Scilab case study project on Solving Lid Driven Cavity flow using Lattice Boltzmann Method

Pravinraj T

PSG Institute of Technology and Applied Research, Neelambur, Coimbatore

Computational Fluid Dynamics

07/05/2025

Abstract

The Lattice Boltzmann Method (LBM) has emerged as a powerful mesoscopic numerical approach for simulating fluid flows, offering advantages in handling complex boundary conditions and parallelization. This study explores the implementation of LBM for the classical 2dimensional lid-driven cavity problem using Scilab, an open-source numerical computation tool. The lid-driven cavity flow serves as a benchmark to evaluate LBM's capability in modeling fluid dynamics, particularly in simulating vortices and flow stability. The study employs a D2Q9 lattice model, where the Boltzmann equation is discretized using the Bhatnagar-Gross-Krook (BGK) collision model. Key parameters such as Reynolds number, relaxation time, and velocity distribution are analyzed to validate the simulation. Scilab’s built-in matrix operations facilitate efficient implementation, visualization, and analysis. The results demonstrate LBM’s effectiveness in capturing flow characteristics, highlighting its advantages over traditional CFD techniques. This study underscores the potential of Scilab in fluid flow research, offering a cost-effective alternative for LBM-based simulations.

1.Introduction

The Lattice Boltzmann Method (LBM) has gained prominence as a mesoscopic numerical approach for fluid flow simulations, providing a robust alternative to traditional computational fluid dynamics (CFD) methods. Unlike conventional Navier-Stokes solvers, LBM operates on a discrete lattice, evolving particle distribution functions over time. This approach inherently supports complex boundary conditions, parallel processing, and multiphysics coupling, making it ideal for diverse applications, including microfluidics, porous media, aerodynamics, and thermal transport. Implementing LBM in Scilab, an open-source numerical computing platform, offers several advantages. Scilab's efficient matrix operations, built-in visualization tools, and scripting flexibility allow for rapid prototyping and performance optimization of LBM models. Compared to commercial software, Scilab provides a cost-effective alternative without compromising computational accuracy.

This project focuses on simulating the lid-driven cavity flow, a classical benchmark problem in fluid dynamics, using LBM in Scilab. The study highlights LBM's capability to capture key flow phenomena, such as vortex formation and recirculating regions, demonstrating the effectiveness of Scilab in implementing high-performance numerical simulations.

2.Problem Statement

The objective of this study is to reproduce the results of the 2D square Lattice Boltzmann Method (LBM) implementation as presented by Rene Fink [1] from Wismar University of Applied Sciences in MATLAB, using Scilab. The study employs the D2Q9 lattice model and Bhatnagar-Gross-Krook (BGK) collision operator to simulate the well-known lid-driven cavity flow problem[2]. The bounce-back boundary condition is applied at solid walls, ensuring no-slip conditions, while a moving wall boundary at the top enforces constant velocity, replicating the classic benchmark case. Through this study, the effectiveness of Scilab has to be the demonstrated for LBM-based flow simulations while ensuring consistency with established results.

3.Basic Concepts

The lid-driven cavity flow is a classical benchmark problem in CFD, where a fluid-filled square domain has a moving top lid inducing recirculating vortices. It is widely used for validating numerical methods due to well-documented analytical and numerical solutions. The Lattice Boltzmann Method (LBM) is a mesoscopic numerical technique used to simulate fluid flows. Unlike conventional computational fluid dynamics (CFD) methods that solve the Navier-Stokes equations, LBM models fluid dynamics using the Boltzmann transport equation at a discrete lattice scale. This approach provides advantages in handling complex geometries, multiphase flows, and parallel computations.

The D2Q9 LB model is a two- dimensional, nine-velocity lattice structure where each fluid particle moves in predefined directions. The directions include one central (rest particle), four axial (1,3,5,7), and four diagonal (2,4,6,8) components. The distribution function evolution follows collision and streaming steps, simulating macroscopic fluid behavior.

The D2Q9 model employs 9 discrete velocities $e_i$defined as:

$$e_i = \begin{cases} (0,0), & i = 0 \\ (\pm 1,0), (0, \pm 1), & i = 1,2,3,4 \\ (\pm 1, \pm 1), & i = 5,6,7,8 \end{cases} \tag{1}$$

Each direction has an associated weight w_i :

$$w_i = \begin{cases} \frac{4}{9}, & i = 0 \\ \frac{1}{9}, & i = 1,2,3,4 \\ \frac{1}{36}, & i = 5,6,7,8 \end{cases} \tag{2}$$

