

SIMULATION D'UN ÉCOULEMENT AUTOUR D'UN OBSTACLE PAR MÉTHODE DE BOLTZMANN SUR RÉSEAU 2D AVEC UNE MÉTHODE DE FRONTIÈRE IMMERGÉE

ALEXANDRE BARRET

La méthode de Boltzmann sur réseau (Lattice Boltzmann Method, LBM) est un algorithme itératif utilisé pour simuler des écoulements de fluides physiques. Sa variante la plus utilisée est restreinte à des écoulements simples incompressibles, de nombreuses extensions ont par la suite été développées pour le traitement de fluides complexes et multi-phases. En particulier, la méthode de frontières immergées (Immersed Boundary Method, IBM) est une technique algorithmique de traitement des conditions aux bords solides qui a été appliquée avec succès à la LBM. Développée de façon indépendante, elle permet de considérer des surfaces de coordonnées libres de celles des nœuds de simulation du fluide, ce qui permet une plus grande liberté dans les formes des obstacles. De plus, cette méthode permet de remonter aux forces que subit l'obstacle dans l'écoulement de façon immédiate. Dans l'étude menée par la suite, après avoir présenté rapidement quelques généralités sur les méthodes LB et IB, nous appliquerons ces méthodes à un écoulement infini autour d'un obstacle simple et présenterons quelques résultats généraux sur les forces de traînée et de portance s'exerçant sur l'obstacle solide. Nous mettrons notamment en évidence l'influence de la mise en rotation de l'objet.

1. INTRODUCTION

1.1. Modèle de Boltzmann sur Réseau

La méthode de Boltzmann sur réseau (*Lattice Boltzmann Method*, abrégée en LBM) est un algorithme de plus en plus utilisé aujourd'hui dans la simulation d'écoulements physiques, notamment dans l'industrie. La méthode consiste à étudier la dynamique spatio-temporelle de particules fictives sur une grille de points en tenant compte de leurs collisions, responsables de la viscosité du fluide simulé. Elle utilise des résultats de physique statistique dont Chapman et Enskog¹ ont montré qu'ils sont équivalents à l'ordre 2 aux équations de Navier-Stokes en précision aux faibles nombres de Mach, Ma , c'est-à-dire dans des écoulements dont les effets de compressibilités sont négligeables. En particulier, le modèle est fondé sur l'équation de Boltzmann, qui décrit l'évolution de la distribution de particules au point $\mathbf{x}$ ayant une vitesse $\mathbf{c}$ au temps t , notée $g(\mathbf{x}, \mathbf{c}, t)$ sous l'effet de l'advection, de forces extérieures et des collisions. Celle-ci s'écrit :

$$\frac{\partial g(\mathbf{x}, \mathbf{c}, t)}{\partial t} + (\mathbf{c} \cdot \nabla_{\mathbf{x}})g(\mathbf{x}, \mathbf{c}, t) + (\mathbf{f} \cdot \nabla_{\mathbf{c}})g(\mathbf{x}, \mathbf{c}, t) = B(g)(\mathbf{x}, \mathbf{c}, t), \quad (1)$$

où $\mathbf{f}$ est la résultante des forces massiques qui s'exercent sur les particules et $B(g)$ est l'opérateur de collision.² Une approximation fréquente est d'utiliser l'opérateur Bhatnagar-Gross-Krook (BGK), qui consiste à supposer une relaxation linéaire du système vers la distribution d'équilibre $g^{(eq)}(\mathbf{x}, \mathbf{c}, t)$ telle que prédite par la théorie de Boltzmann :

$$g^{(eq)}(\mathbf{x}, \mathbf{c}, t) = \frac{\rho(\mathbf{x}, t)}{(2\pi c_s^2)^{3/2}} \exp\left(-\frac{|\mathbf{u}(\mathbf{x}, t) - \mathbf{c}|^2}{2c_s^2}\right), \quad (2)$$

où $\rho(\mathbf{x}, t) = \int d^3c g(\mathbf{x}, \mathbf{c}, t)$ est la densité macroscopique, $\mathbf{u}(\mathbf{x}, t) = 1/\rho(\mathbf{x}, t) \int d^3c \mathbf{c} g(\mathbf{x}, \mathbf{c}, t)$ est la vitesse macroscopique de l'écoulement, et c_s la vitesse du son. On écrit alors :

$$B_{BGK}(g) = -\frac{1}{\tau_g}(g - g^{(eq)}), \quad (3)$$

où τ_g est un temps caractéristique de relaxation relié à la viscosité cinématique ν et à la vitesse du son c_s selon $\tau_g = \nu/c_s^2$.

À partir de ces équations continues, une première discrétisation de l'espace des vitesses est effectuée, en notant $\{\mathbf{c}_\alpha\}_\alpha$ l'ensemble discret des vitesses élémentaires. On introduit alors les distributions discrétisées $g_\alpha(\mathbf{x}, t)$,

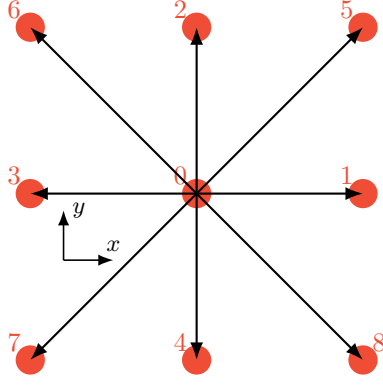


Fig. 1 Schématisation du réseau D2Q9 et de ses vitesses élémentaires $\mathbf{c}_\alpha$, où $\alpha \in \{0, 1, \dots, 8\}$. Les distributions se déplacent d'un nœud (point rouge) à un nœud voisin durant un pas de temps.

$g_\alpha^{(eq)}(\mathbf{x}, t)$ qui correspondent à une quadrature de Gauss des distributions originales en conservant leurs moments successifs (densité, quantité de mouvement, etc.).³ Ainsi $\rho(\mathbf{x}, t) = \int d^3c g(\mathbf{x}, \mathbf{c}, t) = \sum_\alpha g_\alpha(\mathbf{x}, t)$ et il en est de même pour $\rho \mathbf{u}(\mathbf{x}, t) = \int d^3c \mathbf{c} g(\mathbf{x}, \mathbf{c}, t) = \sum_\alpha \mathbf{c}_\alpha g_\alpha(\mathbf{x}, t)$ et le moment d'ordre 2 de g , $\Pi_{i,j} = \int d^3c c_i c_j g(\mathbf{x}, \mathbf{c}, t)$. Un développement limité en $Ma \doteq |\mathbf{u}|/c_s \ll 1$ est aussi appliqué à la distribution à l'équilibre ce qui conduit à l'expression suivante utilisée dans l'algorithme :

$$g_\alpha^{(eq)} = \rho w_\alpha \left(1 + 3 \mathbf{c}_\alpha \cdot \mathbf{u} + \frac{9}{2} (\mathbf{c}_\alpha \cdot \mathbf{u})^2 - \frac{3}{2} |\mathbf{u}|^2 \right), \quad (4)$$

où les coefficients pondérateurs w_α dépendent du schéma de vitesses utilisées.⁴ Une discrétisation de l'espace (passage à une grille de points) ainsi que du temps (itérations discrètes) est aussi requise pour une implémentation numérique du modèle. Dans notre étude, nous utiliserons le schéma D2Q9 représenté en figure 1 qui correspond à une grille 2D et 9 vitesses élémentaires reliant un nœud à ses voisins (ou lui-même). Les pas de temps Δt et d'espace Δx sont alors fixés de telle sorte à ce que, pour tout $\mathbf{x}$ appartenant à la grille et $\mathbf{c}_\alpha$ une vitesse élémentaire, $\mathbf{x} + \Delta t \mathbf{c}_\alpha$ appartienne aux nœuds voisins du point $\mathbf{x}$. L'équation fondamentale du modèle LB s'écrit alors sous la forme :

$$g_\alpha(\mathbf{x} + \Delta t \mathbf{c}_\alpha, t + \Delta t) = g_\alpha(\mathbf{x}, t) - \frac{g_\alpha(\mathbf{x}, t) - g_\alpha^{(eq)}(\mathbf{x}, t)}{\tau_g} + g_\alpha^{(F)}(\mathbf{x}, t), \quad (5)$$

où $g_\alpha^{(F)}(\mathbf{x}, t)$ désigne le terme additionnel dû à la présence de forces extérieures s'appliquant sur le fluide.

