Project Report

Lattice Boltzmann flow simulation

Daniel Dugas

1. Introduction

The first goal of this project is a working Lattice Boltzmann simulation of a cylinder profile in a steady flow stream, modelled with the Navier-Stokes equations. In order to understand and demonstrate the capabilities of the Lattice Boltzmann method, the chosen case is a well-studied one. Additionally, an optional extension of the task consists in simulating a moving cylinder in the same flow conditions.
Once the code is functional, results of the simulation can thus be compared to the literature, and the simulation validated.

The purpose of this undertaking is to gain insight into the theory behind Lattice Boltzmann fluid simulations, and the practical aspects of their execution. As such, limitations of particular methods are discovered and other models which potentially allow such issues to be circumvented can be considered.
The process of implementation, as a complement to the essential theoretical approach, is an engaging way of further understanding the underlying concepts of the method. 'Tinkering', - that is considering minute modifications on the system and identifying their effects, - when combined with theoretical reasoning can provide useful clarity regarding the processes which occur in the implementation and their significance.

The report is organized according to the following structure. The next section provides a description of the selected scheme, and it's governing equations. Section 3 explains the methodology for coding the scheme, along with details about the decisions which were made during the evolution of the simulation. Results of the simulation are provided in section 4, and analysed in comparison with what was expected. The final section, is a conclusion, and discussion about the various aspects of the project.

2. Theory - Scheme

2.1. The LB method

The Lattice Boltzmann method (also written as LBM) is a relatively recent simulation technique. Initiated with an evolution of the Lattice Gas Automata[1], LB methods differ from classical fluid simulation methods in that they are considered *bottom-to-top*, rather than *top-to-bottom* approaches.
Like Cellular Automata which migrate and collide populations based on defined sets of rules[2], Lattice Boltzmann models are based on microscopic particle behaviors. These populations are then, through statistical analysis, combined to yield the macroscopic fluid characteristics (hence the *bottom-to-top* - micro to macro - designation).

Nowadays, with its increasingly numerous derivatives, the LBM method's domain of possible applications has grown, the method itself now being a basis for several specialized models.
The lecture introduced and explained many of those actual LB models, from the kinetic model for advection-diffusion equations in one dimension, to mixture simulation, higher order LBMs, and more.
Due to the nature of the simulated flow conditions, the "standard" LB model - the Lattice BGK model - was considered sufficient for this case and is the one implemented and described here. In this section, a summarised version of the model description is provided in order to give a theoretical basis for the simulation and results.

2.2. Moment System and Target Equations

The kinetic system is introduced with,

Three moments:

$$\rho(x,t) = \sum_i f_i$$

$$j(x,t) = \sum_i c_i f_i$$

$$P(x,t) = \sum_i c_i^2 f_i$$

Two consistency equations :
(i.e. two local conservation laws)

$$\sum_i f_i = \sum_i f_i^{eq}$$

$$\sum_i c_i f_i = \sum_i c_i f_i^{eq}$$

[1] Itself a Cellular Automata, the Lattice Gas Automata propagates population in the same fashion as LBM, yet uses boolean values instead of the density distribution function. Though noisy and relatively rudimental, it can still be used to simulate flow past cylindrical sections, for example - http://www.ift.uni.wroc.pl/~sebastian.szkoda/msc.html
[2] A notable example is Conway's Game of Life - http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life

The target equations are conservation of density (ρ) and momentum (ρ**u**).

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (1)$$

$$\partial_t (\rho \mathbf{u}) + \nabla \cdot \mathbf{P} = 0 \quad (2)$$

and the constitutive equation - using tensor notation for convenience,

$$P_{\alpha\beta} = \rho c_s^2 \delta_{\alpha\beta} + \rho u_\alpha u_\beta - 2\nu\rho \left(\partial_\beta u_\alpha + \partial_\alpha u_\beta \right) \quad (3)$$

which are a simple form of the Navier-Stokes equations at a constant temperature.
v is the *kinematic viscosity coefficient*, c_s the *speed of sound*. For the sake of brevity, momentum is also symbolized with the letter *j* in this report.

In addition, the pressure tensor **P** is separated into two parts, its equilibrium and non-equilibrium components, which are useful for derivation of the kinetic system, as well as for the calculation of Grad's populations, later in this project.

$$P^{eq}_{\alpha\beta} = \rho c_s^2 \delta_{\alpha\beta} + \rho u_\alpha u_\beta \quad (4)$$

$$P^{neq}_{\alpha\beta} = -2\nu\rho \left(\partial_\beta u_\alpha + \partial_\alpha u_\beta \right) \quad (5)$$

Derivation, and discretization of the kinetic system yields the discrete kinetic equation,

$$f_i(\mathbf{x} + c_i \delta_t, t + \delta_t) - f_i(\mathbf{x}, t) = Q_i \quad (6)$$

Where in Q_i is the collision operator which represents a relaxation step - in this case the BGK collision integral,

$$Q_i = -\frac{1}{\tau}(f_i - f_i^{eq}) \quad (7)$$

and the two-dimensional equilibrium distribution function,

$$f_i^{eq} = \rho W_i \prod_{j=1}^{D=2} (2 - \sqrt{1 + 3u_j^2}) \left(\frac{2u_j + \sqrt{1+3u_j^2}}{1-u_j} \right)^{c_{ij}} \quad (8)$$

The equations recovered with this Bhatnagar-Gross Krook (BGK) collision model are in this case - LBGK - limited to low mach numbers, and incompressible. A detailed explanation of this derivation - in the one dimensional case - is provided for example in the lecture 3 script of the course.

