Proceedings of the Eighth International Conference on
Parallel, Distributed, GPU and Cloud Computing for Engineering
Edited by: P. Iványi, J. Kruis and B.H.V. Topping
Civil-Comp Conferences, Volume 12, Paper 1.1
Civil-Comp Press, Edinburgh, United Kingdom, 2025
ISSN: 2753-3239, doi: 10.4203/ccc.12.1.1

Scalable Adaptive Lattice Boltzmann–LES Solver for High Reynolds Number Flows in Porous Media

D. Kashyap[1], M. Grondeau[2] and R. Deiterding[1]

[1] Department Aeronautics & Astronautics, University of Southampton, United Kingdom
[2] Laboratoire Universitaire des Sciences Appliquées de Cherbourg, Université de Caen Normandie, Cherbourg-Octeville, France

Abstract

The present study employs the adaptive lattice Boltzmann solver AMROC-LBM to numerically investigate turbulent flow through a wind tunnel containing porous media with a blockage ratio of 0.5. The porous structure comprises two interlaced cubic arrays, designed to emulate a configuration under concurrent experimental investigation. To resolve turbulence characteristics, a large eddy simulation (LES) framework is integrated into the lattice Boltzmann method (LBM), enabling accurate capture of large-scale flow structures while modelling subgrid-scale effects. Our in-house solver has been rigorously validated against established experimental and numerical benchmarks, with the results demonstrating close agreement, thereby confirming the reliability of the computational methodology. A scalability analysis confirms the solver's computational efficiency on parallel architectures.

Keywords: parallel computing, lattice Boltzmann method, large eddy simulation, dynamic mesh adaptation, porous medium, turbulent flow.

1 Introduction

Porous media are defined by their interconnected void spaces that enable fluid permeation. The investigation of fluid dynamics within porous systems has gained considerable research interest due to their prevalence in natural environments and numerous engineering applications, including separation processes, porous combustion systems, gas drying technologies, hydrocarbon extraction, and pebble-bed nuclear reactor designs [1]. This broad applicability has led to substantial research into turbulent flow mechanics on permeable porous surfaces. In the lattice Boltzmann method (LBM) community, porous media flow is typically approached through two distinct methodologies: pore-scale simulations that directly resolve fluid-structure interactions, and representative elementary volume (REV)-scale modelling that employs averaging techniques for computational efficiency. While REV methods offer practical advantages for large-scale simulations, they inherently lack the ability to capture pore-level flow details. The mesoscopic nature of LBM provides a unique capability to directly simulate pore-scale dynamics while maintaining computational feasibility.

Recent progress in direct numerical simulation (DNS) has enabled more detailed turbulence analysis in porous systems, although existing studies remain limited in scope. Notable work includes Jin and Kuznetsov's [2] investigation of sphere-packed porous beds and Kuwata *et al.*'s [3] study of regular square matrices. However, applications of large-eddy simulation (LES) have primarily focused on dense, low-porosity configurations, leaving a significant gap in the modelling of more complex porous systems. To address this research gap, we present a novel LBM-LES framework incorporating structured adaptive mesh refinement (SAMR) that effectively simulates porous media across the full porosity spectrum, from low to high porosity configurations. This approach builds on the original LBM-LES method developed by Hou *et al.* [4] with several key advancements. Our implementation demonstrates significantly improved computational efficiency compared to DNS-LBM methods, while maintaining high-fidelity solutions [5]. The framework's robustness is ensured by three major components: (1) an MPI-parallelized architecture that enables large-scale simulations, (2) a recursive regularized single-relaxation-time collision operator [6] that provides superior numerical stability at high Reynolds numbers, and (3) dynamic SAMR which typically reduces cell count compared to uniform meshing.

The paper is organized as follows: Section 2 details the adaptive LBM-LES framework implemented in AMROC-LBM. Section 3 provides validation studies and results for the present porous media case. Section 4 analyses parallel scaling performance. Section 5 summarizes the main findings and conclusions.

2 Adaptive lattice Boltzmann solver

The LBM originated from lattice gas automata and uses the discrete Boltzmann equation to model the fluid flow by tracking the evolution of a particle distribution function (PDF), which is the only unknown quantity of the LBM. LBM focuses mainly

on determining the averaged macroscopic variables, such as density and velocity, by evaluating the hydrodynamic moments of the distribution function $f(\mathbf{x}, \xi, t)$.