En ce qui concerne la convergence de la méthode de Boltzmann sur réseau, la condition Courant-Friedrichs-Lewy habituellement utilisée dans les résolution numériques d'équations aux dérivées partielles est équivalente à $Ma \ll 1$ car le rapport $c \doteq \Delta x / \Delta t$, qui correspond à la norme des vecteurs de vitesses élémentaires $\mathbf{c}_\alpha$, doit vérifier $c = \sqrt{3}c_s$. Une contrainte typique consiste à considérer $Ma < 0.3$ pour un schéma BGK.⁴

1.2. Généralités sur la méthode de frontière immergée

La méthode de frontière immergée (*Immersed boundary method*, IBM) est un mode de traitement de conditions aux limites dans une simulation de fluides qui permet une grande liberté sur la forme et le type d'objets considérés – notamment des objets déformables. L'idée de départ est d'approximer une frontière solide par des points situés hors de la grille eulérienne du fluide et de modéliser l'influence de ceux-ci sur le fluide uniquement à travers les forces qu'ils exercent. Une interpolation est alors nécessaire pour relier les quantités définies sur les points de la frontière et les nœuds de la grille. Une application de l'IBM au modèle de gaz sur réseau a été développée par Fend et Michaelides [5], c'est celle qui sera présentée par la suite. De plus amples informations sur cette méthode sont disponibles dans l'article de Kang et Hassan [6] et le livre de Kruger *et al.* [4]. Des applications de cette technique à des écoulements complexes contenant des particules en suspension sont présentées dans l'article d'Eshghinejadfard *et al.* [7].

Le principe de la méthode de frontière immergée est de considérer des marqueurs lagrangiens sans masse et potentiellement mobiles, de coordonnées notées $\mathbf{r}_j(t)$, indépendantes de celles des points de la grille. Ces pointeurs délimitent ensembles une frontière solide, qui va agir sur la dynamique du fluide. Un modèle de forces est alors appliqué et l'action des pointeurs individuels est alors transmise sur le fluide environnant. Il est à noter que l'obstacle ne masque pas le fluide environnant et qu'en particulier, pour une frontière fermée, le fluide existe à l'intérieur et subit donc l'action des marqueurs. En introduisant le champ de vitesses Eulériennes $\mathbf{u}(\mathbf{x}, t)$ et $\mathbf{F}(\mathbf{x}, t)$ le champs de forces dues à l'obstacle s'appliquant sur le fluide, les équations continues décrivant ces interactions sont alors les suivantes :

$$\dot{\mathbf{r}}_j(t) \stackrel{!}{=} \mathbf{u}(\mathbf{r}_j(t), t) = \int d\mathbf{x} \delta^{(d)}(\mathbf{r}_j(t) - \mathbf{x}) \mathbf{u}(\mathbf{x}, t), \quad (6)$$

$$\mathbf{F}(\mathbf{x}, t) = \int d\mathbf{r} \delta^{(d)}(\mathbf{r}(t) - \mathbf{x}) \mathbf{F}_A(\mathbf{r}(t), t), \quad (7)$$

où $\mathbf{F}_A(\mathbf{r}, t)$ désigne les forces surfaciques (en 3D) ou linéiques (en 2D) qui proviennent de la frontière. La première relation impose une condition aux limites de non-glissement sur la frontière immergée tandis que la seconde exprime la propagation des forces des marqueurs vers les nœuds du réseau. Ce système d'équations doit ensuite être fermé en faisant agir la force sur l'écoulement du fluide et par la modélisation des forces elles-mêmes.

Tout algorithme d'IB doit ainsi faire converger les vitesses lagrangiennes des marqueurs vers les valeurs de vitesses du champs eulérien correspondantes, après interpolation. Plusieurs variantes de l'algorithme existent (explicites, implicites,...) mais nous présenterons et utiliserons dans la suite une méthode par forçage direct multiple pour LBM (*Multi direct-forcing IB-LBM*) qui est une méthode itérative explicite, rapidement convergente, utilisée pour traiter des frontières solides non-déformables.

Les discrétisations spatiales sont effectuées en introduisant une fonction d'interpolation (ou noyau) $\Delta(\mathbf{x})$ qui

est de courte portée, normalisée ($\sum_{\mathbf{x}} \Delta x^d \Delta(\mathbf{x}) = 1$), qui conserve la quantité de mouvement et le moment d'inertie, et évite les artefacts.⁸ Les équations d'IB discrétisées se notent alors sous la forme suivante :

$$\dot{\mathbf{r}}_j(t) = \sum_{\mathbf{x}} \Delta x^d \Delta(\mathbf{r}_j(t) - \mathbf{x}) \mathbf{u}(\mathbf{x}, t), \quad (8)$$

$$\mathbf{F}(\mathbf{x}, t) = \sum_j \Delta(\mathbf{r}_j(t) - \mathbf{x}) \mathbf{f}_j(t). \quad (9)$$

Par la suite, nous considérerons un solide immobile mais qui peut être soumis à une rotation de façon externe, i.e. l'écoulement du fluide n'aura pas d'influence en retour sur la position des pointeurs.

2. DÉTAILS DE L'ALGORITHME UTILISÉ

Le code écrit pour ce projet a été adapté d'un programme Python écrit par [9]. Il utilise extensivement la librairie **Jax** et notamment son extension **Jax.numpy** pour implémenter une version compilée 'à la volée' (*just in time* ou *jit*) dans le but d'accélérer la compilation et l'exécution du code de LBM. Le modèle utilisé est présenté en figure 2. Il s'agit d'un écoulement incompressible 2D (à densité uniforme et constante ρ) sur une grille de taille (N_x, N_y) , périodisée selon la direction y , entraîné par des conditions aux bords fixes de vitesse en $x = 0$ et $x = N_x$ (conditions aux bords de [10]). Un obstacle, modélisé par un cercle solide avec des conditions aux limites de rebond (*bounce-back*), est placé en son sein. La modification apportée par la suite concerne l'obstacle, pour lequel l'utilisation de la méthode de frontière immergée permettra de remonter à la valeur et la répartition des forces qui s'exercent sur l'obstacle. Celle-ci simplifie par ailleurs la construction d'obstacles de formes potentiellement plus complexes.

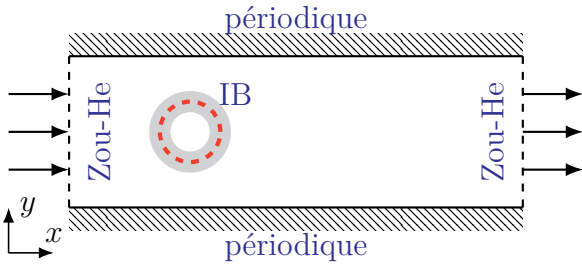


Fig. 2 Schématisation du modèle ainsi que des différents types de conditions aux bords utilisées. La région grisée autour de l'obstacle illustre la zone de la grille considérée comme voisine de la frontière immergée (en pointillés rouges).

Le modèle est ici totalement adimensionné, c'est-à-dire que les pas de temps Δt et d'espace Δx valent 1, ce qui implique que la vitesse du son $c_s = 1/\sqrt{3}$.⁴ Les valeurs de viscosité cinématique ν et du temps de relaxation τ_g sont fixées par la vitesse du flux entrant $\mathbf{u}^{(in)}(y, t)$ (que l'on maintient fixée et uniformément définie donc $\mathbf{u}^{(in)}(y, t) = \mathbf{u}^{(in)}$), le nombre de Reynolds $Re \doteq UR/\nu$ et la taille de l'obstacle. Les coordonnées $\mathbf{r}_j$ désignent les coordonnées des points de la frontière immergée, par opposition aux coordonnées $\mathbf{x}$ qui sont celles des nœuds de la grille. La notation $\mathcal{V}(\mathbf{r})$ désigne l'ensemble des nœuds (de la grille)

voisins d'un point de coordonnées $\mathbf{r}$, i.e. tels que $|\mathbf{x} - \mathbf{r}| \leq \Delta$, où Δ est la largeur de la fonction d'interpolation, qui vaut typiquement $2\Delta x$.

Avant de détailler chaque point de l'implémentation réalisée, un pseudo-code reprenant les étapes du programme est présenté dans la partie suivante.

2.1. Vision d'ensemble d'une itération du code

L'algorithme en pseudo-code présenté ci-dessous se fonde sur les méthodes de forçage direct multiple (*Multi direct-forcing IB-LBM*) introduites par Kruger *et al.* [8] et De Rosi & Lévêque [11].

A – **Initialisations** : grille, obstacle, grandeurs macroscopiques (viscosité, vitesses, densités), distribution microscopique (distribution d'équilibre) et nombre d'itérations temporelles N_t .