3. Methodology

The evolution of this project was established according to the following structure:

Milestone I - Working simulation inside the domain without obstacle or walls
Milestone II - Obstacle and boundary conditions
Milestone III - Walls
Milestone IV - Inlet and Outlet
Milestone V - Force Measurement
Milestone VI - Grad at Outlet
Milestone VII - Moving Obstacle
Milestone IIX - Grad BB

For clarity, this section of the report follows the same structure, each milestone's salient points being discussed in its respective subsection.

3.1. Milestone I

MATLAB or C++ ?
Although C++ offers invaluable performance possibilities, especially when using Euler/Brutus, the modularity of MATLAB, combined with greater experience on the latter meant faster and more intuitive development.
In the end habit and ease-of-use led to the bulk of the code being programmed in MATLAB, which offers some very useful matrix operations.

This milestone was otherwise mostly trivial coding work, it involved some initial decisions as to the future direction of the code, such as what form the population matrices would have, based on the handling they would go through.

3.2. Milestone II

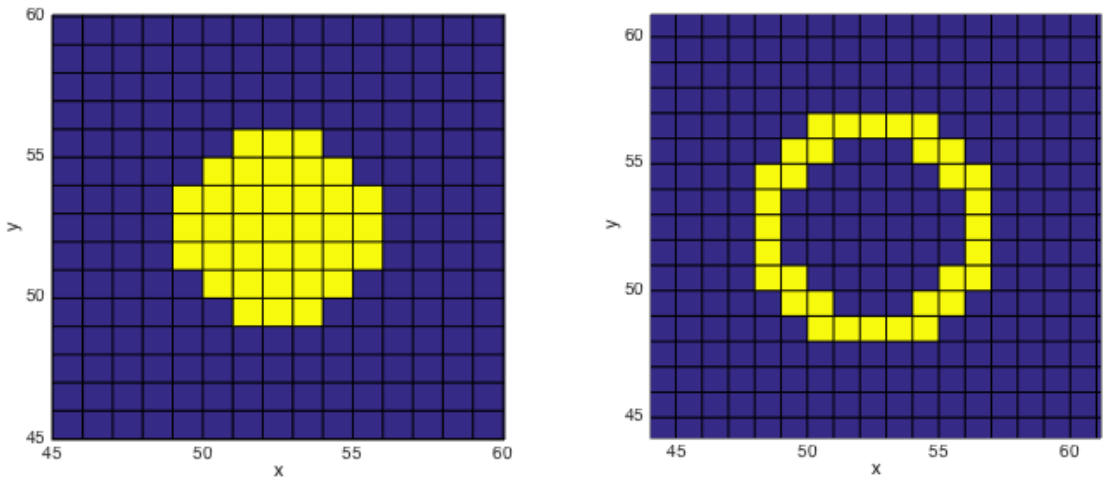


***Figure 1**: Left - solid cylinder nodes. Right - fluid boundary nodes. In the case of a moving object these are recalculated with each time step.*

The addition of an obstacle to the flow simulation consists mainly of implementing special treatment of the populations at the boundary between fluid and solid.
This is due to the fact that fluid nodes in the solid-fluid boundary (left image in figure 1, in this example) require populations advected from neighbouring solid nodes, which are unknown. The set of populations for lattice velocities leading from solid to fluid, are referred to as *missing populations.* A successful boundary condition is one which fills these missing populations in coherence with macroscopic fluid behavior, while preserving stability.
Several methods exist which accomplish this. in this initial implementation a full-bounce-back scheme was pursued.
In this model, populations exiting the fluid boundary nodes in direction of the solid are considered as reflected, and thus applied to missing populations which correspond the opposite of their initial lattice velocity.

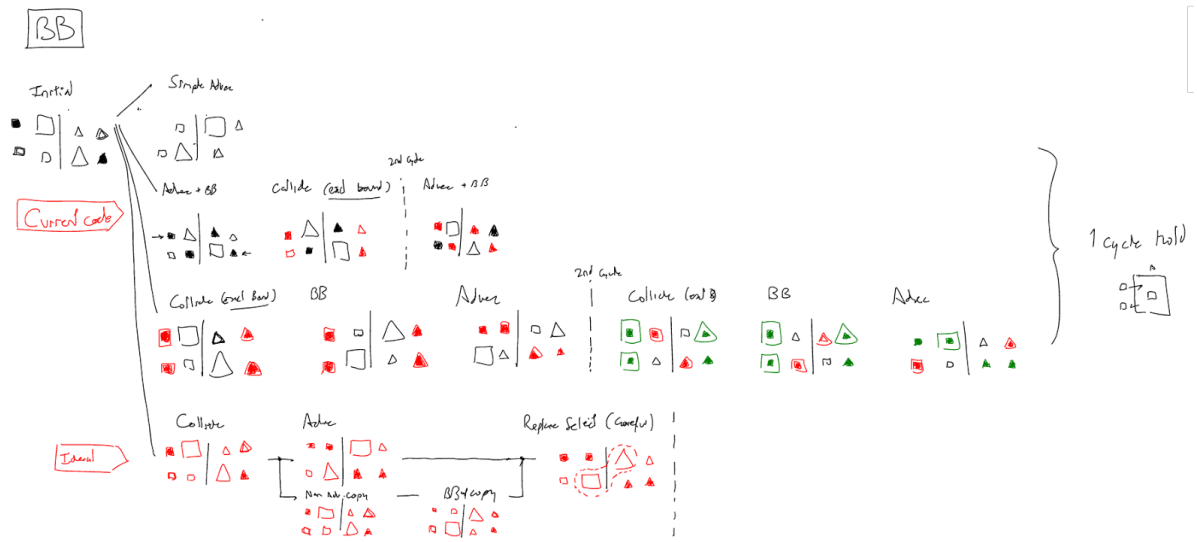


Figure 2: *Early notes for the initial implementation of full bounce-back - with a lag of 1 timestep.*