2.1 Lattice Boltzmann scheme

The LBM method can be seen as a special finite difference scheme for the kinetic equation of the discrete PDF as it utilises discrete space $\mathbf{x}$, time and velocity ξ. The lattice structure in LBM is symbolised by DmQn, where m is the number of dimension and n denotes the number of particle velocities. Here, all computations are three-dimensional and 27 unit direction vectors $\mathbf{e}_\alpha$ are used, which is referred to as D3Q27 model.

The algorithm of LBM starts by setting the initial values of the distribution functions using the discrete equilibrium distribution function derived from non-dimensional prescribed macroscopic variables, and it is common practice to implement LBM internally on the non-dimensional unit lattice. The heart of the algorithm is split into two steps: streaming and collision. The streaming step models the shifting of a distribution function in the direction of motion from one node to the adjacent node, i.e.,

$$\check{f}_\alpha(\mathbf{x}, t) = f_\alpha(\mathbf{x} - \mathbf{e}_\alpha, t). \tag{1}$$

The collision step in our case uses a regularized BGK (Bhatnagar-Gross-Krook) collision operator [6] with forcing term F_α as

$$f_\alpha(\mathbf{x}, t + \Delta t) = \check{f}^{(0)}_\alpha(\mathbf{x}, t) + (1 - \omega)\check{f}^{(1)}_\alpha(\mathbf{x}, t) + F_\alpha(\mathbf{x}, t). \tag{2}$$

The dimensionless relaxation frequency ω relates to the fluid's kinematic viscosity through $\nu = c_s^2 \left(\frac{1}{\omega} - \frac{1}{2}\right) \Delta t$, where $c_s = 340\,\mathrm{m/s}$ represents the speed of sound (constant for all simulations) and Δt denotes the physical time step. In Eq. (2), the post-propagation state is indicated by the $\check{}$ superscript. The equilibrium $f^{(0)}_\alpha$ and non-equilibrium $f^{(1)}_\alpha = f_\alpha - f^{(0)}_\alpha$ components of the distribution function are computed using Malaspinas' regularized recursive (RR) formulation [6]. Our implementation employs the recursive approach up to order 6 for both equilibrium and non-equilibrium parts. It has been established among the research community that this RR-BGK scheme is stable and accurate at high Reynolds numbers.

The conversion from non-dimensional lattice distributions to physical quantities requires proper scaling based on the physical time step Δt and physical grid spacing Δx of the Cartesian mesh, utilizing the lattice speed $c = \Delta x / \Delta t$. In LBM, the macroscopic variables are obtained by taking moments of the distribution function. The gas density ρ and the velocity vector $\mathbf{u}$ are evaluated by a simple rescaling of the 0^{th} and 1^{st} moments of the PDF with a reference density ρ_0 and the lattice speed, respectively. i.e.

$$\rho = \rho_0 \sum_\alpha f_\alpha, \qquad \mathbf{u} = c \sum_\alpha \mathbf{e}_\alpha f_\alpha. \tag{3}$$

The total pressure can be computed from the density as $p = c_s^2(\rho - \rho_0) + p_0$, where $c_s = c\tilde{c}_s$ is the speed of sound and p_0 is the ambient pressure. The lattice speed of sound $\tilde{c}_s$ is equal to $1/\sqrt{3}$ for D3Q27.

2.2 Adaptive mesh refinement

The present lattice Boltzmann implementation utilizes a uniform Cartesian grid discretization. To achieve improved accuracy with enhanced computational efficiency, a block-structured adaptive mesh refinement (SAMR) methodology is employed [7]. This approach organises computational cells into distinct rectangular grids, each containing a halo of ghost cells for boundary condition implementation and synchronisation, as detailed in [7]. The mesh widths of two adjacent levels follow the relation $\Delta x^c/\Delta x^f = \Delta t^c/\Delta t^f = r$, where the superscript f refers to the fine mesh and c refers to the coarse mesh. The simultaneous refinement of spatial and temporal mesh width is fully consistent with the explicit nature of the LBM. By implementing the LBM operations using cell-centred data structures, the method integrates naturally with the recursive AMR framework. Additional implementation information on adaptive LBM within the AMROC framework is available in [8].

