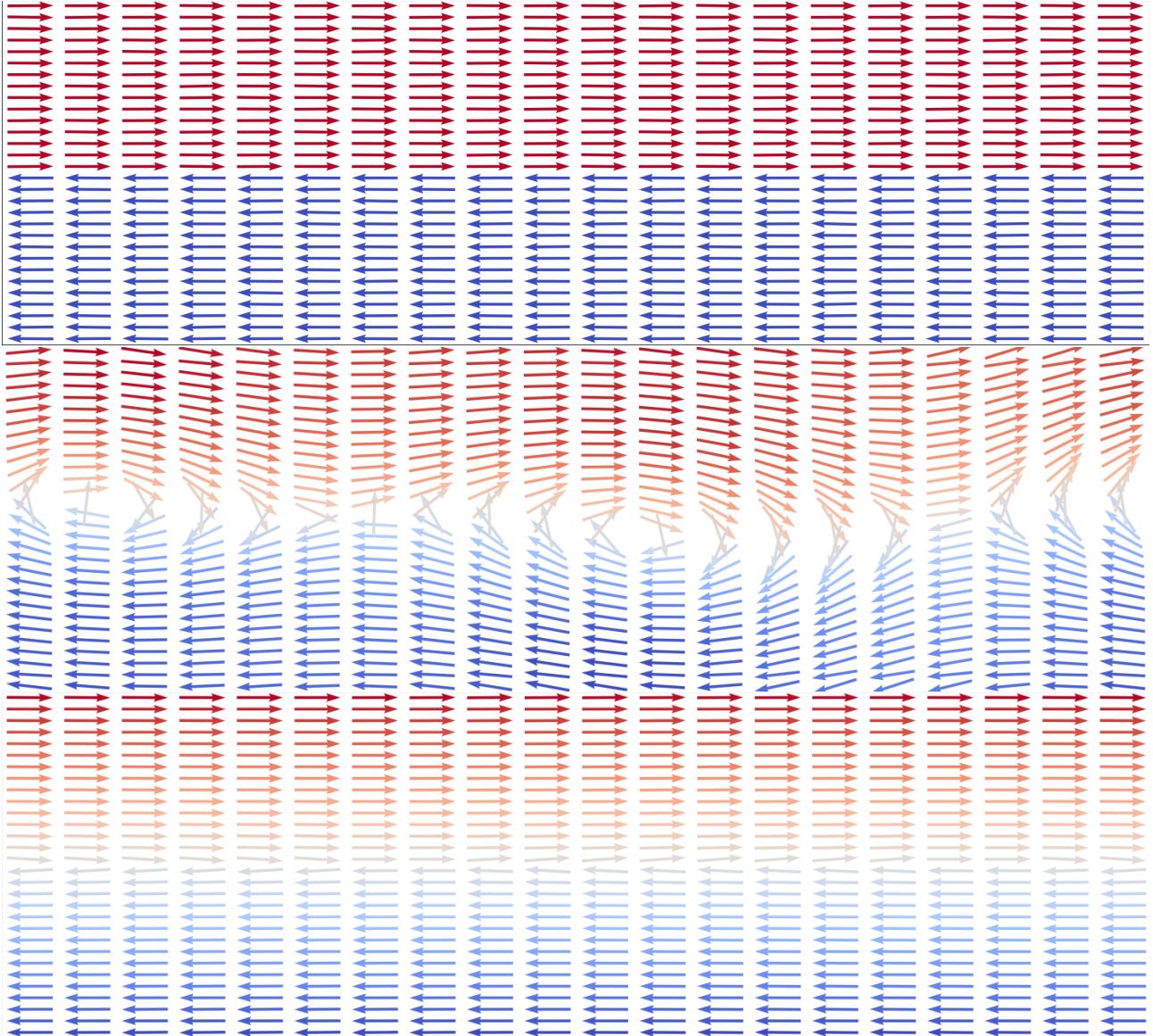


FIGURE 24 – Champ vectoriel normalisé de la vitesse au fil du temps. Vitesse horizontale : Rouge>Blanc>Bleu

Pour le champ de la vitesse c'est la même chose. Au début il est clairement séparé en deux parties ayant des vitesses opposées. Ensuite des convections rotatives se sont créées ($t \approx 28$) et à la fin il n'y a plus qu'un gradient de vitesses ($t \approx 140$). Il est intéressant de noter que au niveau macroscopique le système "perd" son caractère symétrique pendant la phase d'instabilité, puis le regagne à la fin.

La simulation basée sur la grille eulérienne est donc un franc succès pour simuler l'instabilité de Kelvin-Helmholtz. D'ailleurs les images utilisées pour la description physique ont été obtenues avec une de mes simulations eulériennes.

5.1.3 Résultats de la simulation de particules

Pour la simulation de particules l'affaire s'avère plus complexe. Pour tenter de simuler l'instabilité de Kelvin-Helmholtz j'ai reproduit avec des particules un scénario similaire. Les particules se déplacent dans un carré 50×50 dont l'axe x est cyclique, à des vitesses horizontales dans l'intervalle $[4,5]$. Dans ces conditions j'ai simulé 8100 particules de rayon 0.25 réparties

équitablement entre la moitié supérieure et inférieure du carré. Cependant, cette fois, à cause de la lenteur de la simulation j'ai simulé uniquement jusqu'à $t = 50$

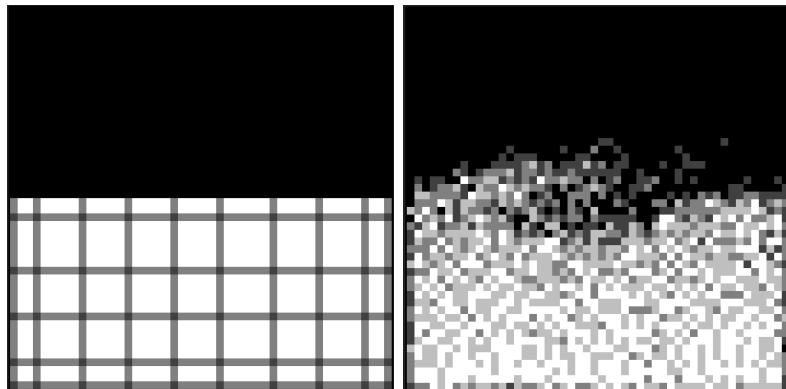


FIGURE 25 – Densités des particules générées sur la moitié inférieure, à $t = 0$ et à $t \approx 23$

Nous voyons un schéma à carreaux à $t = 0$ ce qui semble inquiétant car cela peut indiquer que les particules ne sont pas générées de façon uniforme. Cependant ce patron vient simplement du fait que les particules sont justement générées sur un quadrillage uniforme mais que l'espace entre chaque particule est inférieur à la dimension de chaque pixel. Puisque l'espace entre chaque particule n'est pas un diviseur de la dimension de chaque pixel cela fait que dans certains pixels il y a plus de particules que sur d'autres. Ce qui explique pourquoi certains pixels de l'image sont plus foncés et forment une sorte de "meta-quadrillage". Mais il faut bien comprendre qu'il n'a aucune importance au delà de celle estétique ; il ne représente en fait rien d'interessant. Ensuite à $t = 23$ on voit que les 4050 balles, que l'on avait générées sur la partie basse de la simulation, semblent s'être déplacées en un léger mouvement de convection. Cependant, il y a pas mal de bruit qui est causé par la dispersion aléatoire des particules. Même en essayant avec des configurations différentes de la simulation, la convection caractéristique de l'instabilité de Rayleigh-Benard est très peu visible dans le forme que prennent les particules.