Detail: In figure 2 above, only 1 lattice dimension is considered. Each *diagram* is a 4x2 group of 8 *pictograms* - triangles and squares - with a vertical line separation in the middle, and a title above. Diagrams represents the state of populations after the modifications referenced in their title are applied to the previous state (f.ex., diagrams with the title “Advec” show the state of populations after advection has been applied to the previous diagram).
Each pictogram represents a population; the leftmost column in each diagram contains fluid nodes, the second leftmost is a solid ‘ghost’ node, and the two rightmost columns are solid nodes. The top row contains populations for positive (going right) lattice velocities, the bottom row populations for negative lattice velocities. Triangle pictograms represent populations originally in the solid, and squares populations initially in the fluid or ghost nodes. Ensuring that pictograms did not cross the separation line (black line in the middle of each diagram, separating ghost from solid cells) would ensure that no leakage from solid to fluid occurs and vice-versa.
In this one-dimensional full-BB example, a BB step consists in switching top and bottom pictograms in the columns on both side of the separation - exchanging populations heading left and right.

3.3. Milestone III

Similarly to the solid boundaries, wall nodes have neighbours which do not provide advected populations, and thus display the same missing populations issue if no boundary condition is applied. In this step, periodicity at the top and bottom walls were replaced by a free-slip boundary condition which conserves the horizontal fluid velocity by applying a similar reflection of the populations exiting the domain as in bounce-back, with one difference.
Instead of being reflected in the direction opposite of their inital travel, exiting populations are instead reflected with only their vertical velocity component being mirrored.

Applying this free-slip mirroring condition to the populations can be represented according to the following schematic,

```
---wall---                                  ---wall---
8   1   5                                  7   3   6
  \ | /                                      \ | /
4 - 0 - 2          Becomes,                4 - 0 - 2
  / | \          after Free-Slip:            / | \
7   3   6                                  8   1   5
```

where populations for the lattice velocities of index 7, 3, and 6 are missing - here, contain irrelevant values - before the free-slip condition is applied. The fact that 8, 1, and 5 are replaced with garbage populations does not matter since they will be advected out of the domain in the next time-step.

3.4. Milestone IV

A simple scheme for the outlet inlet is applied here. Since the domain was periodic in the previous milestone, the same analysis as in the BB case has to be conducted to ensure that the outlet and inlet are implemented correctly, and no leakage occur from one to the other.

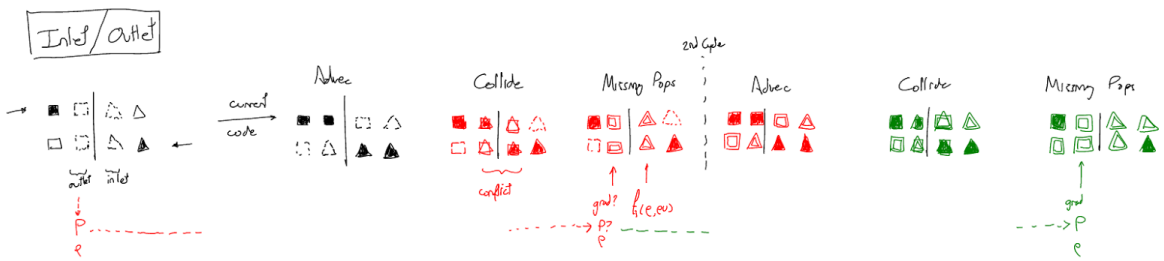


Figure 3: *Early notes for the initial implementation of the inlet and outlet conditions - using the same pictographic representation as in fig. 1*

3.5. Milestone V

In order to calculate the force exerted on the cylinder during simulation, the momentum exchange method was implemented as explicited in [2], whereby,

$$F = \sum_{all\ x_b} \sum_{i \in \overline{D}} \widehat{c}_i [f_i(x_b, t) + \widehat{f}_i(x_{s,i}, t)] \quad (9)$$

in which $\mathbf{x_b}$ is the position of all fluid boundary nodes, $\overline{D}$ the set of missing population indexes, $\widehat{c}_i$ the opposite of lattice velocity of index i (i.e. $\widehat{c}_i = -c_i$), f_i the population at the fluid boundary node for lattice velocity c_i, and $\widehat{f}_i$ the population in the neighboring solid boundary node (where $\mathbf{x_{s,i}} = \mathbf{x_b} + \mathbf{c_i}$)[3] for the lattice velocity $\widehat{c}_i$.

3.6. Milestone VI

Implementing Grad's populations at the outlet was attempted at this stage, leading to difficulties with instability due first-order extrapolation.
Based on the results from the analysis of different boundary conditions in the case of vortex shedding after a cylinder - as available in [1] - improvements over the old populations method in this scenario were deemed negligible for a large enough domain. This fact, in addition to the higher computational cost led to the old populations method being used instead in the following milestones.

3.7. Milestone VII

Adding movement to the obstacle was rather straightforward. At each timestep the cylinder's discrete grid position must be updated based on an arbitrary continuous time-dependent position function, and the solid and fluid nodes and boundaries should be updated accordingly. In addition, the values of the cylinder velocity during this transition have to be taken by differentiating the continuous position function, and not the discrete grid position over two timesteps. This is important, as during any grid displacement of the cylinder, discrete velocity reaches supersonic values.

3.8. Milestone IIX

In order to account for a moving obstacle, the boundary condition at the cylinder had to be modified. The updated implementation is described in [5], using grad's populations to extend the bounce-back method, and take into account the imparted momentum from cylinder to fluid.
In addition, in order to update missing populations of the nodes transitioning from solid to fluid, several method are available. The selected one involved the advection and collision of populations inside the solid as though they were fluid nodes, whereas the macroscopic velocity is set to that of the cylinder.