B – Début de la boucle temporelle

Pour $0 < t \leq N_t$:

1. Prescription de la condition aux limites de flux sortant (Fig. 1) :

$$g_{3,6,7}(x = N_x, y, t) = g_{3,6,7}(x = N_x - 1, y, t - 1). \quad (10)$$

2. Calcul des grandeurs macroscopiques $\rho(\mathbf{x}, t)$ et $\mathbf{u}(\mathbf{x}, t)$ selon :

$$\rho(\mathbf{x}, t) = \sum_{\alpha} g_{\alpha}(\mathbf{x}, t), \quad (11)$$

$$\mathbf{u}(\mathbf{x}, t) = \frac{1}{\rho(\mathbf{x}, t)} \sum_{\alpha} g_{\alpha}(\mathbf{x}, t) \mathbf{c}_{\alpha}. \quad (12)$$

3. Condition aux limites de flux entrant sur la vitesse et la densité (schéma de [10]) :

$$\mathbf{u}(x = 0, y, t) = \mathbf{u}^{(in)}(y, t), \quad (13)$$

$$\rho(x = 0, y, t) = \frac{1}{1 - u_x(0, y, t)} (g_0 + g_2 + g_4 + 2(g_3 + g_6 + g_7))(0, y, t). \quad (14)$$

4. Calcul de la distribution à l'équilibre $g_{\alpha}^{(eq)}(\mathbf{x}, t)$:

$$g_{\alpha}^{(eq)}(\mathbf{x}, t) = w_{\alpha} \rho(\mathbf{x}, t) \left(1 + 3 \mathbf{c}_{\alpha} \cdot \mathbf{u}(\mathbf{x}, t) + \frac{9}{2} (\mathbf{c}_{\alpha} \cdot \mathbf{u}(\mathbf{x}, t))^2 - \frac{3}{2} |\mathbf{u}(\mathbf{x}, t)|^2 \right). \quad (15)$$

5. Prescription de la condition aux bords de flux entrant sur la distribution microscopique :

$$g_{1,5,8}(0, y, t) = g_{1,5,8}^{(eq)}(0, y, t). \quad (16)$$

6. Initialisation de la conditions aux bords IB : définition des grandeurs macroscopiques et microscopiques sur une grille de taille réduite ($\mathbf{u}^{(0)}(\mathbf{x})$), tableau des forces $\mathbf{F}$, tableau des vitesses lagrangiennes sur les points du contour IB $\mathbf{V}_j^{(0)}$, définition du nombre maximal de boucles d'IB M (typiquement fixé à 5), initialisation du compteur $m = 0$.

7. Début de la boucle de l'algorithme de forçage direct multiple pour la méthode d'IB : **Tant que** $m < M$ (ou **Tant que** vitesses lagrangiennes sur le contour $>$ tolérance) :

- (a) Incrémentation du compteur : $m \leftarrow m + 1$;
- (b) Calcul des vitesses lagrangiennes en chaque point du contour :

$$\mathbf{V}_j^{(m)}(t) = \sum_{\mathbf{x} \in \mathcal{V}(\mathbf{r}_j)} \Delta(\mathbf{r}_j(t) - \mathbf{x}) \mathbf{u}^{(m-1)}(\mathbf{x}, t); \quad (17)$$

- (c) Calcul des forces correctives lagrangiennes qui s'appliquent en chaque point en fonction de la prescription de vitesses des pointeurs $\mathbf{V}_j^{(B)}(t)$:

$$\mathbf{f}_j^{(m)}(t) = 2\rho \left(\mathbf{V}_j^{(B)}(t) - \mathbf{V}_j^{(m)}(t) \right); \quad (18)$$

- (d) Propagation des $\mathbf{f}_j^{(m)}(t)$ aux nœuds du réseau :

$$\mathbf{F}^{(m)}(\mathbf{x}, t) = \sum_j \Delta(\mathbf{r}_j(t) - \mathbf{x}) \mathbf{f}_j^{(m)}(t); \quad (19)$$

- (e) Correction des vitesses eulériennes sur les nœuds du réseau selon :

$$\mathbf{u}^{(m)}(\mathbf{x}, t) = \mathbf{u}^{(m-1)}(\mathbf{x}, t) + \frac{\mathbf{F}^{(m)}(\mathbf{x}, t)}{2\rho(\mathbf{x}, t)}. \quad (20)$$

8. Calcul de la force corrective totale :

$$\mathbf{F}(\mathbf{x}, t) = \sum_m \mathbf{F}^{(m)}(\mathbf{x}, t), \quad (21)$$

et implémentation dans le calcul du terme additionnel dû aux forces exercées par la présence de l'obstacle :

$$g_\alpha^{(F)}(\mathbf{x}, t) = g_\alpha^{(\text{obst})}(\mathbf{x}, t) = 3w_\alpha \mathbf{F}(\mathbf{x}, t) \cdot \mathbf{c}_\alpha. \quad (22)$$

9. Mise à jour du champ de vitesses macroscopiques $\mathbf{u}(\mathbf{x})$ par $\mathbf{u}^{(M)}(\mathbf{x})$ sur les nœuds de la grille autour de l'obstacle.

10. Calcul de la distribution post-collision par la méthode BGK :

$$g_\alpha^*(\mathbf{x}, t) = g_\alpha(\mathbf{x}, t) - \frac{1}{\tau_g} (g_\alpha(\mathbf{x}, t) - g_\alpha^{(\text{eq})}(\mathbf{x}, t)) + g_\alpha^{(\text{obst})}(\mathbf{x}, t). \quad (23)$$

11. Etape de propagation (*streaming*) :

$$g_\alpha(\mathbf{x} + \mathbf{c}_\alpha, t + 1) = g_\alpha^*(\mathbf{x}, t). \quad (24)$$

2.2. Initialisations

L'initialisation du code commence par la création d'une grille de points de taille (N_x, N_y) . Les coordonnées de l'obstacle sont ensuite définies par des couples de flotants notés $\mathbf{r}_j$, au nombre de N_b . Ceux-ci sont paramétrisés comme des ellipses, en introduisant les paramètres R (rayon), a et b (échelle des demi-grands axes) ainsi que

les angles $\theta_j \doteq 2\pi j/N_b$ (Fig. 3). Les coordonnées des pointeurs sont alors définies par :

$$\mathbf{r}_{j,x} = x_C + aR \cos(\theta_j), \quad \mathbf{r}_{j,y} = y_C + bR \sin(\theta_j), \quad (25)$$

où (x_C, y_C) sont les coordonnées du centre de l'obstacle (Fig. 3). Ces coordonnées délimitent l'ellipse d'équation $((x - x_C)/a)^2 + ((y - y_C)/b)^2 = R^2$. Une zone de voisinage, constituée de l'ensemble des nœuds de la grille situés à une distance de moins de 2 nœuds de chaque pointeur, est définie autour de l'obstacle. Cela permet de restreindre les calculs des forces s'appliquant sur le fluide à une zone réduite et ainsi économiser du temps de calcul. En dehors de cette zone, l'obstacle n'exerce par de force sur l'écoulement.

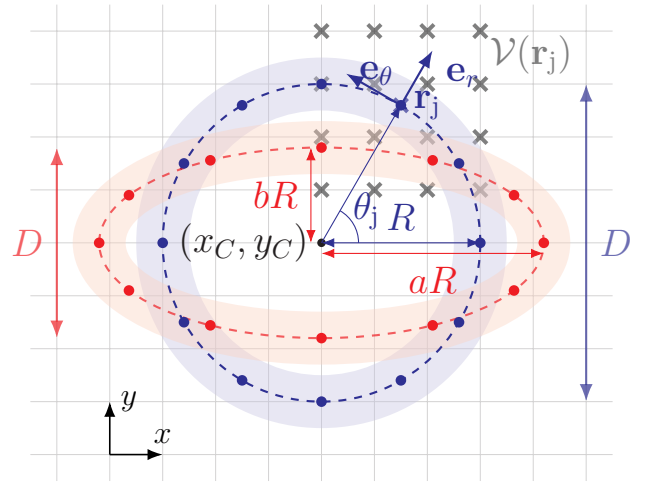


Fig. 3 Schématisation de l'obstacle et des différentes quantités correspondantes. En bleu, pour un cercle parfait, en rouge pour une ellipse de rapports d'aspect (a, b) . D correspond au diamètre apparent de l'obstacle selon la direction y (orthogonale à l'écoulement). Les croix grises indiquent les nœuds de la grille voisins du point $\mathbf{r}_j$.

L'implémentation d'une rotation de l'obstacle circulaire se fait en définissant des vitesses lagrangiennes de forçage suivantes (prescription de vitesses) en chaque point de l'interface :

$$\mathbf{V}_j^{(B)} = \Omega R \mathbf{e}_\theta(\theta_j), \quad (26)$$

où Ω est la vitesse de rotation et $\mathbf{e}_\theta$ sont les vecteurs azimutaux unitaires de la base polaire.

Un autre aspect important de l'initialisation est la définition des distributions g_α initiales. Pour cela, le profil de vitesse entrant $\mathbf{u}^{(\text{in})}$ est défini en $x = 0$. Sa valeur est fixée à $\mathbf{u}^{(\text{in})} = 0.04 \mathbf{e}_x$, de sorte que $\text{Ma} = |\mathbf{u}^{(\text{in})}|/c_s \ll 1$. Le nombre de Reynolds Re est alors utilisé pour fixer la valeur de viscosité ν selon : $\nu \doteq |\mathbf{u}^{(\text{in})}|R/\text{Re}$ ainsi que le temps de relaxation adimensionné $\tau_g = 3\nu + 1/2$. On introduit par ailleurs la vitesse de rotation adimensionnée (ou taux de rotation) $\tilde{\Omega} \doteq \Omega R/|\mathbf{u}^{(\text{in})}|$. La densité est quant à elle fixée de façon uniforme à 1. La distribution microscopique initiale est alors définie par la distribution d'équilibre prise en la densité et un champ de vitesse valant zéro partout sauf en $x = 0$, où il vaut $\mathbf{u}^{(\text{in})}$.