The major Governing-equation which handles streaming and Collision is given by

$$f_i(\vec{x} + \vec{e_i}\delta_t, t + \delta_t) - f_i^t(\vec{x}, t) = -\frac{f_i^t(\vec{x}, t) - f_i^{eq}(\vec{x}, t)}{\tau} \tag{3}$$

where LHS is responsible for streaming and RHS for collision with *i=0, 1,..18* and τ is the dimensionless single relaxation time in the Bhatnagar-Groos-Krook (BGK) collision operator approximation. The kinematic viscosity relates to the relaxation time as:

$$\vartheta = c^2 \left(\tau - \frac{1}{2}\right) \Delta t \tag{4}$$

where c is the velocity of sound. The macroscopic density and the macroscopic velocity at each lattice node is defined in terms of above particle distribution function as

$$\rho(\vec{x}, t) = \sum_{i=0}^{e_i} f_i^t(\vec{x}, t) \tag{5}$$

$$\vec{u}(\vec{x}, t) = \frac{1}{\rho} \sum_{i=0}^{e_i} f_i \vec{e_i} \tag{6}$$

The equilibrium distribution function is only a function of local density and velocity and is given by

$$f_i^{eq}(\vec{x}, t) = w_i \rho (1 + \frac{3\bar{e}_i \bar{u}}{c^2} + \frac{9(\bar{e}_i \bar{u})^2}{2c^4} - \frac{3(\bar{u})^2}{2c^2}) \tag{7}$$

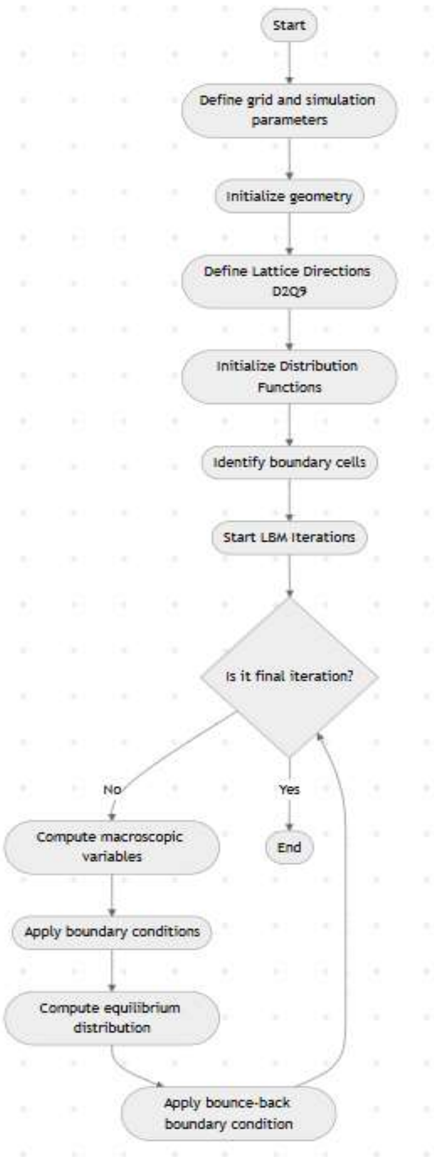
The equilibrium distribution function is computed based on macroscopic fluid density and velocity. Two essential boundary conditions in this project are known as Bounce-Back Boundary Condition which is used for no-slip walls, ensuring zero velocity at solid boundaries.

$$f_{opp(i)}(x, t + \Delta t) = f_i^*(x, t) \tag{8}$$

where f_i^* is the post collision distribution and $f_{opp(i)}$ denotes the direction opposite to i.

The second boundary condition is Moving Wall Boundary Condition $(U_x, U_y)=(U_0,0)$ which is applied at the top boundary to impose a constant velocity, replicating the lid-driven cavity flow problem.

4.Flow chart



5.Software/Hardware used

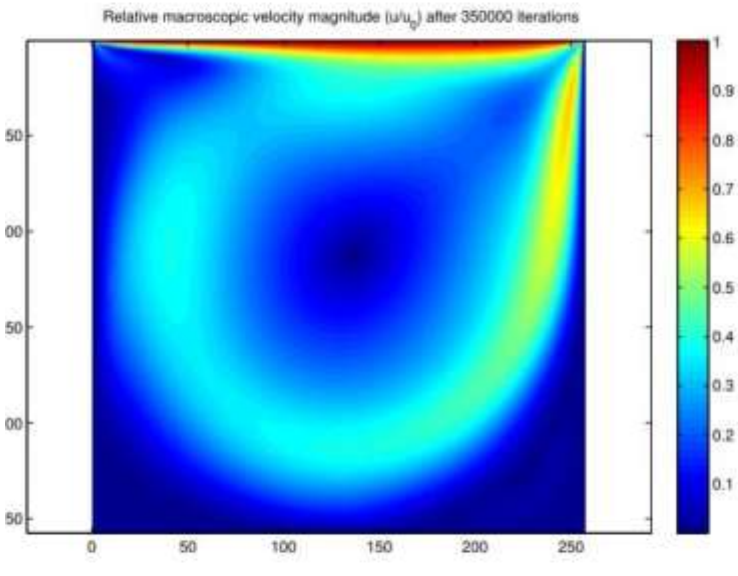
Scilab 2025.0.0, Windows 11 Pro, AMD Ryzen 7 Desktop PC.