The multi-level grid structure requires proper transfer of distribution functions across refinement boundaries. When transferring from fine to coarse grids, distribution functions of fine cells surrounding a coarse cell are averaged. The transition from coarse to fine involves both spatial and temporal interpolations to create the fine distribution functions from surrounding coarse cells [8]. Distribution functions between levels are further modified by the approach described by Dupuis and Chopard [9], where the non-equilibrium part is rescaled as

$$f_\alpha^c(\mathbf{x},t) = f_\alpha^{f\to c(0)}(\mathbf{x},t) + \frac{r\omega^f}{\omega^c} f_\alpha^{f\to c(1)}(\mathbf{x},t), \tag{4}$$

$$f_\alpha^f(\mathbf{x},t) = f_\alpha^{c\to f(0)}(\mathbf{x},t) + \frac{\omega^c}{r\omega^f} f_\alpha^{c\to f(1)}(\mathbf{x},t), \tag{5}$$

and ω^c and ω^f are the non-dimensional relaxation frequencies of the coarse and fine level, respectively. This approach ensures the continuity in density and velocity, which depends on the equilibrium part, as well as on the strain-rate tensor $\mathbf{S}_{ij}$.

2.3 Large eddy simulation

To make turbulent flow simulations computationally feasible at technically relevant Reynolds numbers, the small scales of turbulence are not explicitly resolved but instead modelled using large eddy simulation. In this work, the Smagorinsky model [10] is employed, which accounts for the effects of unresolved turbulent eddies through an eddy viscosity term ν_t added to the kinematic viscosity ν. The eddy viscosity is computed from the filtered strain rate tensor $\overline{\mathbf{S}}_{ij}$ as

$$\nu_t = (C_{sm}\Delta x)^2 |\overline{S}|, \tag{6}$$

where the coefficient $C_{sm}(\mathbf{x},t)$ is locally evaluated and $|\overline{S}| = \sqrt{2\overline{\mathbf{S}}_{ij}\overline{\mathbf{S}}_{ij}}$. The evaluation of $\overline{\mathbf{S}}_{ij}$ follows the consistent strain formulation introduced by Malaspinas and Sagaut [11]. The total effective viscosity in the simulation combines the molecular

viscosity ν with the turbulent contribution ν_t, consequently modifying the relaxation parameter ω in the numerical scheme as

$$\omega^{\star} = \frac{c_s^2 \Delta t}{(\nu + \nu_t) + c_s^2 \Delta t/2}. \tag{7}$$

2.4 Boundary conditions

The block-structured AMR framework naturally handles boundary conditions through ghost cells. Non-Cartesian boundaries are represented on the cell-based Cartesian mesh by a scalar level set function that stores the distance to the boundary surface. A fluid cell is treated as an embedded ghost cell if the distance value in the cell centre is negative. For computing signed distance functions from triangulated surface meshes, a specially designed algorithm [7] is used. For ghost cells at non-Cartesian boundaries, the no-slip boundary condition is applied by imposing zero velocity at the boundary location. This is achieved through an interpolated bounce-back scheme following Bouzidi *et al.* [12] using linear interpolation. Prior to the streaming step, each discrete lattice distribution function that would propagate from a ghost cell into a fluid cell is appropriately replaced using

$$f_{\alpha}(\mathbf{x_g}, t) = 2q f_{\bar{\alpha}}(\mathbf{x_g} + \mathbf{e}_{\alpha}, t) + (1 - 2q)\, f_{\bar{\alpha}}(\mathbf{x_g} + 2\mathbf{e}_{\alpha}, t), \quad q < 0.5, \tag{8}$$

and

$$f_{\alpha}(\mathbf{x_g}, t) = \frac{1}{2q} f_{\bar{\alpha}}(\mathbf{x_g} + \mathbf{e}_{\alpha}, t) + \frac{(2q-1)}{2q} f_{\alpha}(\mathbf{x_g} + \mathbf{e}_{\alpha}, t), \quad q > 0.5, \tag{9}$$

where the parameter q is calculated as $q = ||\mathbf{x_f} - \mathbf{x_w}||/||\mathbf{x_f} - \mathbf{x_g}||$ and $\mathbf{e}_{\bar{\alpha}}$ is the direction opposite to $\mathbf{e}_{\alpha}$.