[3] Assuming a timestep of 1. In this report, timestep, and spatial step - distance between nodes - are omitted in most equations, as their value is chosen to be 1 during this project.

4. Results

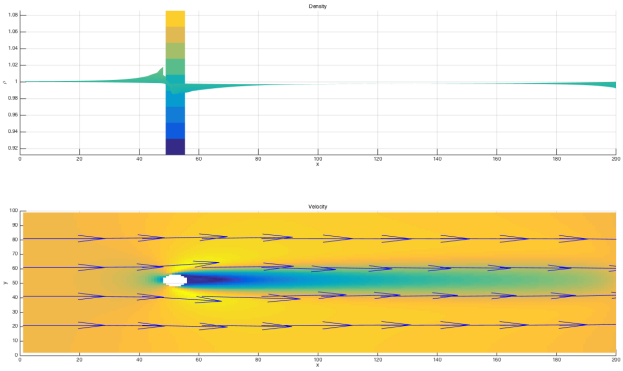
4.1. Vortex Shedding

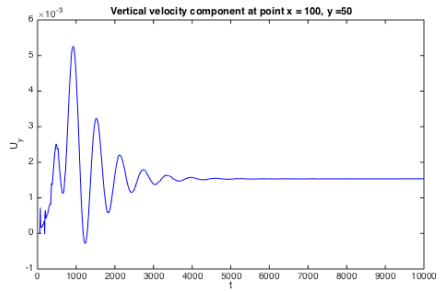
As visible in figure 4, vortex shedding does not appear at low Reynolds number (approximately below Re = 45), in accordance with past experimental data. The unsteady wake does appear in the higher Reynolds number cases, where periodic detachment of the vortices on each side of the cylinder form the visible von Karman vortex street. In these scenarios, after a significant simulation time, a unique frequency of vortice detachment appears.

The Strouhal number gives an adimensionalized measure of the vortex shedding frequency. It can be defined as

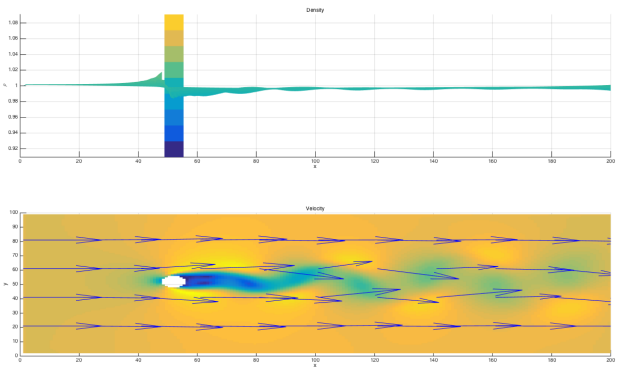
$$St = \frac{f_s D}{u_\infty} \qquad (10)$$

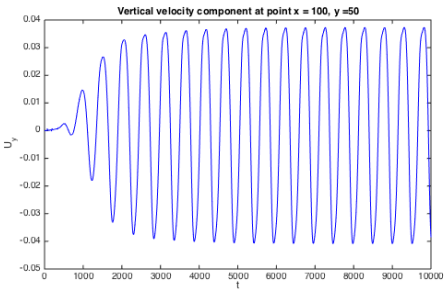
where f_s is the *shedding frequency*, D the *cylinder diameter,* and u_∞ the *free-stream velocity*.





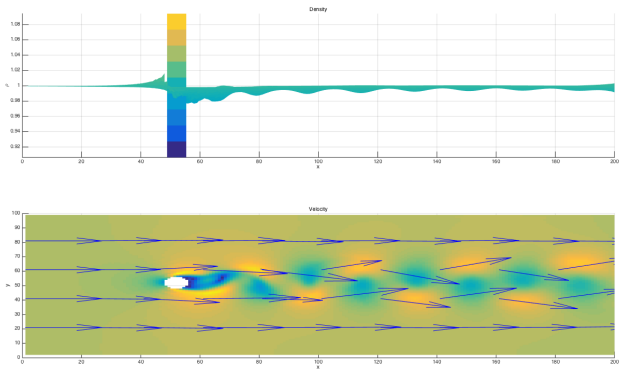
Re = 25

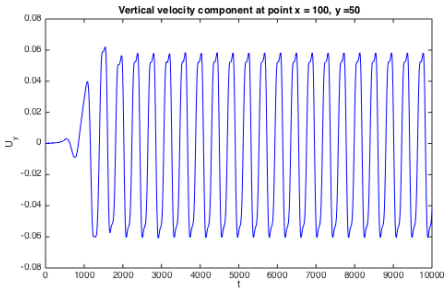




Re = 50

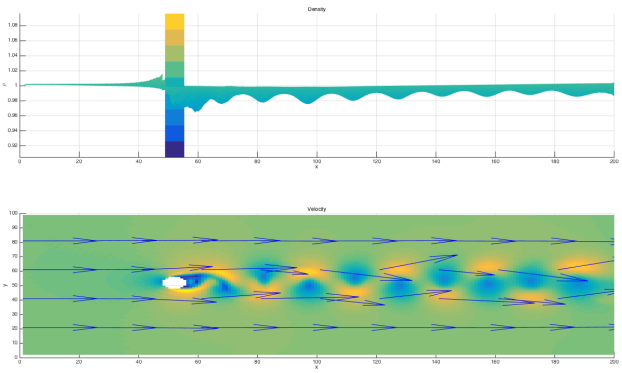
Vortex Shedding is observed displaying a period of 510dt.
Strouhal number: St = 0.137