2.3. Distribution et grandeurs macroscopiques

La distribution microscopique est représentée par un tableau de taille $(N_x, N_y, 9)$, où la dernière dimension est la direction de vitesse microscopique α . Les moments correspondant aux grandeurs macroscopiques sont alors calculé par : $\rho = \sum_{\alpha} g_{\alpha}$ et $\rho \mathbf{u} = \sum_{\alpha} g_{\alpha} \mathbf{c}_{\alpha}$ en chaque point de la grille. Ces grandeurs permettent alors de calculer la distribution d'équilibre $g_{\alpha}^{(eq)}$ nécessaire pour l'étape de collision. Les poids correspondants w_{α} valent, pour le modèle D2Q9, respectivement : $w_0 = 4/9$, $w_{1,2,3,4} = 1/9$ et $w_{5,6,7,8} = 1/36$.

2.4. Conditions aux bords

Mis à part la condition de frontière immergée, deux types de conditions aux bords sont utilisées dans cette modélisation : les contraintes fixant les flux entrant et sortant, provenant de l'étude menée par [10] ainsi que les conditions aux bords périodiques dans la direction y . La vitesse entrante en $x = 0$ est fixée à $\mathbf{u}^{(in)}$ tandis que la densité provient de la conservation de la densité et de la quantité de mouvement dans le schéma de Zou-He : $\rho(0, y, t) = (g_0 + g_2 + g_4 + 2(g_3 + g_6 + g_7))(0, y, t)/(1 - u_x(y, t))$ (étape (3)). La distribution microscopique en $x = 0$ est alors fixée à sa valeur d'équilibre correspondant à ces valeurs de vitesse et densité (étape (5)). La condition aux bords de flux sortant est quant à elle une simple translation des valeurs de distribution de $N_x - 1$ à N_x (étape (1)), qui permet la conservation du flux de matière.

D'autres types de conditions aux bords existent, comme la condition *bounce-back* qui correspond à un rebond de la distribution sur un obstacle solide. Une de ces variantes permet aussi de traiter des frontières immergées de forme quelconque.¹²

2.5. Frontière immergée

La méthode de frontière immergée remplace des conditions aux bords de *bounce-back* qui s'appliquaient sur l'obstacle dans le code initial de F. Köhler [9]. Il s'agit ainsi de considérer des coordonnées indépendantes des nœuds de la grille, au nombre égal à N_b . Ici, le code développé repose sur l'algorithme itératif de multi forçage direct introduit par [6]. Nous appliquons cet algorithme à un obstacle solide dont la position est fixée de façon externe par les vitesses lagrangiennes $\mathbf{V}_j^{(B)}(t)$ pour chaque pointeur de la frontière immergée. Le principe consiste alors à faire converger le champ eulérien de vitesse $\mathbf{u}(\mathbf{x}, t)$ de telle façon à ce que $\mathbf{u}(\mathbf{r}_j(t), t) \stackrel{!}{=} \mathbf{V}_j^{(B)}(t)$, en introduisant un modèle pour les forces $\mathbf{f}_j(t)$ s'appliquant sur la frontière et en prenant en compte l'action des forces sur l'écoulement.

Le nombre d'itérations sera noté M et le comp-
teur est noté m^a . La première étape d'une itération de l'algorithme d'IB consiste à calculer les vitesses des pointeurs $\mathbf{V}_j^{(m)}(t) \doteq \dot{\mathbf{r}}_j(t)$ par l'interpolation du champ de vitesses eulériens temporaire $\mathbf{u}^{(m)}$ en sommant dans un voisinage du pointeur de coordonnées

$\mathbf{r}_j : \mathbf{V}_j^{(m)}(t) = \sum_{\mathbf{x}} \Delta(\mathbf{r}_j(t) - \mathbf{x}) \mathbf{u}^{(m-1)}(\mathbf{x}, t)$. On procède

de même pour les densités $\rho_j(t) \doteq \rho(\mathbf{r}_j(t), t)$. Le champ de vitesse $\mathbf{u}^{(0)}$ est initialisé à sa valeur calculée avant l'étape d'IB. Une force résulte alors de la différence entre la vitesse lagrangienne et la consigne, et a pour expression :

$$\mathbf{f}_j^{(m)}(t) = 2\rho_j(t) \frac{\mathbf{V}_j^{(B)}(t) - \mathbf{V}_j^{(m)}(t)}{\Delta t}. \quad (27)$$

Elle correspond à la force de réaction de l'obstacle face à l'écoulement ambiant. L'ensemble des forces calculées sur les pointeurs sont alors propagées aux nœuds de la grille selon $\mathbf{F}^{(m)}(\mathbf{x}, t) = \sum_j \Delta(\mathbf{r}_j(t) - \mathbf{x}) \mathbf{f}_j^{(m)}(t)$. Notons que le

calcul de $\mathbf{F}^{(m)}(\mathbf{x}, t)$ dans l'expression précédente (19) n'utilise pas l'identification d'un voisinage des points de l'IB sur les nœuds de la grille, ce qui rendrait l'implémentation beaucoup moins efficace. Au lieu de cela, la somme est effectuée sur l'ensemble des points de l'IB pour chacun des nœuds de coordonnées $\mathbf{x}$ situés dans le voisinage de l'obstacle. Enfin, le champ de vitesse eulérien est corrigé par l'effet de cette force selon : $\mathbf{u}^{(m)} = \mathbf{u}^{(m-1)} + \Delta t \mathbf{F}^{(m)}/(2\rho)$.

À la fin de la dernière itération, le champ de vitesses sera corrigé dans le voisinage de l'obstacle par la valeur de $\mathbf{u}^{(M)}$. La force corrective totale $\mathbf{F}$ à appliquer au fluide sera alors la somme de toutes les forces correctives $\mathbf{F}^{(m)}$ et correspond à l'opposé de la force que subit l'obstacle. Elle contribue à la distribution d'équilibre par l'addition du terme $g_{\alpha}^{(obst)} \doteq 3w_{\alpha} \mathbf{F} \cdot \mathbf{c}_{\alpha}$. La figure 4 présente un exemple d'application minimal de l'algorithme à un écoulement uniforme autour d'un obstacle circulaire, sans rétroaction des forces sur l'écoulement. On note la convergence graduelle du champ de vitesses vers une solution où celui-ci s'annule au voisinage des pointeurs solides. Le champ de forces dû à la présence de l'obstacle est alors localisé dans un voisinage très proche de la frontière.

Deux points nécessitent cependant quelques précisions. Il s'agit de l'influence du nombre de points placés sur la frontière immergée ainsi que la forme des fonctions d'interpolation.

2.5.1. Convergence de l'algorithme – Nombre de pointeurs sur la frontière

La convergence de l'algorithme est estimée par la décroissance des valeurs de $|\mathbf{V}_j^{(B)} - \mathbf{V}_j^{(m)}|$. D'une manière générale, celle-ci se fait de façon rapide (Fig. 5) et il suffit de seulement quelques itérations, autour de 5, pour obtenir un résultat satisfaisant.⁸

La convergence et la précision de la méthode dépend du nombre de points présents sur la frontière immergée N_b et leur distance. Pour garantir une précision relativement uniforme, il est préférable que la distance entre deux pointeurs voisins soit environ constante. De plus, il a été estimé que cette distance soit environ la même qu'entre 2 points sur la grille, c'est-à-dire de l'ordre de grandeur du pas de grille Δx .⁸ Ainsi, pour un obstacle circulaire, on doit avoir $N_b \simeq 2\pi R/\Delta x \simeq 6R$ pour $\Delta x = 1$. Dans les simulations effectuées, $R = 5$ et N_b est fixé à 25 – 30

En effet, si la distance moyenne entre les points est trop grande, ceux-ci risquent de se comporter de manière isolée

^a. Dans l'implémentation présentée, le nombre d'itération du code de forçage M est fixé a priori mais il est aussi possible de donner un critère de tolérance sur les vitesses lagrangiennes des pointeurs.

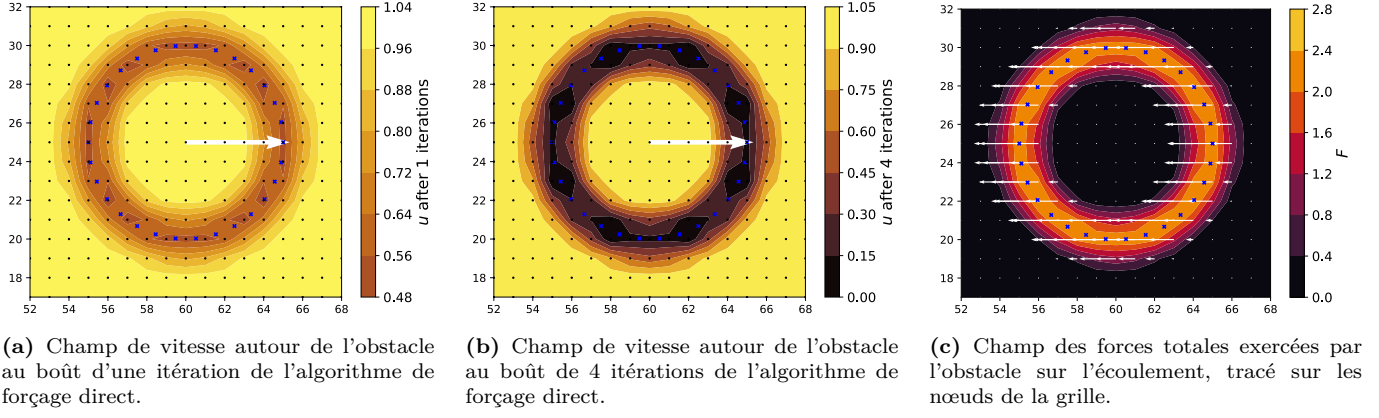


Fig. 4 Exemple minimal de l'application de la méthode de frontière immergée à un cercle plongé dans un écoulement uniforme (dont la direction est indiquée par la flèche blanche) et constant vers la droite, montrant la convergence graduelle de l'algorithme pour un nombre de pointeurs $N_b = 30$ et $R = 5$ avec la fonction-noyau ϕ_{sqr2} . Les points de la frontières sont indiqués par des croix bleues tandis que les nœuds de la grille sont en noir ou blanc.