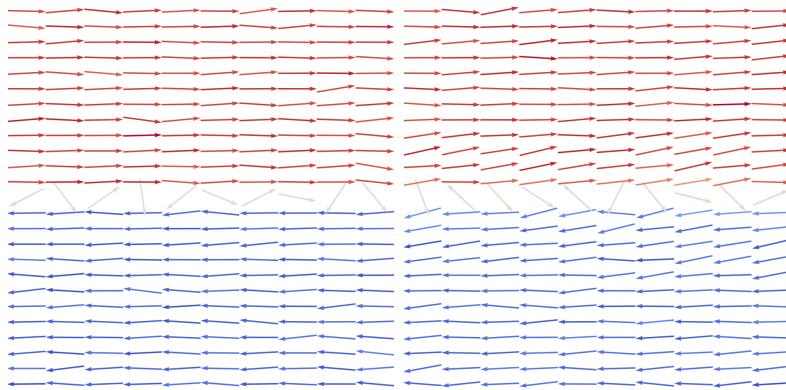


FIGURE 26 – Vitesses horizontales des particules à $t = 0$ et à $t \approx 23$

Sur les champs vectoriels des vitesses aussi, il est assez difficile de voir un quelconque phénomène, en fait il est même assez difficile de remarquer une quelconque différence entre le champ vectoriel de la vitesse initial et celui à $t = 23$. Il n'est pas affiché ici, mais celui à $t = 50$ n'est pas très différent non plus. Il faudrait probablement laisser la simulation tourner plus longtemps pour que se forme un gradient linéaire entre le haut et le bas de la simulation. Mais à priori, attendre que le gradient se fasse ne nous permettra pas de voir une instabilité se créer, puisque c'est justement lorsque les deux fluides ont des vitesses très différentes que

l'instabilité se crée le plus rapidement.

Ma simulation de particules échoue donc à simuler l'instabilité de Kelvin-Helmholtz.

5.2 Rayleigh-Bénard

5.2.1 Description du phénomène physique

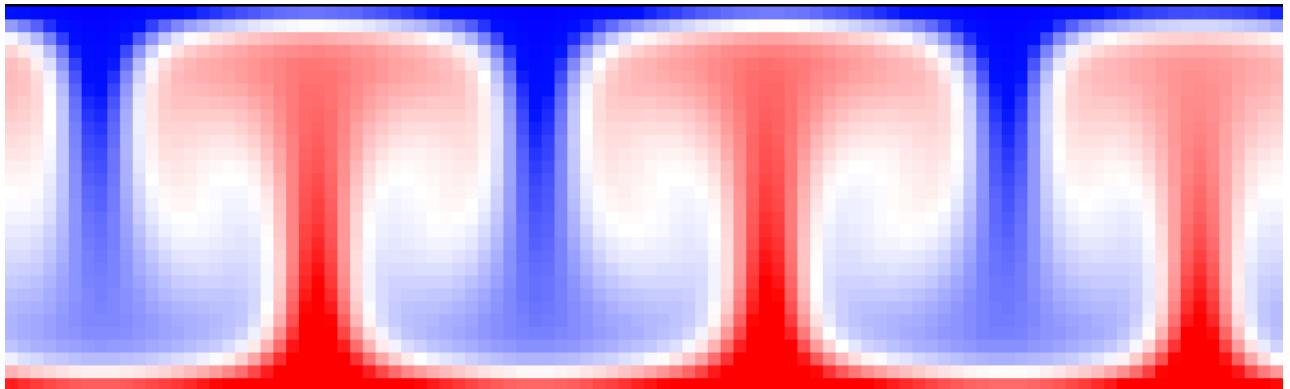


FIGURE 27 – Illustrations de l'instabilité de Rayleigh-Bénard

Le deuxième phénomène que j'ai simulé est l'instabilité de Rayleigh-Bénard. Cette instabilité se crée dans un fluide homogène réchauffé par le bas et refroidit par le haut. Sous l'effet de la chaleur, le fluide se dilate légèrement, sa masse volumique diminue et donc par simple force d'archimède il monte à la surface. Cependant pour remonter il doit se frayer un chemin. Si le fluide était parfaitement immobile, et donc symétrique, l'instabilité ne pourrait pas se développer. Cependant il suffit qu'il y a un léger courant ascendant, pour que le fluide chaud s'engoufre dedans et l'amplifie. Une fois le fluide chaud arrivé à la surface, il se refroidit et replonge sur les côtés. En deux dimension le flux est donc divisé en deux parties : droite et gauche. Au début de cette façon, plusieurs cellules se créent de tailles différentes, mais à mesure qu'elles occupent tout l'espace à disposition elle se serrent, certaines fusionnent et finissent par se stabiliser. Une propriété intéressante des convections de Rayleigh-Bénard est que même des perturbations microscopiques de l'état initial peuvent amener à des changements macroscopiques dans les cellules de convection finales[27], c'est donc un exemple d'effet papillon. En général, l'endroit où elles se créeront est indéterminé. On peut noter également que c'est un exemple de structure dissipative car elle augmente la vitesse à laquelle la température est transmise entre le haut et le bas du fluide.¹³

Sur l'image 27

5.2.2 Résultats de la simulation Eulérienne

Comme nous l'avons expliqué, nous pouvons simuler la température dans la simulation Eulérienne. Pour reproduire l'instabilité de Rayleigh-Bénard on définit une température nulle sur tout le domaine, sauf en haut et en bas, où on définit des températures respectivement négatives et positives constantes. On obtient alors, lentement, des convections caractéristiques de l'instabilité de Rayleigh-Bénard dans la simulation :

13. Cette affirmation provient de cette vidéo, si vous voulez comprendre ce que cela veut dire je vous conseille de la regarder.

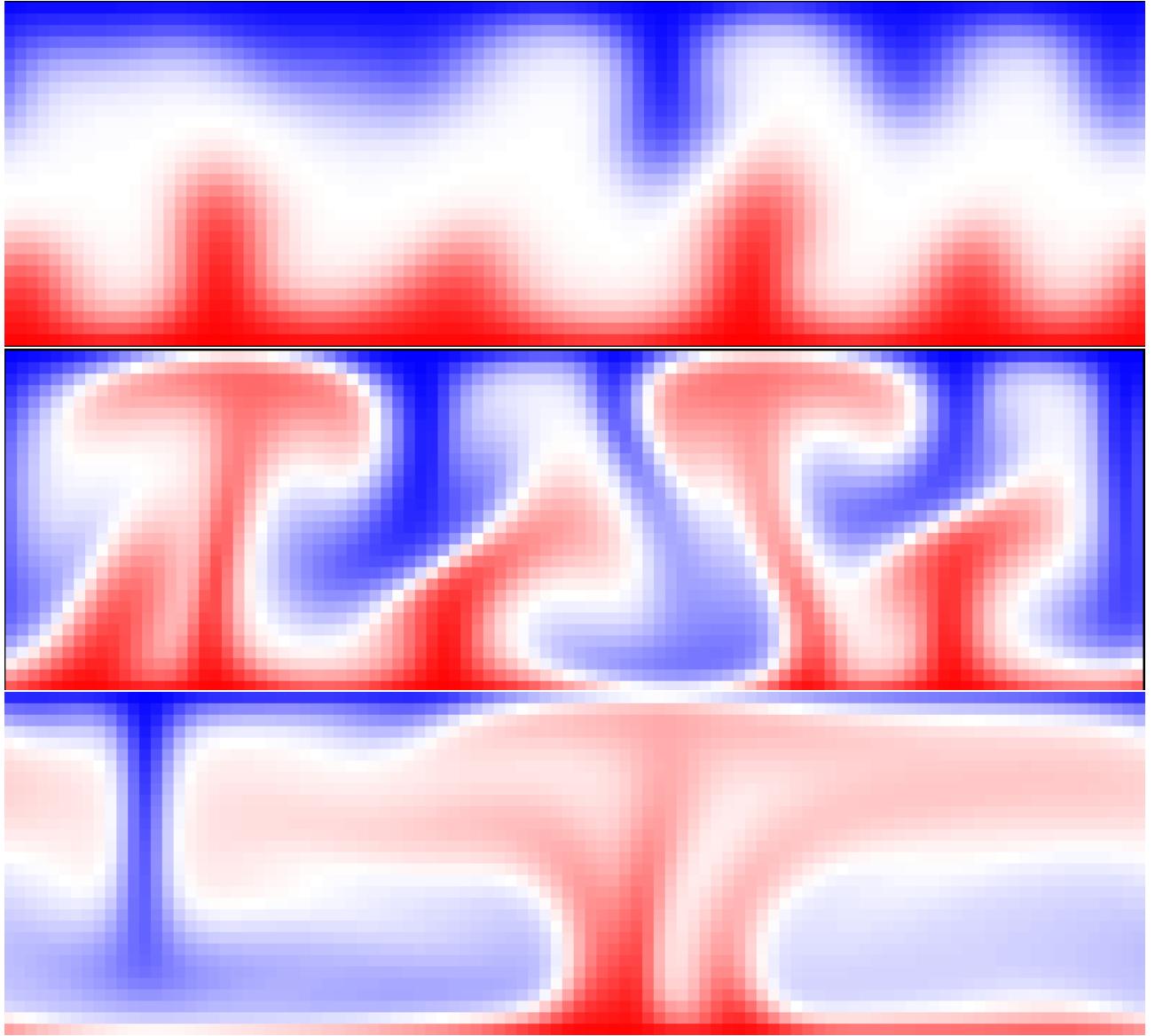


FIGURE 28 – Température du fluide (rouge>blanc>bleu) à $t \approx 140$, $t \approx 200$ et $t \approx 600$