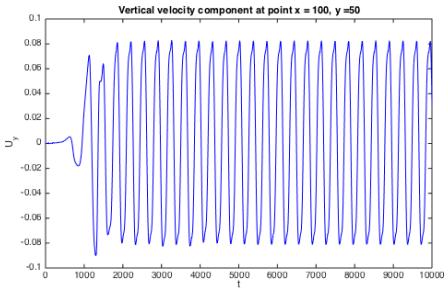




Vortex Shedding of period 410dt is observed.
Strouhal number: St = 0.17

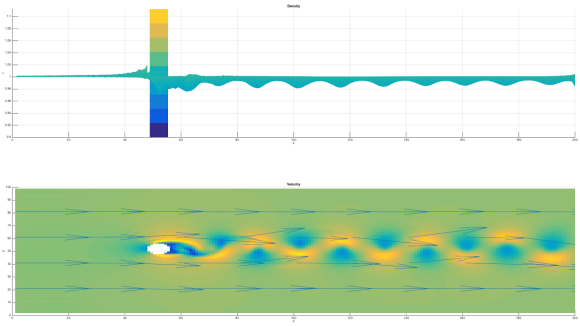
Re = 100

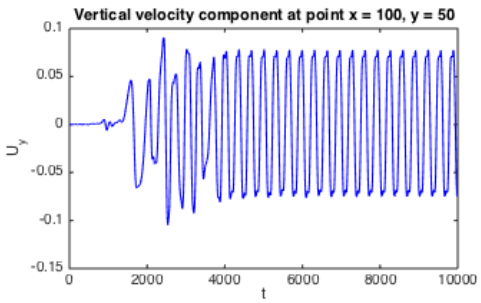




Vortex Shedding of period 350dt is observed.
Strouhal number: St = 0.2

Re = 200





Vortex Shedding of period 320dt is observed.
Strouhal number: St = 0.218

Re = 400

Figure 4: *Vortex Shedding behaviour at various Reynolds numbers.*

The Strouhal number values obtained in these scenarios can thus be compared to empirical values taken from a similar case, as obtained by C. H. K. Williamson in [6]. This relationship is plotted in figure 5.

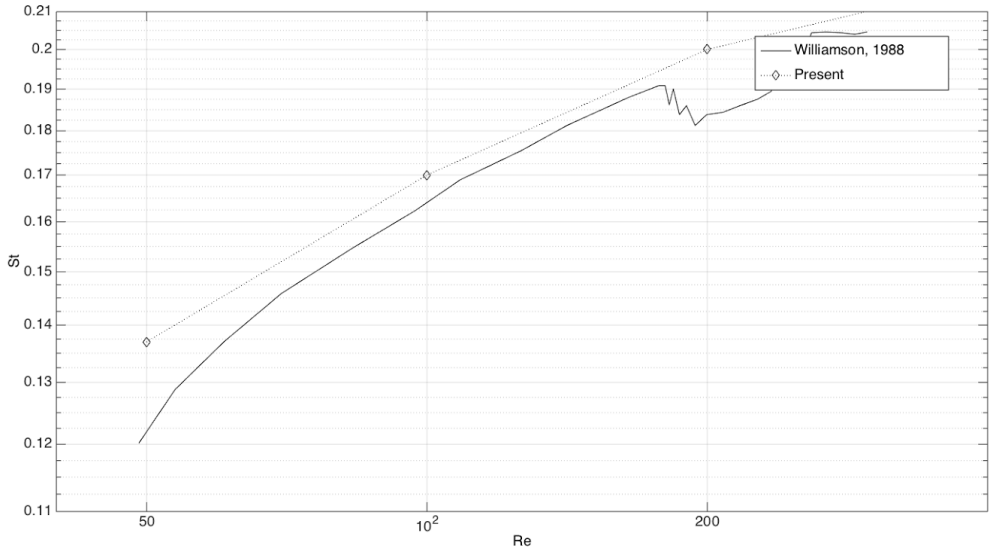


Figure 5: *Dependency of the Strouhal number on the Reynolds number.*

We see that the obtained values are in agreement with the experimental data in the range excluding the lower Reynold limit for vortex shedding and the upper Reynold limit where irregular regimes appear (Re > 180 - see [7]).

4.2. Drag Coefficient

In order to characterize the flow field, the drag coefficient is commonly used, defined by

$$C_D = \frac{F_D}{\frac{1}{2}\rho_\infty u_\infty^2 D} \tag{11}$$

where F_D is the stream-wise component of the force exerted on the cylinder.

In this case, with a cylinder diameter of 10 and a large enough domain to mitigate the effects of boundary conditions at the outlet and walls, the drag coefficient was obtained for a range of Reynolds numbers from 1 to 400. This data is displayed in figure 6a. Furthermore, the agreement of the obtained values with empirical data is visible in figure 6b.

In the chosen implementation of the bounce back condition on the cylinder boundary, interpolation between boundary nodes and possible cylinder exact boundary position was not undertaken. As a result, the simulated cylinder does not have a smooth circular cross section, but rather a rough one - with a contour constituting of right angles and straight edges, such as in the example from figure 1. As visible in figure 6b, this roughness is however not relevant to the drag coefficient in the range of Reynolds numbers allowed by this LB method and studied in this report.