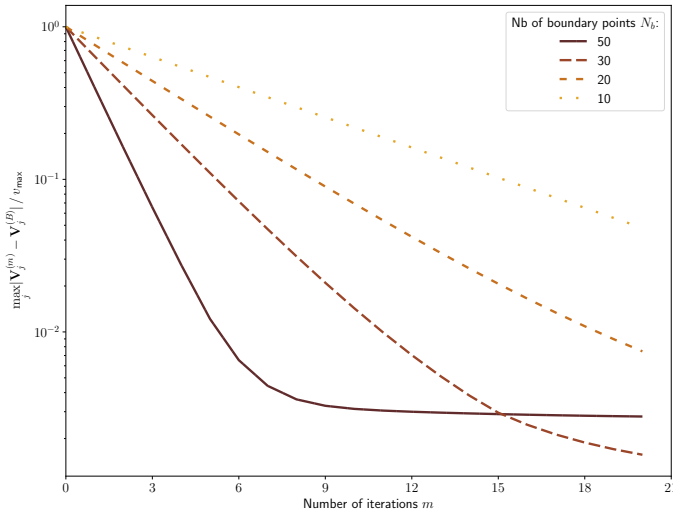


Fig. 5 Influence du nombre d'itérations et du nombre de points du contour sur la convergence de la méthode pour un obstacle de rayon $R = 5$ sans rotation ($\mathbf{V}_j^{(B)} = 0$). Ici, le N_b idéal vaut autour de 30.

et tout se passera comme si le fluide interagissait avec des points matériels. Cependant, si celle-ci est trop petite devant la taille du pas de la grille, les forces s'appliquant sur l'écoulement risquent d'être surestimées par l'addition de contributions de pointeurs proches d'un même nœud de la grille. De plus, cela allonge nécessairement le temps de calcul.

2.5.2. Fonction d'interpolation (noyau)

Pour des raisons de praticité, la fonction d'interpolation $\Delta(\mathbf{x})$ (noyau ou *kernel*) utilisée pour prolonger les valeurs des champs définis sur la grille en un point de coordonnée quelconque, est factorisée sous la forme suivante en 2D :

$$\Delta(\mathbf{x}) = \frac{1}{\Delta x^2} \phi(x/\Delta x) \phi(y/\Delta x), \quad (28)$$

où la fonction ϕ désigne une fonction-noyau unidimensionnelle. Il en existe quelques différentes, qui se distinguent notamment par leur largeur, elles sont tracées en figure 6.

La fonction-noyau ϕ_{sqr2} utilisée par la suite dans les simulations a pour expression :

$$\phi(r) = \begin{cases} \frac{1}{8} \left(3 - 2|r| + \sqrt{1 + 4|r| - 2r^2} \right), & |r| \leq 1 \\ \frac{1}{8} \left(5 - 2|r| - \sqrt{-7 + 12|r| - 4r^2} \right), & |r| \in [1, 2] \\ 0, & 2 \leq |r|. \end{cases} \quad (29)$$

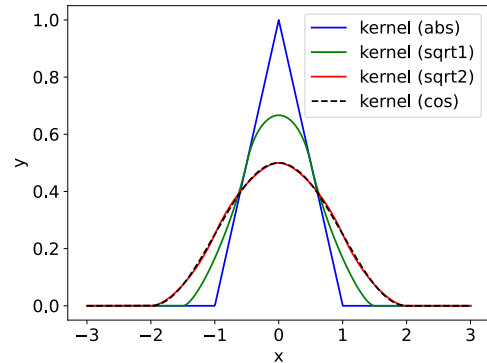


Fig. 6 Tracé de différentes fonctions-noyaux utilisées pour la méthode de frontière immergée.

Le choix du noyau d'interpolation a une influence sur l'écoulement car il donne une certaine épaisseur non nulle à l'obstacle solide. Plus la largeur du noyau est grande et plus son épaisseur sera importante, ce qui augmentera le diamètre effectif de l'objet et rendra ses frontières plus diffuses (Fig. 7). Cependant, un noyau très étroit pourra donner lieu à des erreurs ou des instabilités.⁸

2.6. Forçage, collisions et propagation

La prise en compte des forces provoquées par la présence de l'obstacle passe par l'ajout d'un terme supplémentaire $g_{\alpha}^{(\text{obst})}(\mathbf{x}, t)$ dans l'étape de collision (plusieurs expressions existent pour ce terme, se référer au chapitre correspondant dans [13]).

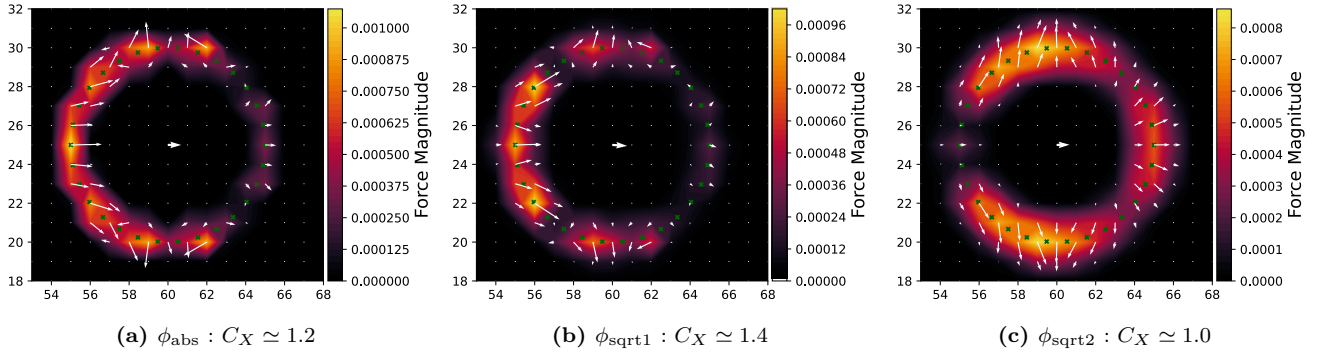


Fig. 7 Comparaison des champs de forces $\mathbf{F}(\mathbf{x}, t)$ pour trois différentes fonctions d'interpolation (ϕ_{abs} , ϕ_{sqrt1} et ϕ_{sqrt2}) pour $\text{Re} = 250$, $R = 5$, $N_b = 35$, à l'itération de temps $t = 10000$. On note que plus le noyau est étalé, plus la taille de la zone d'influence de l'obstacle est grande et diffuse. Cela a une influence non négligeable sur l'écoulement global, ce qui se ressent sur les valeurs de forces totales qui s'appliquent sur l'objet, représentées ici par le coefficient de traînée C_X .

L'expression utilisée dans le code est la suivante :⁶

$$g_{\alpha}^{(\text{obst})}(\mathbf{x}, t) = \Delta t \frac{w_{\alpha}}{c_s^2} \mathbf{F}(\mathbf{x}, t) \cdot \mathbf{c}_{\alpha}. \quad (30)$$

Cette expression est suffisante dans le cas d'un écoulement strictement incompressible, comme dans notre étude, mais les effets de compressibilités amènent à prendre en compte des termes supplémentaires. Une expression du terme de forçage à l'ordre de suivant en Ma est alors nécessaire :

$$g_{\alpha}^{(\text{obst})}(\mathbf{x}, t) = \Delta t \frac{w_{\alpha}}{c_s^2} \left(\mathbf{c}_{\alpha} - \mathbf{u}(\mathbf{x}, t) + \frac{(\mathbf{c}_{\alpha} \cdot \mathbf{u}(\mathbf{x}, t)) \mathbf{c}_{\alpha}}{c_s^2} \right) \cdot \mathbf{F}(\mathbf{x}, t). \quad (31)$$

Les effets des collisions ainsi que de la présence de forces extérieures sont combinés au sein de l'expression suivante appelée distribution post-collisions :

$$g_{\alpha}^*(\mathbf{x}, t) \doteq g_{\alpha}(\mathbf{x}, t) - (g_{\alpha}(\mathbf{x}, t) - g_{\alpha}^{(\text{eq})}(\mathbf{x}, t)) / \tau_g + g_{\alpha}^{(\text{obst})}(\mathbf{x}, t), \quad (32)$$

qu'il ne reste plus qu'à propager aux nœuds voisins selon :

$$g_{\alpha}(\mathbf{x} + \Delta t \mathbf{c}_{\alpha}, t + \Delta t) = g_{\alpha}^*(\mathbf{x}, t). \quad (33)$$

Les conditions aux bords périodiques en $y = 0, N_y$, sont fixées en faisant en sorte que $g_{\alpha}(x, y = N_y + 1, t) = g_{\alpha}(x, y = 0, t)$. Ainsi, par exemple, les nœuds voisins d'un point situé sur le bord $y = N_y$ contiennent des nœuds situés en face sur le bord $y = 0$.