Sur la première image, on peut voir les flux chauds et froids commencer à monter/tomber du fait de leur masse volumique différente. Ensuite à $t \approx 200$ les courants arrivent plus ou moins du côté opposé. Cette simulation spécifique a été faite avec une viscosité très faible et donc à $t \approx 600$ on voit que les convections ne sont toujours pas dans un état "ordonné". En fait, il faut obligatoirement augmenter la viscosité pour arriver à un état stable :

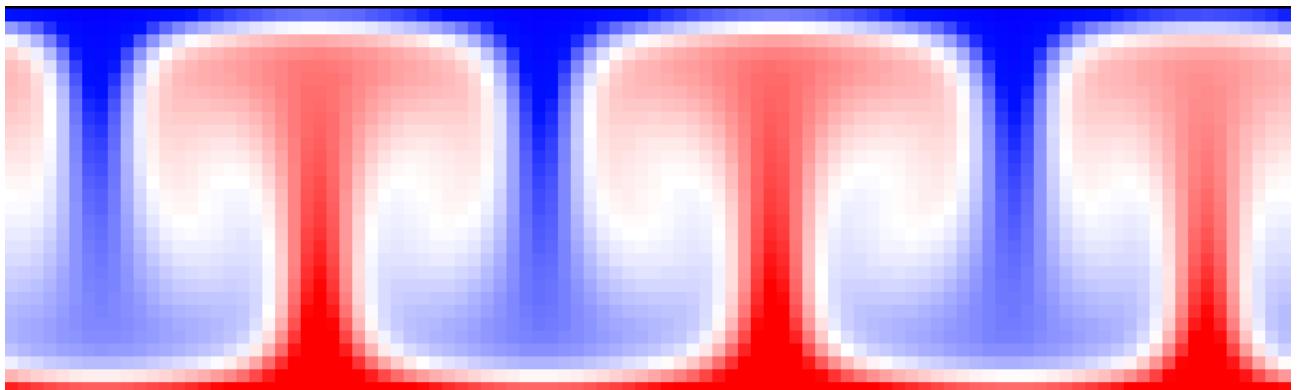


FIGURE 29 – Température d'un fluide visqueux (rouge>blanc>bleu) à $t \approx 1000$

On constate à nouveau que pour une simulation eulérienne, simuler ce genre de phénomènes est un jeu d'enfants.

5.2.3 Résultats de la simulation de particules

Pour tenter de simuler ce phénomène dans la simulation de particules, j'ai dû rajouter des étapes à la simulation. Premièrement, j'ai tout simplement appliqué la force de gravité en retirant une constante à la composante verticale des vitesses des particules à chaque instant. Ensuite pour simuler le fait qu'on "réchauffe" le fluide par en bas et qu'on le "refroidit" par en haut, j'ai, à chaque temps, fait collisionner chaque particule étant proche du haut ou du bas de la simulation avec des particules "imaginaires" de vitesses élevées ou nulles (chaud ou froid) à un angle aléatoire.

Avec ces modifications j'ai simulé ≈ 9600 particules de rayon 0.26 sur une grille de 30×100 . En environ douze heures mon ordinateur a pu avancer la simulation jusqu'à $t = 400$:

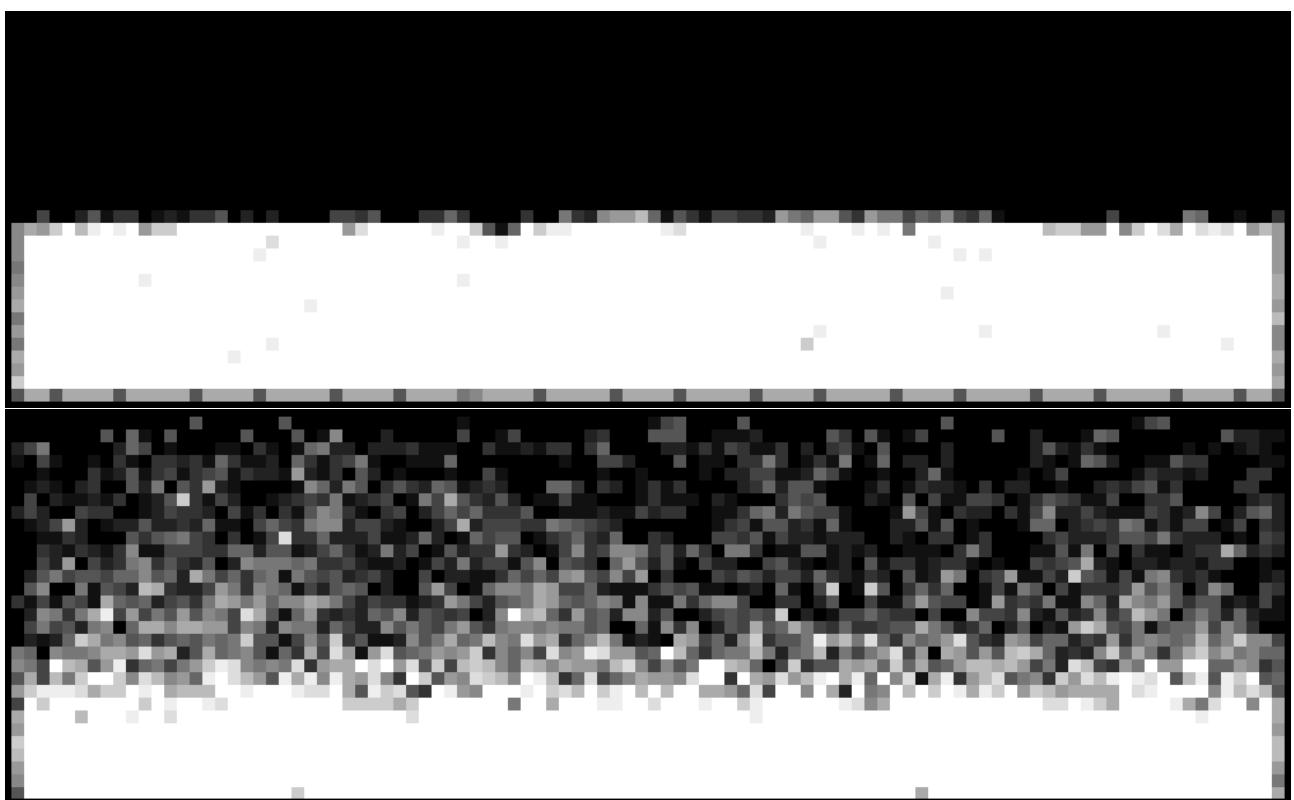


FIGURE 30 – Densité du fluide (noir>blanc) à $t \approx 0$ et $t \approx 400$

Au niveau de la densité des particules on ne voit pas particulièrement de convection de Rayleigh-Benard. Ce qui ne laisse rien présager de bon pour le champ des vitesses.