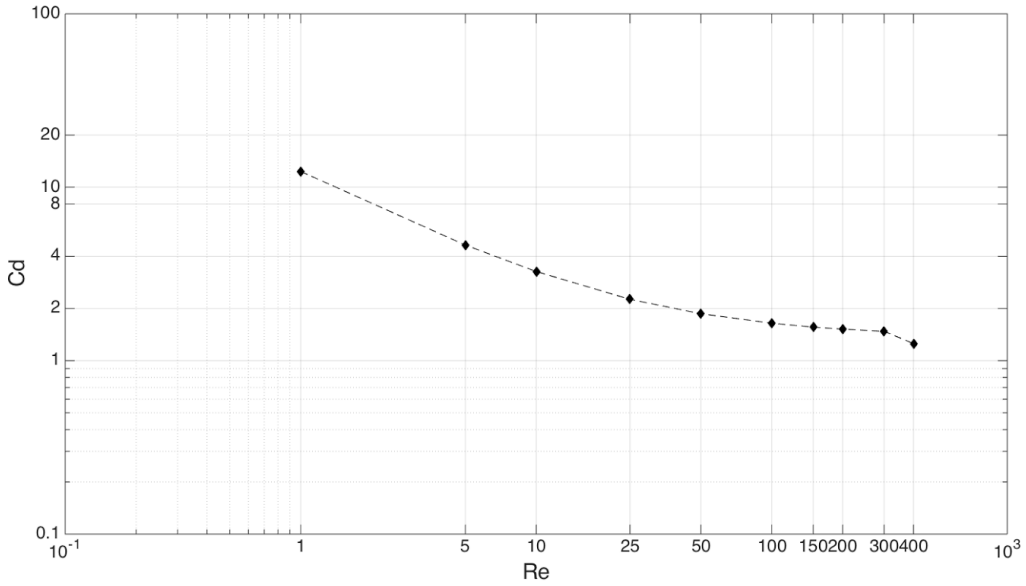


***Figure 6a**: Resulting dependency of the Drag Coefficient on the Reynold's number.*

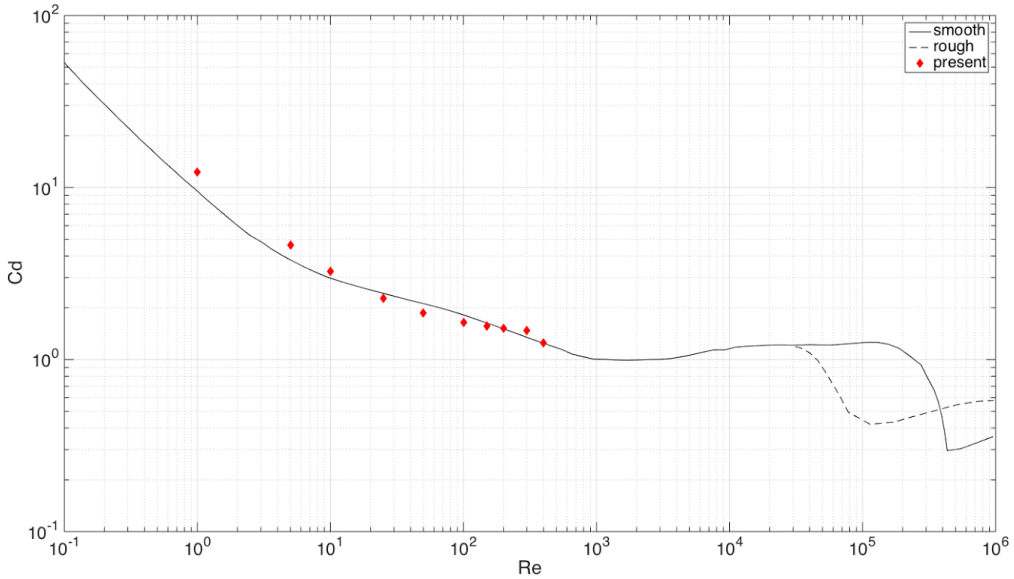


***Figure 6b:** Present simulation compared to empirical values of Cd vs. Re in the litterature*[4]

4.3. Limits of the scheme

As a short demonstration of the limits of this relatively simple form of the lattice boltzmann implementation, the scheme was tested at increasing values of the Reynolds number. This was done in order to reproduce the instabilities theoretically predicted due to the anomalous terms - found after the Chapman-Enskog expansion by perturbation in the invariance equation - which are not annihilated.

[4] ref. *Boundary Layer Theory (7th ed)* by H. Schlichting

As expected, for low viscosity flows the three-velocity model, without correction for over-relaxation, or entropy, shows unstable behavior starting at Reynolds numbers above 400, an example of which is shown in figure 7.

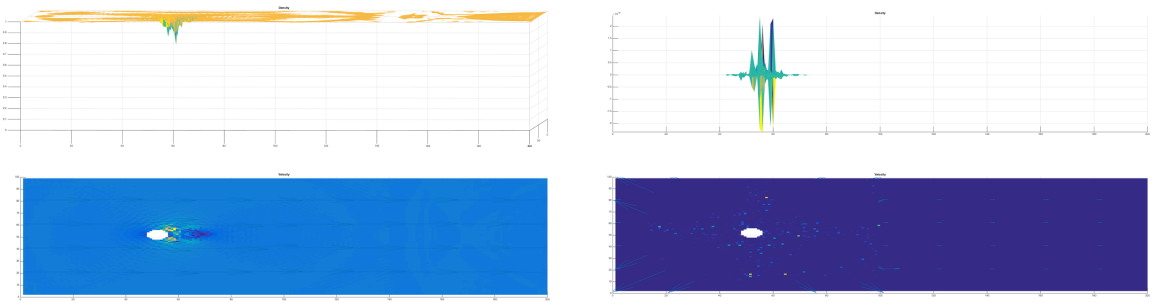


Figure 7: *Instability at high Reynolds numbers - In this case Re = 500*

4.4. Oscillating Cylinder

Simulated behavior of the flow disturbances caused by a vertically oscillating cylinder was also studied here. In this sub-section the movement of the cylinder is purely vertical, according to a sinusoidal term corresponding to the deviation from the cylinder's inital vertical position,

$$\Delta y(t) = A \cdot D \sin(2\pi t/T) \qquad (12)$$

where A is the amplitude of the oscillations relative to the cylinder diameter, D is the cylinder diameter, and T the period of the cylinder vertical oscillation.