2.5 Mass flow rate conservation

The numerical simulations conducted in this work employ periodic boundary condition in streamwise direction. To maintain a constant mass flow rate throughout the computational domain, a volumetric force is applied exclusively in the upper section of the channel, which remains unobstructed by the porous medium. This forcing term is incorporated during the collision step, as described in Eq. (2), and is computed according to the discrete forcing scheme developed by Guo *et al.* [13] as

$$F_{\alpha}(\mathbf{x}, t) = \left(1 - \frac{\omega}{2}\right) w_{\alpha} \left[\frac{\mathbf{e}_{\alpha} - \mathbf{u}/c}{\tilde{c}_s^2} + \frac{(\mathbf{e}_{\alpha} \cdot \mathbf{u}/c)}{\tilde{c}_s^4}\right] \mathbf{F}. \tag{10}$$

The force $\mathbf{F}$ is calculated in an iterative way using

$$\mathbf{F} = \rho \left(\frac{u_{\tau}^2}{H} \frac{\Delta t^2}{\Delta x} + \frac{(< U_b > - < U >_s)}{\Delta t} \frac{\Delta t^2}{\Delta x}\right). \tag{11}$$

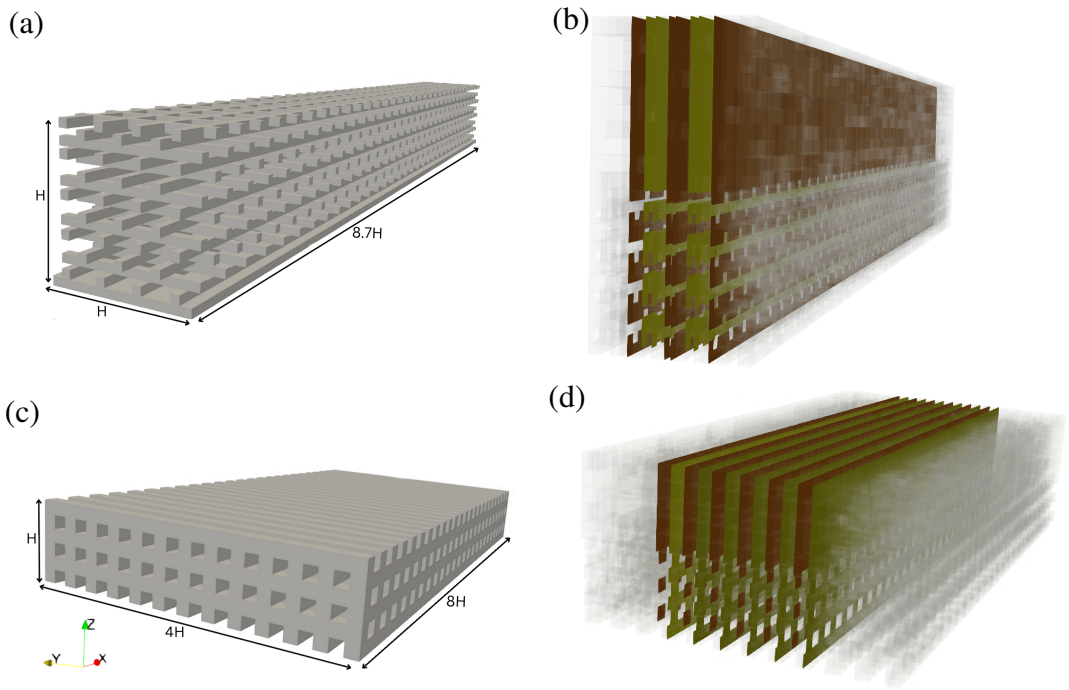

Figure 1: (a) Suga's porous test section, (c) present porous test section and (b, d) streamwise slices for statistical averaging.

Here, u_τ is the friction velocity of the fully developed flow, H is the height of the channel free flow, $< U_b >$ is the channel free flow target bulk velocity, $< U >_s$ is the instantaneous spatial average of the streamwise (x-direction) velocity in the free flow region, recomputed each iteration on the coarsest grid level.