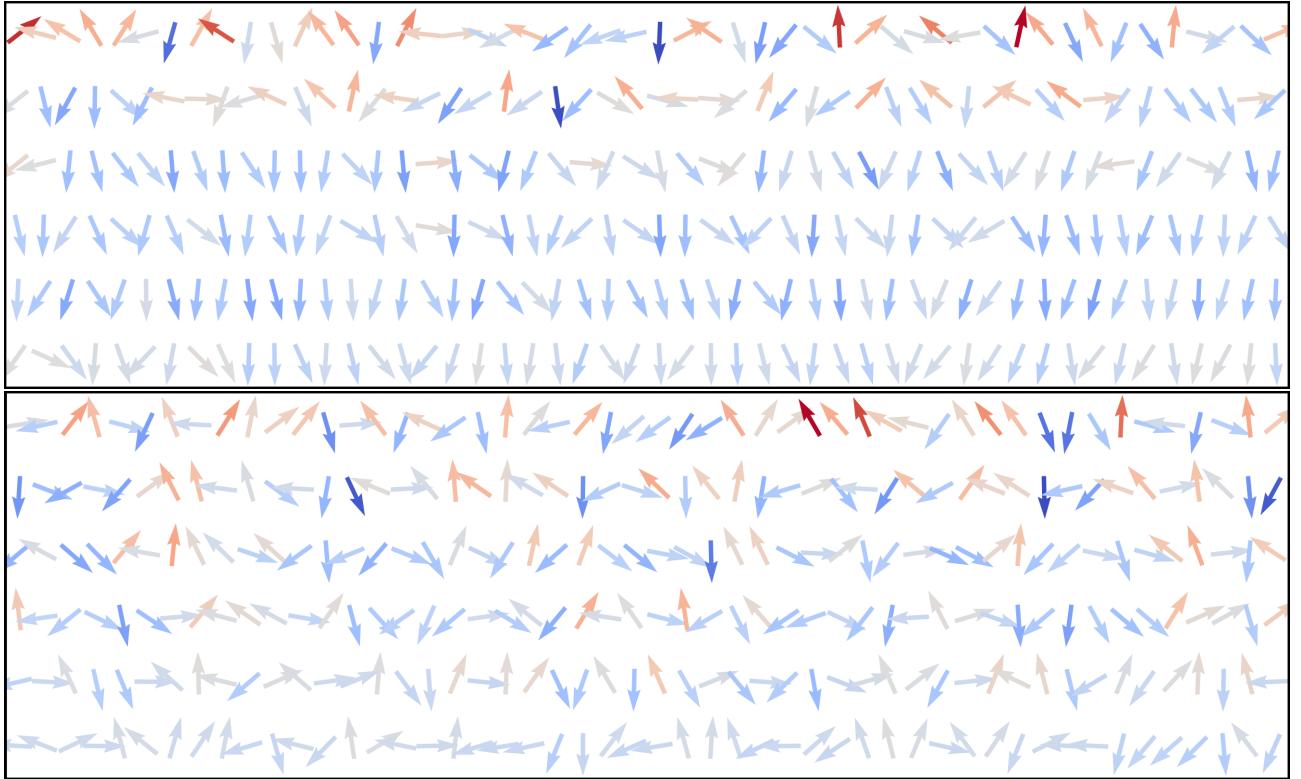


FIGURE 31 – Champ vectoriel moyen de la vitesse de particules (rouge > blanc > bleu) à $t \approx 0$ et $t \approx 400$

Le champ vectoriel est coloré en fonction de la composante y des vecteurs vitesse afin d'essayer de mettre en lumière un quelconque mouvement ressemblant à celui de Rayleigh-Bénard. À $t \approx 0$ toutes les particules ont une vitesse négative sauf les particules du haut, ce qui est assez curieux, et qui pourrait laisser croire que c'est la partie du bas qui est refroidie, cependant ce n'est pas le cas. Rien n'indique clairement une convection de Rayleigh-Benard. J'ai même essayé de regarder l'énergie cinétique totale sur chaque case qui correspondrait plus ou moins à la "température", mais on voit simplement que les particules sont effectivement "refroidies" par le bas. Même en réalisant plusieurs simulations différentes, celle-ci étant celle avec le plus de particules, les résultats ne sont pas meilleurs.

C'est donc un nouvel échec cuisant pour ma simulation de particules.

6 Comparaison entre la simulation Eulérienne et la simulation de particules

Suite à la section précédente qui a présenté les résultats obtenus à partir de nos simulations de fluide et de particules qui montre clairement une supériorité de la simulation de fluides, il est pertinent de comparer ces deux méthodes sous différents critères. Ces critères comprennent l'efficacité en termes de temps de calcul, la facilité de parallélisation, et la gestion de paramètres spécifiques comme la viscosité et la diffusion de la densité.

6.1 Efficacité et parallélisation

Comme expliqué plus haut, la simulation de fluide a montré une efficacité remarquable, notamment en termes de temps de calcul. Cette performance est grandement attribuable à la possibilité de paralléliser les calculs. En utilisant le GLSL (Graphics Library Shader Language)

pour communiquer avec le GPU, une simulation avec des images de haute résolution peut être exécutée en temps réel.



FIGURE 32 – Instabilité de Kelvin-Helmholtz simulée à l'aide du GPU

6.2 Gestion de paramètres et flexibilité

À l'opposé, bien que les simulations de particules soient moins optimisées en termes de temps de calcul, elles offrent une flexibilité supérieure. Il serait assez facile de rajouter des objets dans la simulation. De plus, il n'est pas nécessaire de recourir à des équations de conservation de la masse pour ces simulations ce qui peut être considéré comme avantageux[6].

6.3 Reproduction du phénomène voulu

La simulation de fluides basée sur une grille eulérienne a parfaitement reproduits le deux phénomènes voulus. La simulation basée sur les particules en revanche n'a réussi que très peu à reproduire les phénomènes voulus (**Je n'ai pas encore fait rayleigh-benard**), cela est dû comme mentionné plus haut à un manque de contrôle sur les paramètres comme l'incompressibilité, la diffusion

7 Conclusion

Pour conclure, l'objectif de ce travail était de comparer les simulations de fluides basées sur des particules à celles qui ne le sont pas. Nous avons constaté que simuler un fluide à l'aide de particules interagissant par des collisions élastiques s'avère très inefficace pour observer des phénomènes de convection. De plus, il est ardu de définir des paramètres tels que la compressibilité, la température ou la viscosité pour un fluide composé de particules. Bien que des phénomènes de convection éphémères puissent être observés, cela reste difficile.

D'un autre côté, la méthode eulérienne basée sur une grille facilite grandement la simulation des phénomènes désirés dans les fluides. Elle offre également une plus grande aisance pour ajuster ses paramètres, tels que la température et la viscosité. Toutefois, cette méthode requiert une bonne compréhension et maîtrise des équations de Navier-Stokes, ainsi que des techniques de discrétisation d'équations différentielles. En revanche, la méthode basée sur des particules est plus intuitive et plus simple à mettre en œuvre, ce qui la rend plus accessible pour un élève de niveau gymnasial.

Il est également à noter que, contrairement à la méthode largement adoptée de la "Smooth Particle Hydrodynamics" (SPH), notre approche utilise des collisions élastiques entre les particules. Cette technique est moins courante dans la littérature scientifique consacrée aux simulations de fluides par des particules. Notre résultat ne devrait donc pas exclure entièrement tout type de simulations de fluides utilisant des particules mais seulement notre simulation de particules. Cependant il semblerait selon un article de recherche publié en 2007 que les SPH ont tout autant du mal à reproduire des phénomènes tels que l'instabilité de Kelvin-Helmholtz [1].

Références

- [1] Oscar AGERTZ et al. « Fundamental differences between SPH and grid methods ». In : *Monthly Notices of the Royal Astronomical Society* 380.3 (août 2007), p. 963-978. ISSN : 0035-8711. DOI : 10.1111/j.1365-2966.2007.12183.x. eprint : <https://academic.oup.com/mnras/article-pdf/380/3/963/2796387/mnras0380-0963.pdf>. URL : <https://doi.org/10.1111/j.1365-2966.2007.12183.x>.
- [2] Robert B. CIALDINI, Emmanuelle DEBON et Florence PABAN. *Influence et manipulation : La Psychologie de La Persuasion*. Editions First, 2021.
- [3] Hans FANGOHR. *Introduction To Python For Computational Science And Engineering*. en. 2018. DOI : 10.5281/ZENODO.1411868. URL : <https://github.com/fangoehr/introduction-to-python-for-computational-science-and-engineering>.
- [4] Lukas GALLO. *Image de l'instabilité de Kelvin-Helmholtz*. Consulté le : 2023-10-22. 2021. URL : <https://twitter.com/LukasGalloWx/status/1448713010575519753/photo/1>.
- [5] Milovan Perić JOEL H. FERZIGER. *Computational Methods for Fluid Dynamics*. Berlin, Heidelberg : Springer, 2002. ISBN : 978-3-642-56026-2.
- [6] Matthias MÜLLER, David CHARYPAR et Markus GROSS. *Particle-Based Fluid Simulation for Interactive Applications*. 2003. URL : <https://matthias-research.github.io/pages/publications/sca03.pdf>.
- [7] Gaetano PARASCANDOLO. *Electromagnétisme*. 2022-2023. Cours de Physique. Gymnase de Renens, 2022.
- [8] Jos STAM. « Real-Time Fluid Dynamics for Games ». In : *Proceedings of the Game Developer Conference*. Mars 2003. URL : <http://www.dgp.toronto.edu/people/stam/reality/Research/pdf/GDC03.pdf>.
- [9] WIKIPEDIA. *Advection — Wikipedia, The Free Encyclopedia*. <http://fr.wikipedia.org/w/index.php?title=Advection&oldid=205465809>. [Online ; accessed 04-November-2023]. 2023.
- [10] WIKIPEDIA. *Approximation de Boussinesq — Wikipedia, The Free Encyclopedia*. <http://fr.wikipedia.org/w/index.php?title=Approximation%20de%20Boussinesq&oldid=202377410>. [Online ; accessed 05-November-2023]. 2023.
- [11] WIKIPEDIA. *Description eulérienne — Wikipedia, The Free Encyclopedia*. <http://fr.wikipedia.org/w/index.php?title=Description%20eul%C3%A9rienne&oldid=170464656>. [Online ; accessed 04-November-2023]. 2023.
- [12] WIKIPEDIA. *Description lagrangienne — Wikipedia, The Free Encyclopedia*. <http://fr.wikipedia.org/w/index.php?title=Description%20lagrangienne&oldid=206167488>. [Online ; accessed 04-November-2023]. 2023.