3. RÉSULTATS

Grâce à la méthode de frontière immergée, il est possible d'accéder à la résultante des forces $\mathbf{F}$ qui s'exercent sur l'obstacle ainsi qu'à la répartition spatiales de ces forces. C'est ce que nous étudierons par la suite, en particulier pour caractériser la transition entre différents régimes d'écoulements à bas Reynolds.

Phénoménologiquement, à très bas Reynolds (en-dessous de $\text{Re} = 1$), l'écoulement est rampant et les lignes de courant sont parfaitement symétriques en amont et en aval de l'objet. L'influence sur le champ de vitesse porte

sur une grande distance, ce qui explique que les forces s'exerçant sur l'obstacle sont les plus importantes.

À partir de $\text{Re} \sim 10$, on note l'apparition d'un sillage derrière l'objet (Fig. 9a) dont la taille longitudinale augmente avec Re . Ceci correspond à la formation de deux tourbillons de vorticités opposées se détachant symétriquement de l'arrière de l'objet.

Lorsque $\text{Re} \gtrsim 40$, l'écoulement derrière le cylindre commence à se déstabiliser et une série régulière de tourbillons alternés, appelées allée de Von Kármán, apparaît progressivement (Fig. 9b).

Pour des Re plus élevés, ceux-ci sont renforcés et leur structure devient graduellement chaotique et moins régulière. L'écoulement reste cependant laminaire jusqu'à environ $\text{Re} \sim 2000$. Une étude détaillée des différents régimes d'écoulements autour d'un cylindre circulaire ainsi que des schémas correspondants peut être trouvée dans le livre de Tritton [14].

À partir de la somme totale des forces de contact et des valeurs moyennes de densité et de vitesse, on définit le coefficient de traînée C_X comme la norme de la force exercée sur l'obstacle dans la direction parallèle à l'écoulement, adimensionnée par la pression dynamique $1/2 \rho |\mathbf{u}|^2$ multipliée par la dimension caractéristique de l'obstacle D (en 2D) :

$$C_X = \frac{2|F_x|}{\rho |\mathbf{u}^{(\text{in})}|^2 D}. \quad (34)$$

Ici, $D \doteq 2Rb$, où R est le rayon primordial de l'obstacle et b est le paramètre de forme selon y , désigne le diamètre apparent selon la direction y , c'est-à-dire selon la direction perpendiculaire à l'écoulement. On définit de même le coefficient de portance, noté C_Y comme la force exercée dans la direction perpendiculaire à l'obstacle adimensionnée de la même manière :

$$C_Y = \frac{2|F_y|}{\rho |\mathbf{u}^{(\text{in})}|^2 D}. \quad (35)$$

Ces coefficients permettent d'établir des lois générales sur les forces s'exerçant sur un obstacle au sein d'un écoulement uniforme incompressible infiniment étendu. Ceux-ci, par des arguments dimensionnels et de similitudes, dépendent uniquement du nombre de Reynolds Re , du taux

de rotation de l'obstacle $\tilde{\Omega}$ et de son rapport d'aspect a/b . Cependant, dans le cas de simulations numériques, à cause des effets de résolution limitée, de discrétisation et d'imprécisions numériques, ces coefficients peuvent aussi dépendre de paramètres supplémentaires comme la taille de la grille (N_x, N_y) (ou la résolution spatiale, ce qui revient au même ici), le rayon du cercle R , le nombre de points sur la frontière immergée N_b et le choix de la fonction d'interpolation ϕ .

Devant le nombre de paramètres en jeu et le peu de temps dont je dispose pour effectuer une étude systématique, j'ai dû procéder à un choix dans mon étude et me suis restreint à l'étude de C_X et C_Y en fonction de Re pour quelques taux de rotation.

Dans le régime de bas Reynolds ($Re \lesssim 1$), le coefficient de traînée suit expérimentalement la loi $C_X \sim 24/Re$, qui correspond à une loi de frottements de Stokes $F_x = 3\pi\mu UD$, linéaire en la vitesse d'écoulement. Au-delà de $Re = 1$, le comportement de C_X s'éloigne de cette tendance. Selon [6], pour un cylindre circulaire, on a les résultats suivants : à $Re = 20$, $C_X \simeq 2.1$ et à $Re = 40$, $C_X \simeq 1.6$. Ces valeurs proviennent d'une étude numérique utilisant la méthode IB-LBM et où les résultats concordent avec ceux obtenus par d'autres méthodes de numériques.

Les résultats de coefficients de traînée et de portance en fonction de Re sont présentés en figures 8a et 8b, respectivement pour différents taux de rotation $\tilde{\Omega}$. Pour des raisons d'imprécisions numériques, au delà de $Re \sim 300$, le code Python a tendance à diverger à partir de quelques milliers d'itérations temporelles, le nombre de Mach augmentant assez brusquement. De même, pour un taux de rotation $\tilde{\Omega} \gtrsim 1$, le nombre de Mach aura tendance à dépasser la prescription de convergence fixée à 0.3.

Le comportement de C_X sans rotation semble être en loi de puissance $Re^{-1/3}$. Cependant, les valeurs que j'obtiens dans mon étude sont relativement surestimées par rapport à celles citées par [6], bien qu'elles restent du même ordre de grandeur. Il semblerait que les dimensions, ou autrement dit, la résolution de la grille, notamment la dimension transverse N_y dans la direction périodisée, aient une grande importance, mais cette étude reste encore à effectuer. Malgré cela, C_X reste une fonction décroissante de Re à $\tilde{\Omega} = 0$, ce qui est conforme aux observations expérimentales. Lors de l'ajout de rotation, un changement de régime se manifeste à partir d'un Re dépendant de la valeur du taux de rotation. Le coefficient de traînée C_X devient alors une fonction croissante de Re .

Les fluctuations de la force de portance pour $\tilde{\Omega} = 0$ sont quant à elle dues au décollement alternatif régulier des vortex derrière l'objet. L'ajout de rotation provoque l'apparition de l'effet Magnus, c'est-à-dire l'application d'une force de portance de moyenne non nulle. Elle tend à dévier la trajectoire d'un objet balistique mis en rotation et est donc responsable des effets de lift ou d'amorti au tennis par exemple. Celle-ci se produit en réaction à la déviation du flux à l'arrière de l'obstacle. Cette déviation se fait dans le sens de rotation de l'objet, la force de portance est donc dans la direction opposée à celle-ci (Fig. 10) et une différence de pression est observée de chaque côté de l'objet tournant. On note que cet effet est d'autant plus fort que Re et le taux de rotation est important.

4. CONCLUSION

Ce projet a consisté à adapter le code déjà existant d'un écoulement 2D incompressible périodisé autour d'un obstacle simple en LBM et à modifier le type de conditions aux bords. J'ai implémenté la méthode de frontière immergée, qui permet d'accéder aux valeurs et à la répartition des forces qu'exerce l'écoulement sur l'obstacle. Cette méthode permet de plus, de considérer des obstacles solides de formes quelconques et d'implémenter l'action d'un mouvement imposé depuis l'extérieur (ou subi par l'effet de l'écoulement). Dans l'étude que j'ai menée, je me suis intéressé à l'influence du nombre de Reynolds et à l'effet de la mise en rotation de l'obstacle sur les coefficients de traînée et de portance.

Même si les résultats obtenus sont quantitativement inexactes comparés à des études numériques similaires, ils reproduisent et expliquent certains phénomènes physiques observés, comme l'effet Magnus. Une étude plus approfondie de l'influence des différents paramètres physiques (rapport d'aspect, rayon du cercle, etc.) et numériques (résolution spatiale, choix de la fonction noyau, nombre de points, etc.) permettrait de tracer un espace des phases plus complet du comportement de l'écoulement autour d'un obstacle. Il aurait aussi pu être intéressant d'étudier la répartition des forces sur l'obstacle, l'orientation de la résultante des forces, la fréquence d'émission des tourbillons, ou encore l'établissement et la localisation des fluctuations de vitesses et de force. De même, il est possible de s'intéresser à l'influence de la forme de l'obstacle ou à l'effet de la translation ou la déformation de l'obstacle (extérieure ou sous l'effet du fluide), possibilités offertes par la méthode de frontières immergées.

REMERCIEMENTS

Je remercie Emmanuel Lévêque pour ses précieux retours ainsi que Guillaume Laibe et Antoine Venaille. Je remercie Élie pour ses quelques conseils spécifiques. Mais surtout un immense merci à toute l'équipe du JPCE <3!!!