In order to characterize the influence of this motion on the cylinder wake, the natural vortex shedding wake frequency - i.e. the shedding frequency of the von Karman vortex street when the cylinder is immobile - is first found through simulation. Throughout this sub-section the Reynolds number is kept constant (Re = 200).
A period of T_s = 570 timesteps, and thus a natural vortex shedding frequency of f_s = 0.001754 Hz was found.
With f_s known for this case, the effect of periodic movement on the simulated wake can be studied by determining the drag coefficient for various oscillation frequencies f.
A frequency coefficient is introduced as

$$F = \frac{f}{f_s} \qquad (13a)$$

or equivalently,

$$F = \frac{T_s}{T} \qquad (13b)$$

For example, the time evolution of C_D for F = 0.7 is shown in figure 8.

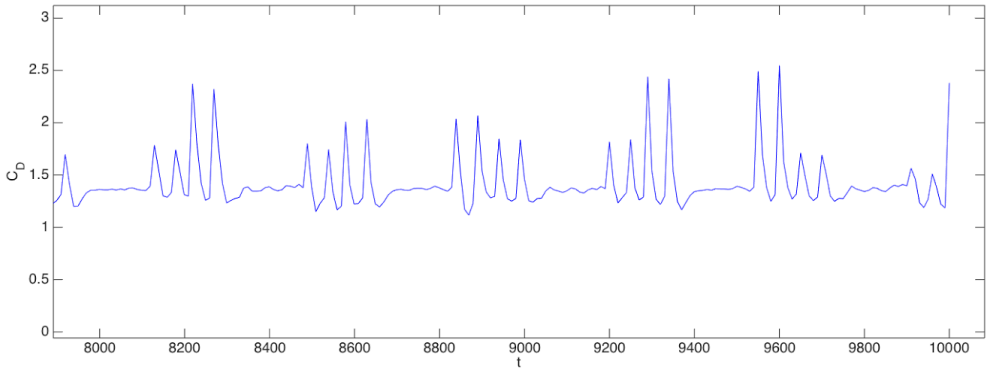


***Figure 8**: Periodicity of C_D in the case of a vertically oscillating cylinder. $f = 0.7*f_{shedding}$ $C_{D,mean} = 1.39$*

The resulting values of $C_D(t)$ are time-averaged, and a mean value, $C_{D,mean,}$ is found. These values as calculated for several frequency ratios are displayed in figure 9. In addition, the drag coefficient values for identical conditions are plotted as found by S. Bao et al., and A. Placzek et al. in [3] and [4], respectively.
Comparison shows that the computed values are in the correct range, however the agreement is not exact, especially in the so-called lock-out regimes.

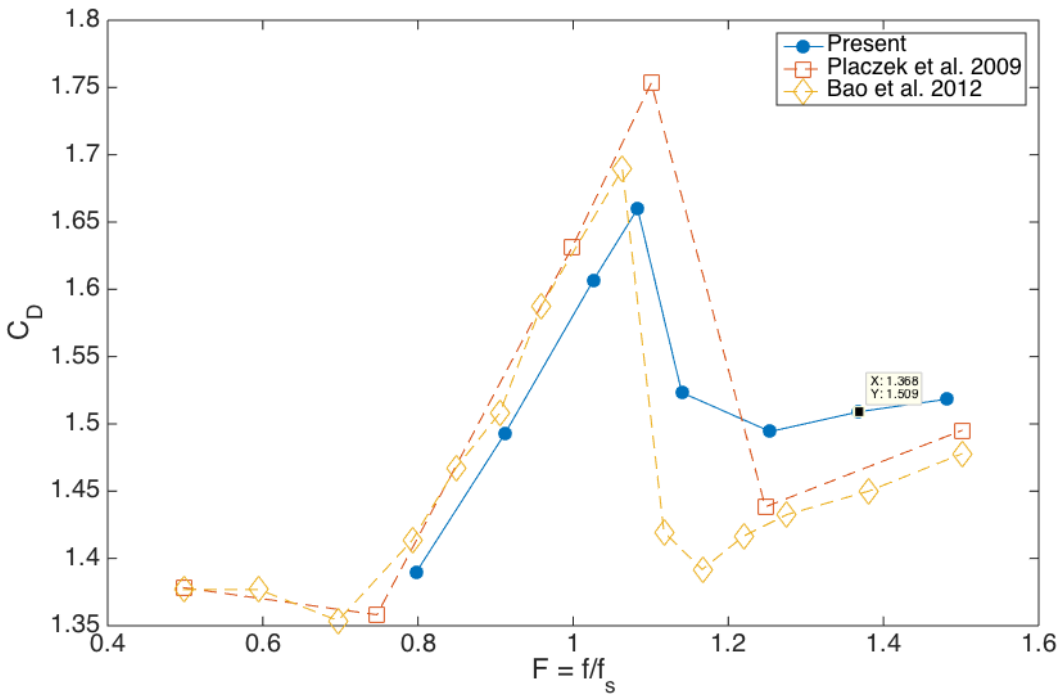


***Figure 9**: resulting mean drag coefficient for different excitation frequencies, in comparison to the litterature.*

5. Discussion

Many intricacies in the implementation of a Lattice Boltzmann model became apparent while making this flow simulator.

One of the strong impressions which emerged in the execution of this project is the intuitiveness of the Lattice Boltzmann method when compared to classical CFD models. Thinking in terms of populations, and subsequently considering the effects that the evolution of those populations will have on the macroscopic fluid behaviour, felt rather natural. In addition, the unintuitive behaviours of interpolation, extrapolation and finite differences being for the most part absent, along with, especially, the localised nature of the method, are a relief to anyone who has had to deal with the spurious and unexpected consequences of errors in classical CFD algorithms.