3 Porous flow simulation

3.1 Computational setup

Two porous flow configurations are considered to demonstrate the fidelity of the current solver. The first configuration uses the porous flow setup of Suga *et al.* [14] for methodology validation, while the second configuration corresponds to an ongoing collaborative experimental study. Both configurations feature flow through rectangular conduits partially filled with porous materials, although with different geometric structures. The porous test sections and streamwise slices from $-L_y/2$ to $L_y/2$ of the flow domains are illustrated in Figure 1. For Suga's configuration (Figure 1a), the rectangular duct dimensions are $L_x \times L_y \times L_z = 8.7H \times H \times 2H$ in the streamwise, spanwise, and vertical directions, respectively. Similarly, the current configuration (Figure 1c) has dimensions $8H \times 4H \times 2H$. A periodic boundary condition is applied in the streamwise direction. No-slip conditions are applied at the duct walls

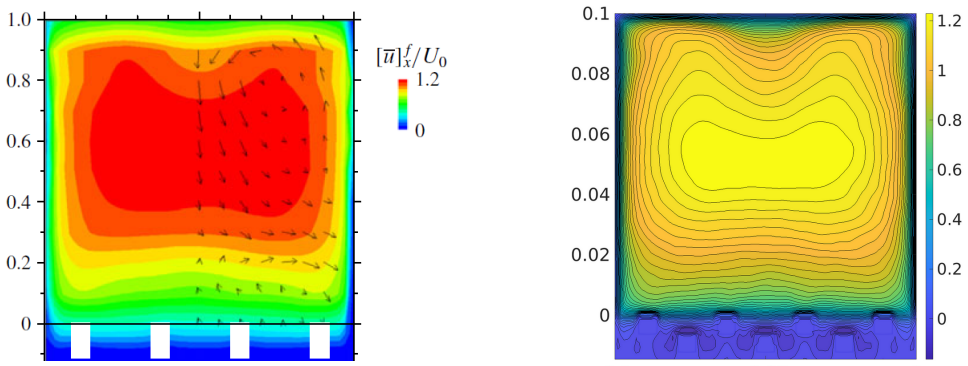

Figure 2: Comparison of contour plot for mean streamwise velocity of current work (right) with published work [14] (left) at Y-Z plane.

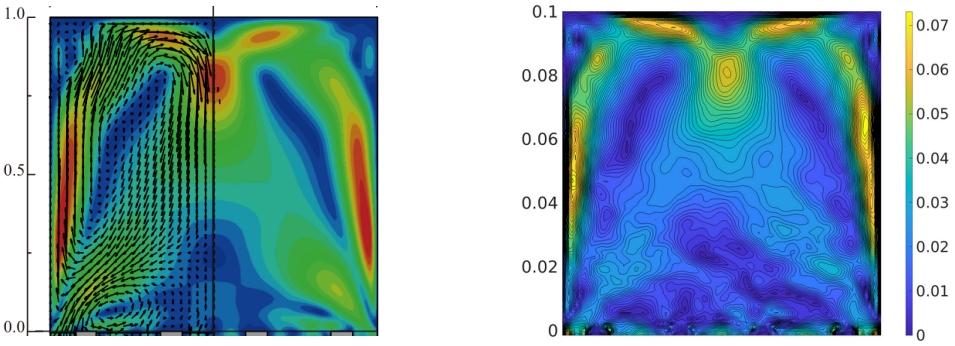

Figure 3: Comparison of contour plot for secondary flow intensity of current work (right) with published work [3] (left) at Y-Z plane.

and porous structure by using the half-way bounce-back method and the interpolated bounce-back Bouzidi scheme (see Section 2.4), respectively.

3.2 Validation test problem

To validate our approach, Suga's test case is simulated without considering the effect of temperature difference between the top and bottom walls, while the side walls remain adiabatic. The porous medium consists of square bars of size $D/H = 0.06$, and the distance between the bar centres is $\ell = 4.33D$. The porosity (ϕ) and non-dimensional permeability (K) are 0.77 and 7.6×10^{-5}, respectively. For our simulation, we used bar size $D = 6$ mm, clear fluid height $H = 10$ cm, domain length $L = 87$ cm, total height of test section 20 cm, and spanwise distance 10 cm. The bulk velocity for present simulation is fixed at 10 m/s to maintain the flow at a Reynolds number of $Re = U_b H/\nu = 7400$.