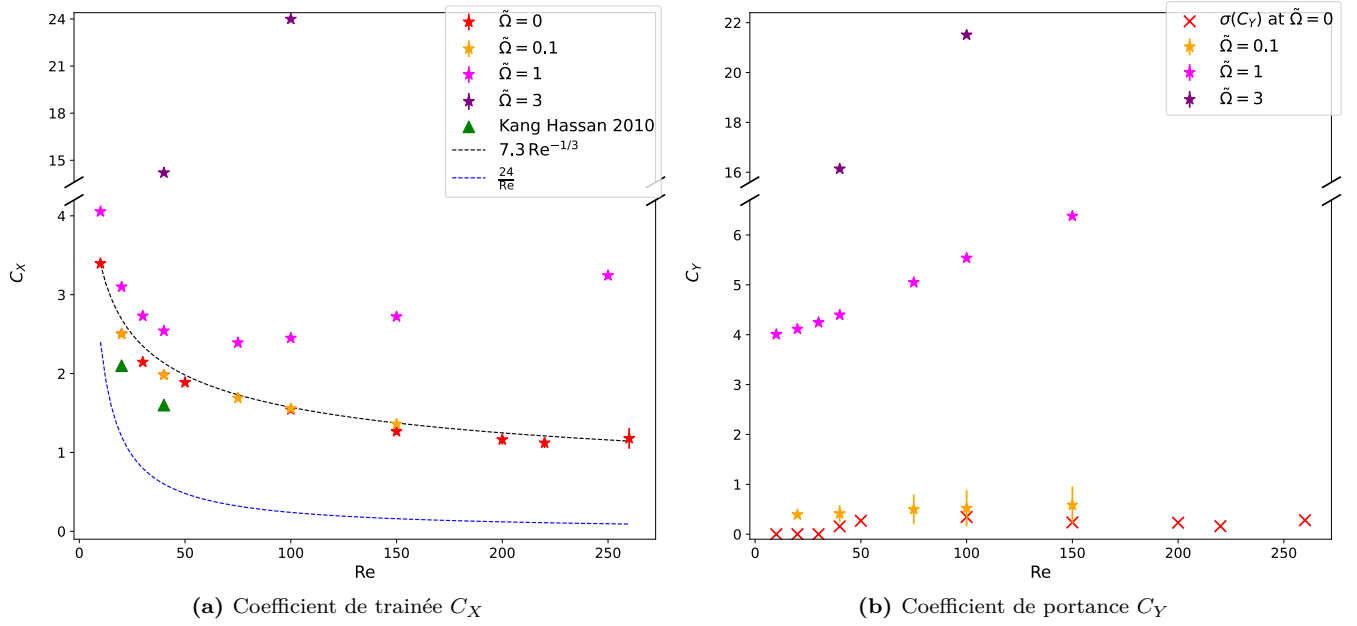
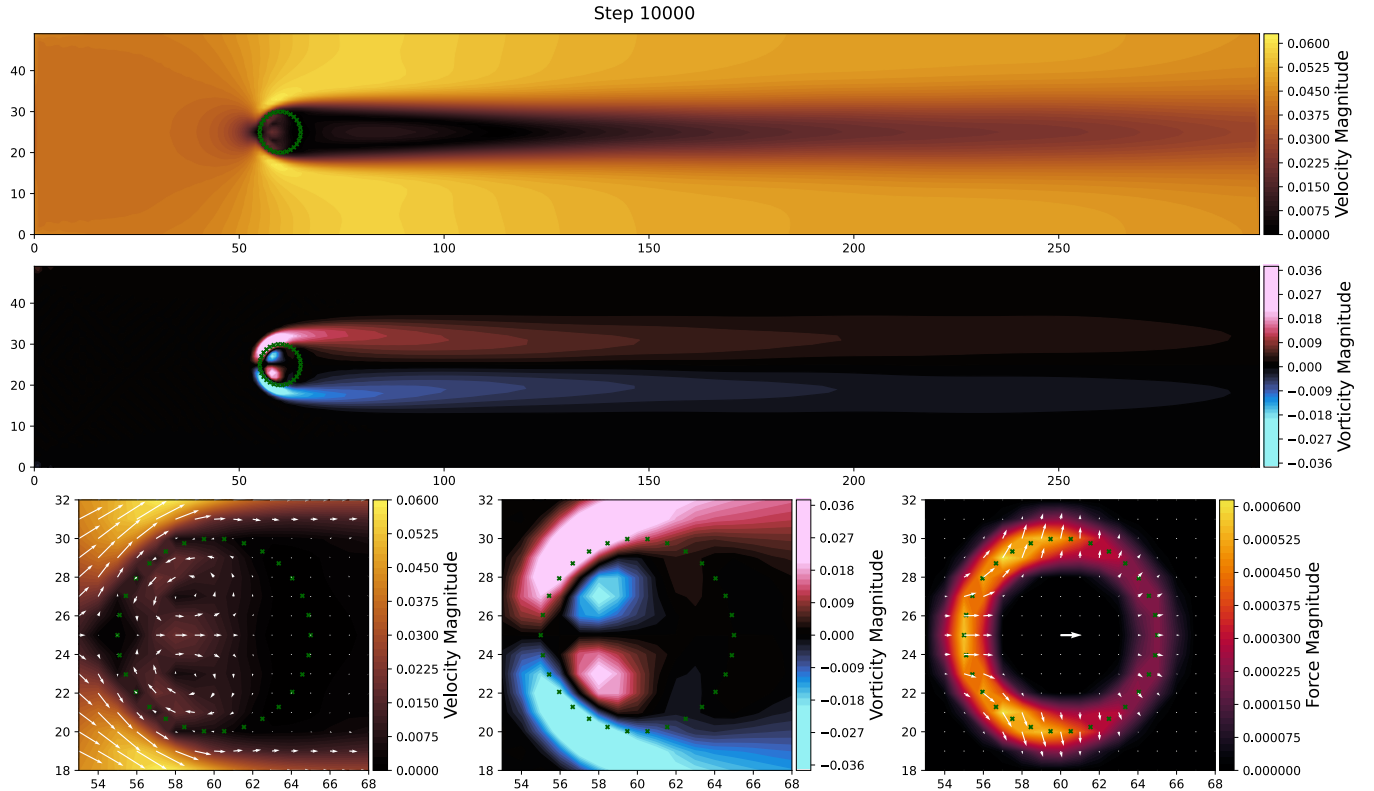


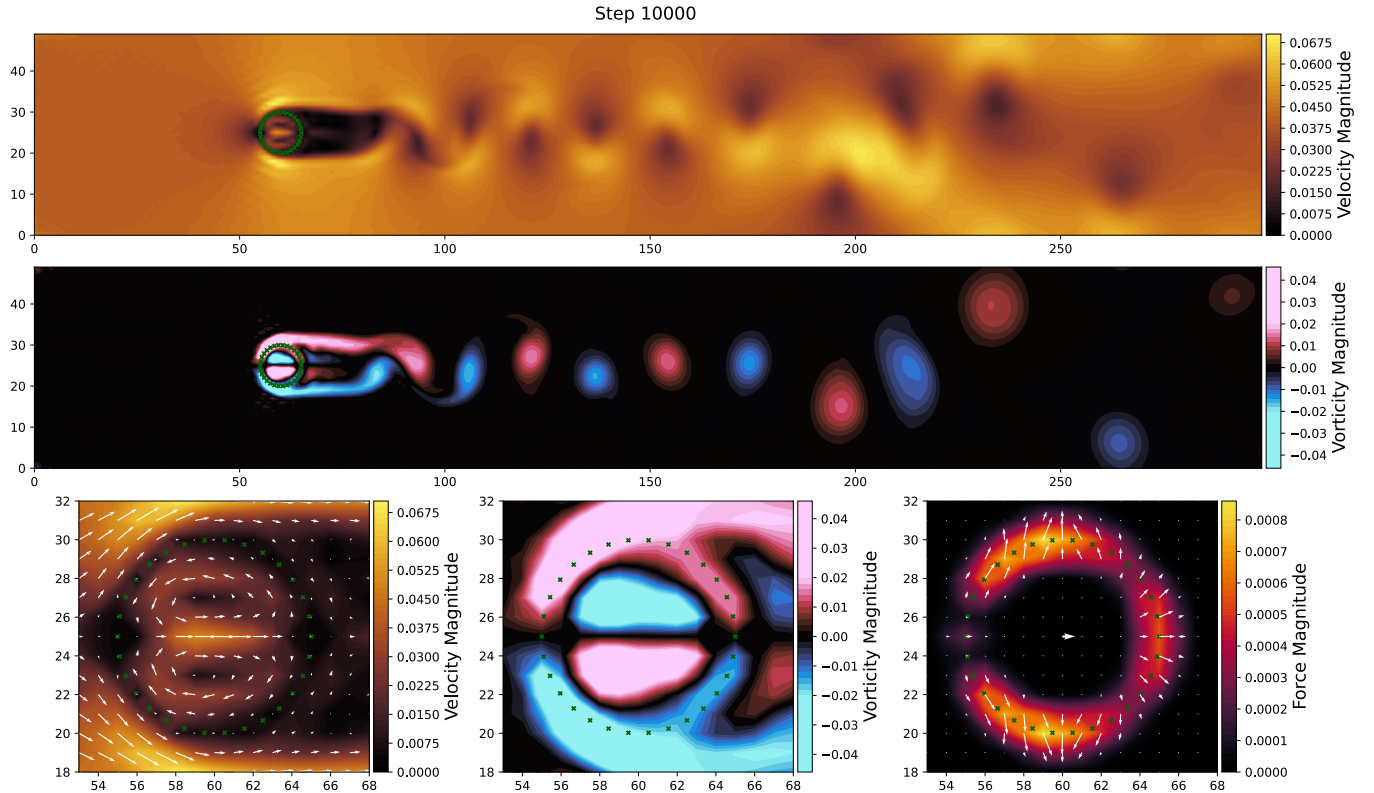
Fig. 8 Comportement des coefficients de trainée et portance en fonction du nombre de Reynolds, Re , et du taux de rotation, $\tilde{\Omega}$. Ici, l'obstacle est circulaire de rayon $R = 5$, avec $N_b = 35$, $(N_x, N_y) = (300, 50)$. La fonction d'interpolation utilisée est ϕ_{sqrt2} . Ces valeurs sont moyennées sur une dizaine de millier d'itérations après stabilisation de l'écoulement et les barres d'erreurs proviennent donc des fluctuations dues à la turbulence, à la déstabilisation de l'écoulement ou au décollement alternatif des vortex.