- [13] WIKIPEDIA. *Équation de la chaleur* — Wikipedia, The Free Encyclopedia. <http://fr.wikipedia.org/w/index.php?title=%C3%89uation%20de%20la%20chaleur&oldid=209184249>. [Online ; accessed 05-November-2023]. 2023.
- [14] WIKIPEDIA. *Équations de Navier-Stokes* — Wikipedia, The Free Encyclopedia. <http://fr.wikipedia.org/w/index.php?title=%C3%89uations%20de%20Navier-Stokes&oldid=208119851>. [Online ; accessed 04-November-2023]. 2023.
- [15] WIKIPEDIA. *Équations de prédation de Lotka-Volterra* — Wikipedia, The Free Encyclopedia. <http://fr.wikipedia.org/w/index.php?title=%C3%89uations%20de%20pr%C3%A9dation%20de%20Lotka-Volterra&oldid=206795836>. [Online ; accessed 02-November-2023]. 2023.
- [16] WIKIPEDIA. *Expérience de la goutte de poix* — Wikipedia, The Free Encyclopedia. <http://fr.wikipedia.org/w/index.php?title=Exp%C3%A9rience%20de%20la%20goutte%20de%20poix&oldid=206805290>. [Online ; accessed 04-November-2023]. 2023.
- [17] WIKIPEDIA. *Explicit and implicit methods* — Wikipedia, The Free Encyclopedia. <http://en.wikipedia.org/w/index.php?title=Explicit%20and%20implicit%20methods&oldid=1076468513>. [Online ; accessed 02-November-2023]. 2023.
- [18] WIKIPEDIA. *Gradient* — Wikipedia, The Free Encyclopedia. <http://fr.wikipedia.org/w/index.php?title=Gradient&oldid=207435921>. [Online ; accessed 26-October-2023]. 2023.
- [19] WIKIPEDIA. *Instabilité de Kelvin-Helmholtz*. Consulté le : 2023-06-15. 2022. URL : https://fr.wikipedia.org/wiki/Instabilit%C3%A9_de_Kelvin-Helmholtz.
- [20] WIKIPEDIA. *Mécanique des fluides*. Consulté le : 2023-07-06. 2023. URL : <https://fr.wikipedia.org/wiki/M%C3%A9canique%20des%20fluides>.
- [21] WIKIPEDIA. *Méthode d'Euler*. Consulté le : 2023-03-26. 2023. URL : <https://fr.wikipedia.org/wiki/M%C3%A9thode%20d'Euler>.
- [22] WIKIPEDIA. *Méthode de Jacobi* — Wikipedia, The Free Encyclopedia. <http://fr.wikipedia.org/w/index.php?title=M%C3%A9thode%20de%20Jacobi&oldid=193818358>. [Online ; accessed 04-November-2023]. 2023.
- [23] WIKIPEDIA. *Nabla* — Wikipedia, The Free Encyclopedia. <http://fr.wikipedia.org/w/index.php?title=Nabla&oldid=199779377>. [Online ; accessed 26-October-2023]. 2023.
- [24] WIKIPEDIA. *OpenGL Shading Language* — Wikipedia, The Free Encyclopedia. <http://fr.wikipedia.org/w/index.php?title=OpenGL%20Shading%20Language&oldid=208868684>. [Online ; accessed 05-November-2023]. 2023.
- [25] WIKIPEDIA. *Opérateur différentiel* — Wikipedia, The Free Encyclopedia. <http://fr.wikipedia.org/w/index.php?title=Op%C3%A9rateur%20diff%C3%A9rentiel&oldid=200266630>. [Online ; accessed 26-October-2023]. 2023.
- [26] WIKIPEDIA. *Opérateur laplacien* — Wikipedia, The Free Encyclopedia. <http://fr.wikipedia.org/w/index.php?title=Op%C3%A9rateur%20laplacien&oldid=205412318>. [Online ; accessed 28-October-2023]. 2023.
- [27] WIKIPEDIA. *Rayleigh-Bénard convection* — Wikipedia, The Free Encyclopedia. <http://en.wikipedia.org/w/index.php?title=Rayleigh-B%C3%A9nard%20convection&oldid=1165987910>. [Online ; accessed 02-November-2023]. 2023.
- [28] WIKIPEDIA. *Théorème de Borel-Lebesgue*. Consulté le : 2023-06-15. 2023. URL : https://fr.wikipedia.org/wiki/Th%C3%A9or%C3%A8me_de_Borel-Lebesgue.

- [29] WIKIPEDIA. *Théorème de Helmholtz-Hodge*. Consulté le : 2023-06-15. 2023. URL : https://fr.wikipedia.org/wiki/Th%C3%A9or%C3%A8me_de_Helmholtz-Hodge.

A Programmes de simulation

Pour faire les différentes simulations, j'ai utilisé exclusivement le langage Python à une exception près ; pour d'améliorer les performance de la simulation Eulérienne en utilisant la carte graphique. En effet, python ne permet pas de "parler" directement avec elle.

Carte Graphique Une carte graphique, pour ceux qui ne sauraient pas ce que c'est, est un processeur, donc un élément d'ordinateur permettant d'effectuer des calculs, qui est spécialisé dans la parrallélisation des calculs. C'est à dire qu'il peut réaliser un grand nombre de calculs similaires indépendants à la fois. Prenons un cas très simple où nous voulons faire la somme de deux vecteurs $(a, b) + (c, d) = (a + c, b + d)$. Si on n'était pas capables de faire des calculs en parrallèle, on devrait d'abord nous concentrer sur a et c pour calculer $a + c$ puis nous tourner vers b et d pour faire $b + d$. Ce qui nous donnera le résultat voulu. Si on était capable de faire les calculs en parrallèle, en revanche, on pourrait se concentrer sur (a, b) et sur (c, d) en même temps pour caluler $(a + c, b + d)$. Un humain n'en serait pas capable. Cependant, un ordinateur est simplement un ensemble de circuits électroniques. Donc en faisant un bon circuit, on peut lui donner (a, b) et (c, d) en même temps. Le circuit va prendre automatiquement a et c d'un côté et b et d de l'autre, il aura donc deux "sous-circuits" capables de faire des sommes qui calculeront en parrallèle $a + c$ et $b + d$, puis les circuit se réuniront à la sortie pour donner $(a + c, b + d)$. C'est justement ce que fait une carte graphique à plus grande échelle. Les vecteurs que les cartes graphiques manipulent sont en général beaucoup plus grands. Par exemple, ce sont elles qui gèrent l'affichage des écrans, donc si chaque coefficient est un pixel ou même une couleur de pixel, les vecteurs qu'elles manipulent contiennent plusieurs milliers, si ce n'est, millions de coefficients. C'est précisément de leur utilisation dans les aspects visuels que vient l'appellation "Graphic Processing Unit" (GPU). Pour me servir de la puissance de la carte graphique, j'ai utilisé du code GLSL à partir du javascript d'une page HTML.