6.Procedure of execution

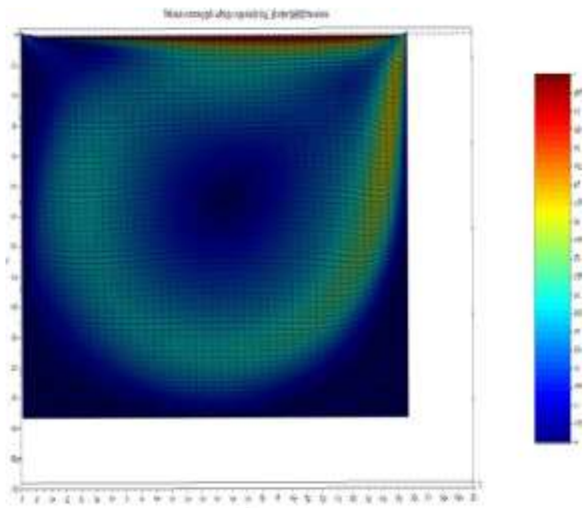
Open Scilab → Open the file " Lid_Driven_Cavity_LBM" → ‘Save and Execute”

7.Result

The case setup was run on Windows 11 Pro with AMD Ryzen 7 5700 machine. The chosen dimension was 257 x 257 with 350000 iterations and simulation lasted approximately for 8 hours to reproduce the results reported in Rene Fink [1]. In case user wants to simulate the faster version, they can reduce the grid size and iterations for example 120 x 120 with 2000 iterations could run quick with reasonable accuracy.

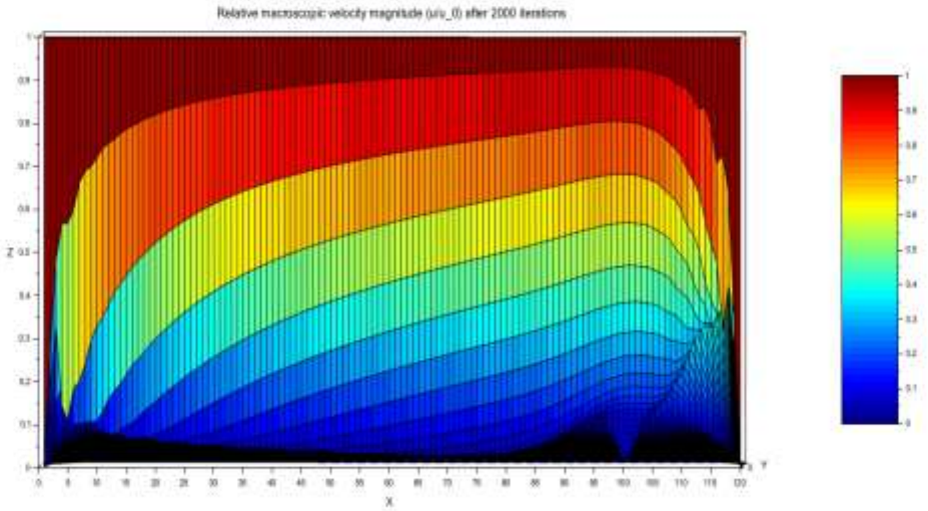


2D LDC Results obtained by Fink[1]



2D LDC Results obtained by this work using Scilab(for 257 x 257 with 350000 iterations

To reproduce the exact results reported by Fink [1], the simulation must be run with `nx = 257` and `iterations = 35,000`. However, this simulation takes approximately 8 hours to complete due to the time-evolution nature of the Lattice Boltzmann Method (LBM). The computation time can be significantly reduced by parallelizing the domain using GPU acceleration or high-performance computing (HPC) systems. For users who wish to view the initial results without reaching steady state, the same simulation can be run with `nx = 120` and `iterations = 2,000`.



2D LDC Results obtained by this work using Scilab (for 120 x 120 with 2000 iterations

In this case study, the Lattice Boltzmann Method (LBM) was successfully implemented in Scilab for simulating lid-driven cavity flow. Enhancing 3D rendering, color mapping, and customizable colorbars improved visualization quality, while better debugging tools, error messages, and a built-in profiler aided code optimization. The simulation produced accurate and reproducible results, demonstrating Scilab's capability as a reliable tool for fluid dynamics modeling. These findings highlight Scilab's effectiveness for LBM simulations while also identifying areas for further improvement to enhance usability and performance.

References

1. Fink, René. *"Implementing the 2D Square Lattice Boltzmann Method in Matlab."* Wismar University of Applied Sciences, Research Group Computational Engineering and Automation, 29 Sept. 2006.

2. Fink, René, et al. "Comparing the Efficiency of Parallelisation Techniques for Simulation Tasks by means of the ARGESIM Benchmarks for Parallel Simulation." *Proc. EUROSIM 2007-6th EUROSIM Congress on Modeling and Simulation*. ARGESIM/ASIM, 2007.

3. Hou, Shuling, et al. "Simulation of cavity flow by the lattice Boltzmann method." *Journal of computational physics* 118.2 (1995): 329-347.