RÉFÉRENCES

- (1) CHAPMAN, S. , *The mathematical theory of non-uniform gases, An account of the Kinetic Theory of Viscosity, Thermal Conduction and Diffusion in Gases*, 1952.
- (2) CERCIGNANI, C. in *The Boltzmann Equation and Its Applications*; Springer New York : New York, NY, 1988, p. 40-103.
- (3) HE, X.; LUO, L.-S. *Phys. Rev. E* **1997**, *56* , 6811-6817.
- (4) KRÜGER, T.; KUSUMAATMAJA, H.; KUZMIN, A.; SHARDT, O.; SILVA, G.; VIGGEN, E. M. , *The Lattice Boltzmann Method: Principles and Practice*; Graduate Texts in Physics; Springer International Publishing : Cham, 2017.
- (5) FENG, Z.-G.; MICHAELIDES, E. E. *Journal of Computational Physics* **2004**, *195* (2) , 602-628.
- (6) KANG, S. K.; HASSAN, Y. A. *International Journal for Numerical Methods in Fluids* **2011**, *66* (9) , 1132-1158.
- (7) ESHGHINEJADFARD, A.; ABDELSAMIE, A.; JANIGA, G.; THÉVENIN, D. *Particuology* **2016**, *25* , 93-103.
- (8) KRÜGER, T.; KUSUMAATMAJA, H.; KUZMIN, A.; SHARDT, O.; SILVA, G.; VIGGEN, E. M. in *The Lattice Boltzmann Method: Principles and Practice*; Graduate Texts in Physics; Springer International Publishing : Cham, 2017, p. 153-230.
- (9) KÖHLER, F. LBM Fluid Simulation in JAX, 2023, <https://github.com/Ceyron/Lattice-Boltzmann-Method-JAX>.
- (10) ZOU, Q.; HE, X. *Physics of Fluids* **1997**, *9* (6) , 1591-1598.
- (11) DE ROSIS, A.; LÉVÊQUE, E. *Computers & Mathematics with Applications* **2016**, *72* (6) , 1616-1628.
- (12) WANG, Z.; WEI, Y.; QIAN, Y. *Applied Mathematical Modelling* **2020**, *81* , 428-440.
- (13) KRÜGER, T.; KUSUMAATMAJA, H.; KUZMIN, A.; SHARDT, O.; SILVA, G.; VIGGEN, E. M. in *The Lattice Boltzmann Method: Principles and Practice*; Graduate Texts in Physics; Springer International Publishing : Cham, 2017, p. 231-263.
- (14) TRITTON, D. J. in *Physical Fluid Dynamics*; Springer Netherlands : Dordrecht, 1977, p. 18-29.



(a) Champs de vitesses eulériennes, de vorticité et de forces (avec zooms autour de l'obstacle) pour $Re = 50$ (régime de bas Reynolds). L'écoulement reste laminaire et se déstabilise peu initialement.



(b) Champs de vitesses eulériennes, de vorticité et de forces (avec zooms autour de l'obstacle) pour $Re = 250$ (régime de plus fort Reynolds). Dans ce régime, une allée de tourbillons commence à se former derrière l'objet.

Fig. 9 Écoulements autour d'un obstacle circulaire à l'itération de temps $t = 10000$ dans deux régimes. L'objet a pour rayon $R = 5$ et possède $N_b = 35$ pointeurs. La fonction d'interpolation utilisée est $\phi_{\text{sqr}2}$. La vorticité selon z , $\omega \doteq \partial_x u_y - \partial_y u_x$, permet de mieux visualiser l'apparition et la localisation des tourbillons dans le sillage de l'objet.

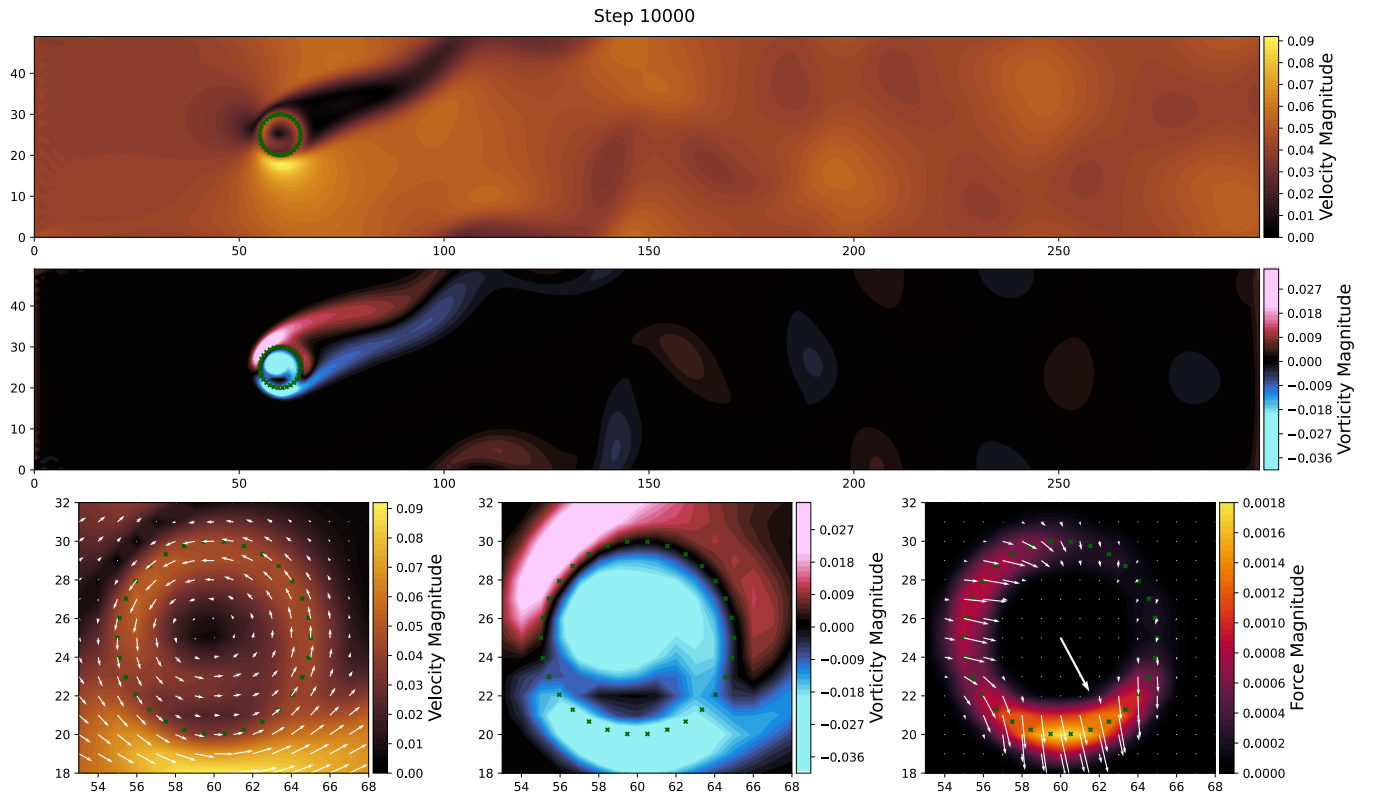


Fig. 10 Champs de vitesses eulériennes, de vorticité et de forces (avec zooms autour de l'obstacle) pour $Re = 50$ avec mise en rotation de l'obstacle à $\tilde{\Omega} = 1$ à l'itération de temps $t = 10000$. L'objet a pour rayon $R = 5$ avec $N_b = 35$ et la fonction d'interpolation utilisée est $\phi_{\text{sqr}t2}$. On note l'apparition d'une composante non négligeable de la force s'appliquant sur l'objet selon la direction $-\mathbf{e}_y$ qui provient de l'effet Magnus.