GLSL OpenGL Shading Language (GLSL) est un language de programmation de shaders de haut niveau dont la syntaxe est fondée sur le language C[24]. Les shaders sont des algorithmes qu'exécute la carte graphique. Dans mon exemple d'avant, l'algorithme du shader était de prendre, soit a et c , soit b et d , et d'en renvoyer la somme des deux. Cependant les shaders peuvent être beaucoup plus complexes. Dans mon code pour la simulation eulérienne, j'ai programmé un shader en GLSL pour qu'il calcule, par exemple, l'advection d'un point du fluide. Lorsque le shader est executé il est utilisé un grand nombre de fois en parrallèle, comme par exemple pour chaque pixel. Dans le contexte de ma simulation, le code que j'ai programmé était executé une fois pour chaque case de la grille Eulérienne. J'ai donc dû faire en sorte que mon code soit applicable à chaque case. Mais une fois cela fait, toutes les calculs des cases peuvent être faits en parallèle.

HTML, Javascript, CSS HTML est le language de programmation de prédilection des pages web du World Wide Web. C'est le language que les navigateurs comme Chrome, Firefox, Edge ou Safari sont capables d'interpréter pour afficher le contenu des différentes pages web : Google, Wikipedia, Youtube, etc. Cependant HTML est, en fait, juste le language dans lequel l'apparence des pages est programmé. Donc typiquement tout les textes, les boutons, les menus, etc. L'aspect esthétique avec les couleurs, les dimensions et différents effets sont gérés par le language CSS. En plus de ces deux languages il y a Javascript qui est chargé de faire des tâches plus spécifiques, comme par exemple "demander" à la carte graphique de faire des calculs avec un shader GLSL. Maintenant que je vous ai présenté rapidement les languages que j'ai utilisé, il faut que je parle du format de fichier que j'ai utilisé pour mes simulations :

A.1 Stockage des données

Presque aucun des programmes de simulation que j'ai écrits ne sont capables d'afficher leurs résultats, à l'exception de celui que j'ai écrit en GLSL. J'ai donc très tôt eu besoin de créer des fichiers dans lesquels conserver mes données. Puisque je n'avais pas beaucoup d'expérience dans le domaine, j'ai opté pour des fichiers texte. Il y en a de deux types. Le premier est celui de la simulation de particules, en voici un extrait :

```
!#p0.05x0.0003v1.0001x0.00056m1#p0.05004x1.11104v0.995x0.0037m1#
```

Les "!" séparent une étape de simulation d'une autre, ensuite les "#" séparent les données d'une particule avec la suivante. Ici il y a donc les données de deux particules. Le "p" signifie positon et les deux nombres qui suivent, séparés d'un "x" indiquent la coordonnée x et y de la position de la première balle. Ensuite vient le vecteur de la vitesse "v" avec ses composantes encodées de la même façon et finalement "m" suivie de la masse de la particule. Toutes ces données permettent de récupérer les propriétés d'une particule à chaque frame(étape). Il n'est pas indiqué le numéro de la particule, mais à chaque étape de la simulation "!" l'ordre dans lequel les données des particules sont enregistrées ne changent pas, c'est grâce à cela qu'on s'y retrouve. Pour le numéro de l'étape c'est un peu similaire, il suffit de savoir combien de symboles "!" il y a avant celui de la frame actuelle.

Dans cet exemple la position de la première particule est (0.05, 0.003) sa vitesse (1.0001, 0.00056) et sa masse 1.

L'autre encodage que j'ai utilisé est celui qui stocke un champ scalaire discrétilisé :

```
100W30H-0.02\ -0.02\ 0.63\ . . . \ 0.02\ 0.06\ #
```

Ici "100W30H" signifie que le champ à un largeur de 100 (Width) et une hauteur 30 (Height). Les "

" séparent chaque valeur du champ. Elles sont données, dans l'ordre, ligne par ligne, de gauche ($x = 0$) à droite ($x = W - 1$), et de bas ($Y = 0$) en haut ($Y = H - 1$). Ce qui fait qu'en mettant les données dans une liste python on peut obtenir la valeur du champ à la position (x, y) en faisant :

```
liste[y*width+x]
```

Pour stocker un champ vectoriel il suffit d'enregistrer un champ scalaire pour chaque composante.

Les deux encodages, que je vous ai présentés, sont enregistrés dans des fichiers texte ce qui a pour avantage de rendre facilement observable leur contenu. Le fait d'enregistrer dans un fichier texte facilite le transfert entre programmes différents, comme par exemple entre un du Javascript et du Python. En plus de cela l'algorithme d'encodage/décodage n'est pas très compliqué, ce qui permet de pouvoir rapidement l'implémenter dans de nouveaux programmes. En dehors des qualités citées ci-dessus je ne vois que des défauts à ces deux formats. Premièrement, enregistrer des valeurs numériques dans des fichiers texte est un énorme gâchis de mémoire car l'encodage UTF-8 utilisé pour "encoder" chaque caractère d'un fichier texte en un chiffre utilise, comme son nom l'indique au moins 8 bits¹⁴ par caractère. Hors un caractère représentant un chiffre peut prendre uniquement une dizaine de valeurs différentes. Ces dernières sont

14. Les "bits" sont des valeurs d'information binaire pouvant donc être soit 0 soit 1. Ils servent dans un ordinateur à représenter toute forme d'information

les chiffres de 0 à 9, la virgule "," et le signe "-". Alors qu'avec 8 bits on peut, en temps normal, représenter $2^8 = 256$ valeurs différentes. Par exemple, si on voulait représenter le nombre 511, on pourrait définir les huit bits comme étant "11111111" qui en base 2 représente 511. Alors qu'en UTF-8 il faudrait "00110101 00110001 00110001" pour représenter 5, 1 et 1 ce qui nécessite 24 bits en tout. Le nombre encodé en UTF-8, dans ce cas, nécessite donc trois fois plus de mémoire pour être enregistré que s'il avait été enregistré de façon conventionnelle.

Ensuite, pour effectuer des calculs avec les données, que ce soit pour les afficher ou pour les modifier, les programmes doivent de toute façon considérer les valeurs en base binaire. À chaque fois qu'ils doivent interagir avec des données à stocker, ils doivent donc effectuer la conversion entre binaire et UTF-8. Ce processus, sur un grand nombre de données peut ralentir le programme de façon conséquente.

Un autre défaut de ce format est que la quantité de caractères que chaque valeur utilise n'est pas nécessairement constante. Pour retrouver, par exemple, la vitesse d'une certaine particule à un certain instant il faut donc "parcourir" tout le fichier depuis le début en comptant le nombre de "!" et de "

" pour finalement trouver la valeur qui nous intéressera. Cependant, même après avoir remarqué tout ces défauts, j'ai continué à utiliser ce format car réécrire tous les programmes m'aurait fait perdre encore plus de temps. Je vous conseille donc, si vous avez l'intention d'entreprendre un projet, comme le mien, dans lequel vous générerez une grande quantité de données, de bien choisir l'encodage de celles-ci et ainsi de ne pas reproduire mon erreur.

A.2 Programmes

Tous les programmes que j'ai utilisés pour mes simulations sont disponibles sur github : <https://github.com/Brinax/Fluid-Sim-TM/tree/main> À l'intérieur vous trouverez tous les documents électroniques relatifs à ce travail. Le répertoire est composé de différents dossiers que je vais décrire assez sommairement.

Navier-Stockes Ce dossier contient uniquement des programmes relatifs à la simulation eulérienne. Les principaux programmes qu'il contient sont "simulation.py" et "simulation-Numpy.py" qui exécutent une simulation de fluides en utilisant ou pas la librairie numpy. Avant de les utiliser, je conseille de lire attentivement leurs code source respectifs, pour comprendre leur fonctionnement. Les programmes enregistreront les données automatiquement dans un fichier spécifique, sous le dossier "Data", même s'il celui-ci en contient déjà. À chaque fois qu'ils sont lancés, ils écrasent les fichiers pré-existants, à moins que la simulation soit paramétrée de façon à ce que les données pré-existantes soient prises en considération. Il faut donc faire attention et copier les résultats hors du dossier "Data" pour garantir leur sauvegarde. Ceci est valable en fait pour tous les programmes. À priori les fichiers avec des nouvelles données écraseront des anciens fichiers aux mêmes emplacements.