Another interesting impression concerns how organically a project evolves when implementing the Lattice Boltzmann method. With the basic principle of population advection and collision remaining mostly unchanged - not only in the scope of this project, but also when switching to models which take into consideration entropy, temperature, and so on, - going from basic flow simulation to more advanced flows around complex structures is an elegant process. This is particularly evident when switching from a 1 dimensional model to a 2 dimensional one, or specifying more stringent boundary conditions for example.

Those impressions being of course mostly subjective, the usefulness of such an alternative to classical CFD is however undeniable. Expansive development of theoretical and experimental study of the Lattice Boltzmann in the recent years as evidence of that fact, and is ongoing still.

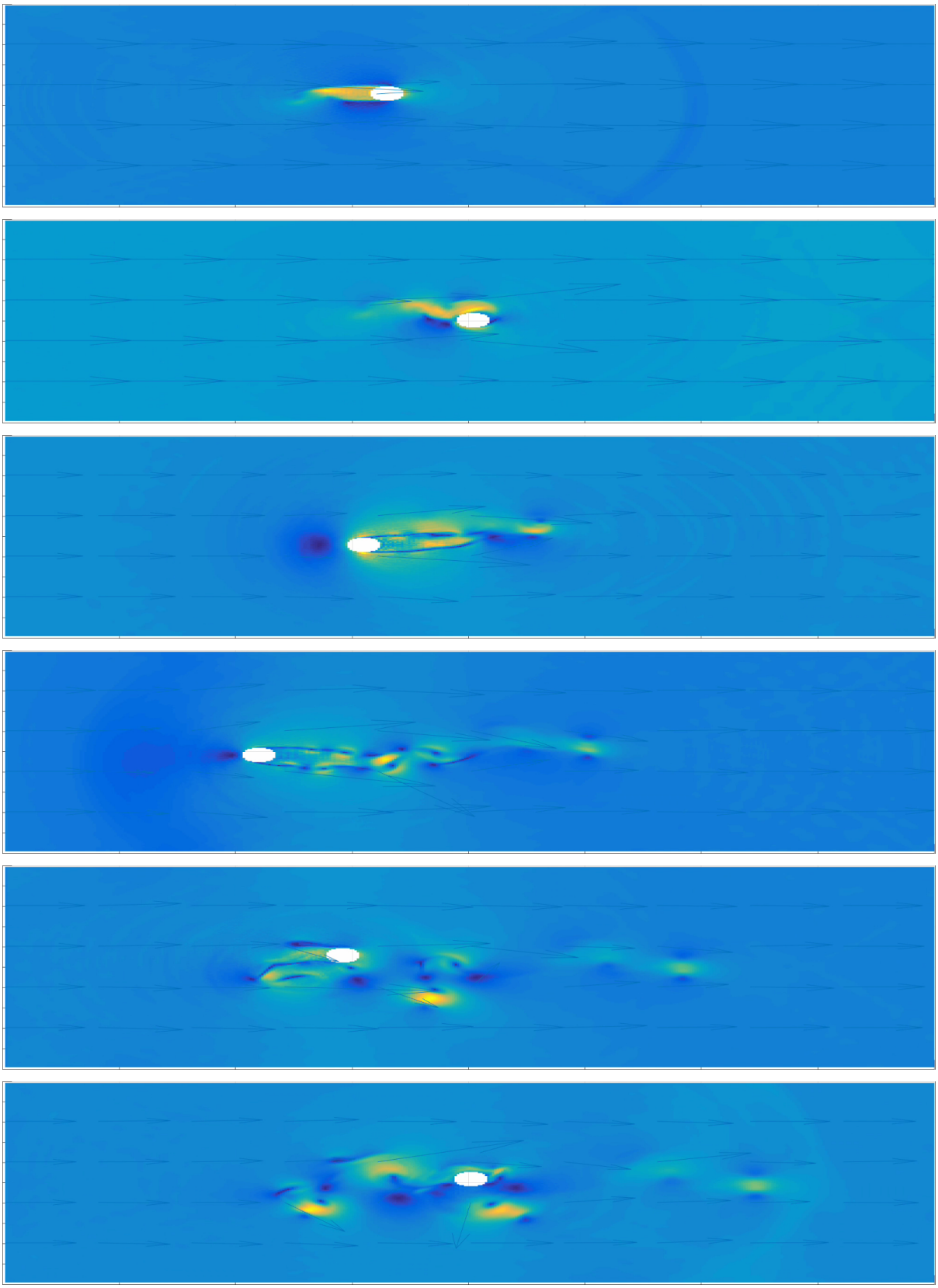


Figure 10: *Simulation of a cylinder undergoing elliptical oscillations in a steady flow.*

References

[1] Zhaoxia Yang, *Lattice Boltzmann outflow treatments: Convective conditions and others,* Computers & Mathematics with Applications (2013)
[2] R. Mei, D. Yu, W. Shyy, L-S. Luo, *Force evaluation in the lattice Boltzmann method involving curved geometry*, Phys. Rev. E (2000)
[3] S. Bao, S. Chen, Z. Liu, J. Li, H. Wang, C. Zheng, *Simulation of the flow around an upstream transversely oscillating cylinder and a stationary cylinder in tandem,* Physics and Fluids (2012)
[4] A. Placzek, J-F. Sigrist, A. Hamdouni, *Numerical simulation of an oscillating cylinder in a cross-flow at low Reynolds number: Forced and free oscillations,* Computers & Fluids (2009)
[5] P. Lallemand, L.-S. Luo, *Lattice Boltzmann method for moving boundaries,* Journal of Computational Physics (2002)
[6] C. H. K. Williamson, *Defining a universal and continuous Strouhal-Reynolds number relationship for the laminar vortex shedding of a circular cylinder,* Physics of Fluids (1988)
[7] F. L. Ponta, H. Aref, *Strouhal-Reynolds Number Relationship for Vortex Streets*, Physical Review Letters (2004)