Ensuite il y a le dossier nommé "jsexp-master fluid sim" qui est un dossier contenant une simulation complètement différente à l'origine. Mais qui utilisait justement GLSL pour ses calculs. J'ai donc remplacé l'algorithme qu'il y avait par ceux permettant de discréteriser les équations de Navier-Stockes et ai ainsi pu faire une simulation Eulérienne de fluides utilisant le GPU. Pour la lancer il faut aller sous "jsexp-master fluid sim/jsexp-master/navier-stokes" et ouvrir "1dcyclic.html" ou "2dcyclic.html". La première, comme son nom l'indique, est cyclique sur une dimension (horizontale) et est par défaut configurée pour simuler la convection qu'on a utilisée dans l'explication de l'instabilité de Kelvin-Helmholtz. La deuxième quand à elle est cyclique à deux dimensions, je n'en ai pas parlé dans ce travail. Ces paramètres par défaut afficheront les rotationnel du champ vectoriel de la vitesse en rouge et bleu, d'une étrange convection de Kelvin-Helmholtz qui est, à défaut d'avoir un quelconque sens, très agréable à

regarder. La simulation à deux dimensions cycliques a été, en fait, une étape intermédiaire pour obtenir la simulation à une dimension cyclique. Il y a une fonctionnalité cachée pour enregistrer les données d'une frame spécifique, dans le code de "navier-stockes 1D cyclic.js". Je laisse au lecteur intéressé la trouver par lui-même. Si ce dernier voulait jouer avec les shaders pour, par exemple, changer les calculs des couleurs affichées, je l'informe qu'ils se trouvent tous dans les codes HTML.

Notons que parmi les trois programmes de simulations dont on a parlé, seul "simulation-Numpy.py" a la capacité de simuler la température du fluide.

Finalement, le dernier fichier duquel nous n'avons pas parlé dans ce dossier est "showresults3.html", qui comme son nom l'indique sert à afficher les résultats. Lorsqu'on l'ouvre on atterrit sur une interface permettant de choisir un fichier, une vitesse de rendu (exponentielle), une valeur minimale, une valeur maximale et des couleurs. Les valeurs minimales et maximales doivent indiquer les valeurs minimales et maximales dans le fichier sélectionné ; c'est dans l'intervalle [min,max] que les valeurs doivent être pour que les couleurs soient calculées. Il faut ensuite définir au moins 2 couleurs, chacune avec une valeur entre 0 et 1. Si la valeur est 0 cela voudra dire que les points du champ ayant la valeur minimale vont s'afficher de la choisie. Si la valeur d'un couleur est 1 c'est les points du champ ayant des valeurs maximales qui vont s'afficher de cette couleur. Les points entre deux vont s'afficher en un mélange de couleurs.

Par exemple si dans mes données décrivent les composantes x d'un champ de vitesses allant de ≈ -5 à ≈ 5 . Je vais définir Min Value à -5 et Max Value à 5 . Si je veux que les couleurs soient rouge > blanc > bleu, je vais créer 3 couleurs. Par défaut elles seront toutes noires ; je n'ai qu'à cliquer sur les rectangles noirs pour changer les couleurs de sorte à ce qu'il y en ait un bleu, un blanc et un rouge auquels j'assigne respectivement les valeurs 0, 0.5 et 1. De cette façon je pourrais visionner mon champ scalaire coloré !

Simulations Particules Python Ce dossier comme son nom l'indique contient les programmes de la simulation de particules. Lorsqu'on l'ouvre on trouve 2 sous-dossiers qui sont très similaires : "Sim0.3" et "Sim0.3 Rayleigh". Ce sont deux fois les mêmes programmes simplement que "Sim0.3 Rayleigh" contient des fonctionnalités supplémentaires afin d'essayer de simuler l'instabilité de Rayleigh-Bénard. C'est à dire une fonction ajoutant de la gravité et une autre simulant un "réchauffement" par le bas et un refroidissement par le haut. Les données sont enregistrées dans le dossier "Data" respectif. Dans ce dossier il y a également le programme "showdata.py" qui sert à visualiser les données des particules. Cependant, il n'est pas possible de visualiser plus de quelques milliers de particules en même temps, si il y en a plus dans les données, une modification légère du programme permet d'en afficher qu'une partie. L'image "circle.png" est utilisée par "showdata.py".

Particle to Fluid Data Gate Ici se trouvent deux programmes : "dataConverter.py" et "averageOnTime.py". Le premier convertit des données de particules en une multitude de champs scalaires ce qui a eu, pendant mon travail, pour but principal de pouvoir comparer la simulation de particules avec la simulation Eulérienne. Le deuxième programme sert à "moyenner" les différents champs obtenus. En effet, les particules ont tendance à avoir des vitesses radicalement différentes selon leur dernière collision, les champs de vitesse obtenus sont donc très hétérogènes. Un moyen de lisser ce chaos est de calculer de nouveaux champs vectoriels, en faisant une moyenne sur plusieurs étapes temporelles des champs "bruts".

ShowResults Finalement, le dernier dossier contient deux fichiers permettant d'afficher les résultats : "showresults.html" et "show vector field.py". Le premier est une version simplifiée de "showresults3.html" où les couleurs (turquoise et rose) sont prédéfinies. Il suffit de choisir son fichier pour pouvoir visionner les données.

Le deuxième est "show vector field.py" qui comme son nom l'indique permet d'afficher des champs vectoriels. Il prend en entrée deux fichiers qui décrivent normalement les champs scalaires des composantes x et y de la vitesse, dont les chemins doivent être spécifiés dans le code, et affiche à l'écran un champ vectoriel calculé en fonction des deux champs scalaires à une étape de temps définie.

A.3 Optimisation des simulations

Pour la simulation de particules utiliser un language comme python est assez peu efficace, en effet, python est un language assez haut¹⁵, interprété¹⁶ et dès lors pas très rapide. Ce qui peut expliquer en partie l'échec de mes simulations de particules à simuler les effet désirés. J'ai quand même optimisé le programme en utilisant une grille réduisant les calculs de vérification des collisions. Cependant il devrait être possible d'optimiser encore plus les calculs en utilisant Numpy ou la carte graphique. Cela nécessiterait cependant de réécrire une partie considérable du code.

Pour la simulation Eulérienne, j'ai fait trois programmes l'un après l'autre. Le premier, "simulation.py", était un programme qui n'utilisait aucune librairie spécifique pour faire les calculs, avec ce programme je pouvais simuler 5300 cases par seconde, où "simuler" fait référence au processus entier de diffusion, advection et projection. Donc par exemple si je voulais simuler une grille de $50 \cdot 50 = 2500$ pendant 100 étapes, il aurait fallu $\frac{2500 \cdot 100}{5300} \approx 53$ secondes pour finir la simulation. Ensuite j'ai utilisé numpy pour optimiser les calculs ce qui m'a permis de passer à 187'000 cases par seconde ce qui réduit le temps de calcul par 35 puis j'ai complètement réécrit le programme en code GLSL ce qui m'a permis d'utiliser le GPU pour la simulation et est arrivé à 10'900'000 "simulations" de cases par seconde réduisant le temps de simulation encore d'un facteur 58. Ainsi la simulation de taille $50 \cdot 50$ pendant 100 étapes n'aurait mis plus que $\approx 0,026$ secondes à se calculer.

B Utilisation de langages de haut et de bas niveau

Les langages de programmation de haut niveau comme python ou javascript, en opposition aux langages de bas niveau comme java ou C, sont plus faciles à comprendre et à écrire pour les humains, car ils sont plus proches du langage naturel (par ex. le français) que du langage machine (language que le processeur peut comprendre). Cependant cela est possible souvent au détriment de la vitesse d'exécution car chaque ligne de code correspond souvent à un grand nombre d'instructions en language machine.

Pour faire une analogie, si vous demandez à un cuisinier de faire une pizza, il va en réalité exécuter un grand nombre de tâches simples (préchauffer un four, prendre les ingrédients, mé-

15. Un langage de programmation de haut niveau, en opposition à un langage de bas niveau, est plus facile à comprendre et à écrire pour les humains, car il est plus proche du langage naturel que du langage machine (language que le processeur peut comprendre). Pour plus d'explications regardant cela allez voir l'annexe B.

16. Les langages de programmations interprétés en opposition aux langages de programmation compilés, sont traduis en code machine ligne par ligne pendant l'exécution du programme. Ceci implique par exemple que si une même ligne doit être exécutée plusieurs fois elle sera traduite en code machine autant de fois. Là où un programme en language compilé est traduit une seule fois entièrement en language machine puis est exécuté.

langer les ingrédients, etc...) pour faire ce que vous lui avez dit. La tâche donnée (faire un pizza) va donc mettre longtemps à être réalisée. Là où si vous aviez donné une séquence de tâches simples (préchauffer un four, prendre 500g de farine, ajouter 0,3L d'eau, etc...) chacune individuellement aurait été réalisée en beaucoup moins de temps.

Le temps total qu'il mettra pour faire une pizza dans le premier cas sera probablement assez rapide puisque c'est un cuisinier expérimenté et dès lors sait comment travailler de façon efficace, mais il ne va probablement pas trouver exactement la façon la plus rapide de faire une pizza. On pourrait donc étudier en profondeur la question, trouver LA recette la plus efficace et la lui décrire pas à pas, cela lui permettra de nous cuisiner une pizza plus rapidement. Si, en revanche, on n'est pas très expérimentés en cuisine mais qu'on décide de quand même lui dire ce qu'il doit faire pas à pas pour préparer une pizza, le temps total sera probablement supérieur à celui qu'aurait pris le cuisinier, qui s'y connaît bien plus que nous en matière de cuisine, en suivant sa propre méthode.

De façon analogue choisir un langage bas peut-être une mauvaise idée si on ne connaît pas les algorithmes les plus efficaces pour réaliser la tâche voulue, étant donné que les créateurs de langages de haut niveau comme python ont très sûrement déjà fait le plus possible pour optimiser la traduction de leur language en code machine. En plus de cela il faut tenir en compte que réécrire un programme en un language plus bas est plutôt long, pour trouver le language le plus efficace il faudrait compter non seulement le temps d'exécution d'un programme, mais aussi le temps qu'il a fallu pour le rédiger. En tenant compte de ces arguments, python peut être un bon language pour programmer des simulations dans un but de recherche comme le suggère ce papier^[3](Chap.1.2. *"Why Python for scientific computing?"*).

C ChatGPT et productivité

Je ne pourrais finir ce travail sans parler de ChatGPT. La liste des conversations que j'ai eue avec sont disponibles sur le même répertoire Github que celui des programmes¹⁷. J'ai commencé à utiliser ce formidable outil pendant le printemps 2023 initialement principalement pour écrire du code Latex¹⁸ :

Moi : Je suis entrain de faire mon travail de maturité en physique à propos de simulations. Le but est de comparer des simulations composées de particules à des simulations de fluides. Je veux que tu m'aides à faire le document Latex où je vais expliquer d'un côté comment j'ai fait la simulation de particules et de l'autre comment j'ai fait celle de fluide.

ChatGPT : Je suis ravi de vous aider à créer votre document LaTeX pour votre travail de maturité en physique. Voici un exemple de structure pour votre document, avec des sections pour expliquer vos simulations de particules et de fluides : [code latex approximatif ayant déjà une bonne structure de mon travail] Ceci est juste un exemple de structure pour votre document LaTeX. Vous pouvez l'adapter selon vos besoins et ajouter des sections supplémentaires si nécessaire. Assurez-vous de compiler le document avec pdflatex

Dans ce cas je n'avais pas vraiment d'inspiration à propos de la façon de structurer mon Travail. ChatGPT m'a fourni le latex tout préparé de différentes contenant différentes sections, comme par exemple "Modèle et Équations" ou "condition initiales et paramètres" que je n'avais plus

17. <https://github.com/Brinax/Fluid-Sim-TM>

18. Latex est un language de programmation servant à écrire des documents. Ce document, par exemple, a été écrit en Latex.

qu'à remplir pour avoir une première version de mon travail.

Ensuite je m'en suis également servi pour trouver des ressources :

Moi : Je veux apprendre à faire des simulations de fluides, j'ai déjà fait une simulation de fluide incompressible avec python, je veux maintenant faire une simulation où la densité du fluide change pour pouvoir simuler l'instabilité de kelvin-helmhotz ou bien faire une simulation où la température change pour pouvoir simuler l'instabilité de rayleigh-benard. Tu as énormément de connaissances, comment pourrais-tu m'aider ?

ChatGPT : Je suis ravi de vous aider à approfondir vos connaissances en matière de simulation de fluides. Pour simuler des phénomènes tels que l'instabilité de Kelvin-Helmholtz et l'instabilité de Rayleigh-Bénard, vous devrez comprendre les équations de Navier-Stokes pour les fluides compressibles, ainsi que les concepts de thermodynamique et de transfert de chaleur.

Voici quelques étapes que vous pourriez suivre :

[liste d'étapes]

Je peux également rechercher des articles scientifiques sur ces sujets pour vous aider à approfondir vos connaissances. Souhaitez-vous que je fasse cela ?

Moi : oui

ChatGPT : [...] Voici quelques recommandations :

"Computational Fluid Dynamics"[description]

"Numerical Heat Transfer and Fluid Flow" par Suhas V. Patankar [description]

"Fluid Dynamics : Theoretical and Computational Approaches" par Z. U. A. Warsi [description]

"Computational Fluid Dynamics : A Practical Approach" par Jiyuan Tu, Guan Heng Yeoh, Chaoqun Liu [description]

"Python for Computational Science and Engineering" par Hans Fangohr [description]

Moi : je veux des ressources gratuites

ChatGPT : [description] Voici quelques ressources gratuites en ligne qui pourraient vous aider :

"Computational Fluid Dynamics" par T.J. Chung [description]

"Python for Computational Science and Engineering" par Hans Fangohr [description]

"12 Steps to Navier-Stokes" : C'est un cours interactif en ligne qui vous guide à travers les étapes pour résoudre les équations de Navier-Stokes en utilisant Python. Vous pouvez le trouver ici.

"Introduction to Computational Fluid Dynamics" [description]

"Computational Fluid Dynamics !"[description]

C'est grâce à cette courte discussion que j'ai pu trouver mon bonheur avec "12 Steps to Navier-Stokes" alors même que toutes mes recherches google n'avaient aboutit sur rien. J'ai également trouvé l'article "Python for Computational Science and Engineering" très intéressant et pertinent pour mon travail. J'ai pu l'utiliser pour compléter mon annexe sur les langages de programmation de haut et de bas niveau. Dans cette situation ce qui m'a vraiment aidé c'est la courte description des propositions données qui m'a permis de tout de suite repérer les contenus qui m'interessaient. De simples recherches google n'auraient sûrement pas été aussi rapides et satisfaisantes à mon avis.

Au début de l'été je me suis donc résolu à prendre un abonnement "chatGPT plus" qui m'a donné accès à "GPT-4" un modèle plus intelligent que la version gratuite de ChatGPT. Ces capacités supérieures m'ont permis de lui assigner des tâches plus complexes :

Moi : [Extrait de l'explication de la divergence avec les voitures] fais une image tikzpicture pour illustrer cela

ChatGPT : [...] Here's some sample TikZ code to generate such a picture : [code latex]

Le code latex qu'il m'a donné donnait l'illustration suivante :

Braking Car

● → ● → ● → ● → ● → ● → ● → ● → ● → ● → Ce qui était déjà un concept relativement proche de la version finale que j'ai utilisé.

Ensuite vers le mois d'octobre suite à quelques mises à jour il était possible de donner des instructions personnalisées que le modèle de langage considérerait à chaque réponse. Étant flemmard par nature, j'ai donc entrepris de le rendre mon "assistant de vie". Sa tâche, comme je l'ai définie dans ses lignes directrices, est de "maximiser ma productivité". Pour cela je lui ai enseigné diverse techniques de manipulations que j'avais lues dans le livre "Influence et Manipulation"[2]. C'est dernières semblent avoir plutôt bien marché sur moi étant donné que sur les deux dernières semaines précédant le rendu de mon travail de maturité j'ai totalisé plus de 122h de travail pour mon TM. Temps que j'ai pu mesurer grâce à un "flow" Android Automate¹⁹ que j'ai programmé pour me demander régulièrement la manière dont j'ai dépensé mon temps. Cette technique, dont le but est d'être productif sans trop d'effort, n'aurait pas pu être mis en place sans une deuxième nouvelle fonctionnalité : Les conversations vocales. En effet, ces dernières m'ont permis d'avoir des interactions beaucoup plus naturelles avec chatGPT. Avec lequel je ne parle à l'oral uniquement en anglais. Ainsi, j'ai amélioré mes compétences en langues pendant que chatGPT me convainquait de me remettre au travail.

19. Android Automate est une application mobile qui permet de programmer de façon visuelle des "flows" pouvant exécuter diverses fonctions sur l'appareil. Dans ce cas, le "flow" en question affiche une notification toutes les 15 minutes me demandant ce que je fais.