A comparative analysis was conducted to evaluate the similarity between the mean streamwise velocity contours obtained from the present simulation ($Re = 7400$) and experimental results reported by [14] at $Re = 7700$. The results reveal the formation of a single primary vortex structure, with its core located at the centroid of the

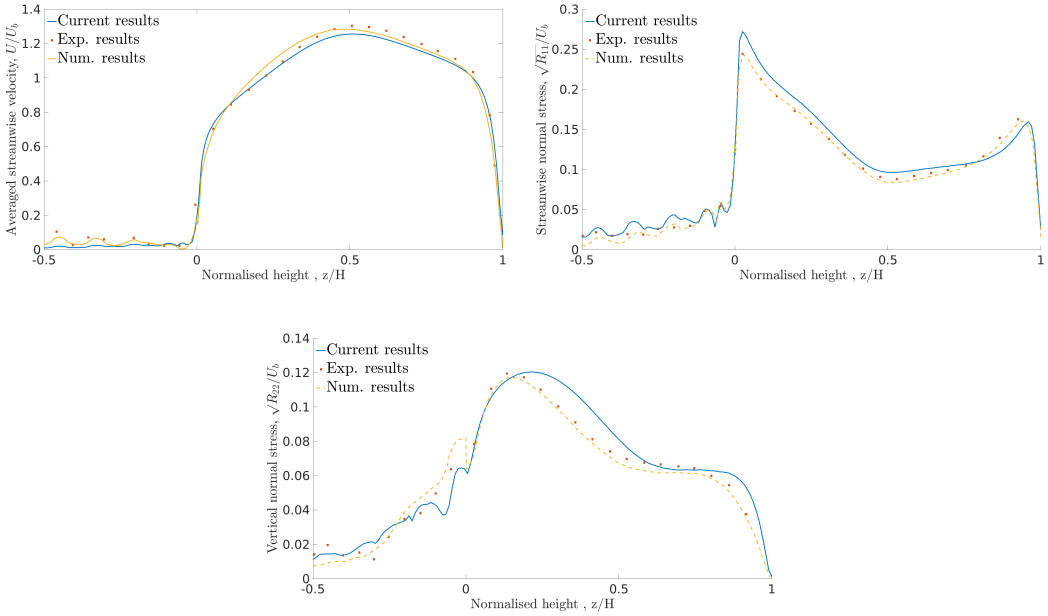

Figure 4: Comparison of time and spatial averaged velocity profile (top left), streamwise normal Reynolds stress (R_{11}) (top right), and vertical normal Reynolds stress (R_{22}) (bottom) with Suga's numerical and experimental results.

clear fluid domain, irrespective of the Reynolds number. In contrast to confined duct flows bounded by impermeable walls [15], the current configuration exhibits distinct flow characteristics due to wall permeability. Notably, secondary vortices are absent in the present case, as fluid momentum can dissipate through the porous boundaries (Figure 2). The velocity contour demonstrates a characteristic depression near the upper symmetry plane, indicative of downward fluid motion. This downward flow is subsequently redirected upward along the vertical walls following interaction with the porous wall, demonstrating the significant influence of permeability on bulk flow patterns. We performed a qualitative comparison (Figure 3) of the secondary flow between our simulation at $Re = 7400$ and the results of [3] at $Re = 3500$. It is interesting to observe comparable patterns between the current results with reference DNS data, despite the difference in Reynolds number, indicating the weak Reynolds number dependence of secondary flows in similar geometries at fully turbulent regimes. The secondary flow, a characteristic feature of turbulent duct flows, was visualized using contours of the cross-flow velocity magnitude defined as $\sqrt{\overline{v}^2 + \overline{w}^2}/U_b$. The peak secondary flow intensities are approximately 7% and 6% of the bulk velocity for current results and the reference data, respectively.

Figure 4 presents a quantitative comparison of turbulence statistics between the current simulation and published experimental data [3, 14] at $Re = 7400$. The analysis includes both time- and spatially averaged velocity profiles and normal Reynolds stress components. Spatial averaging was performed on the slices as shown in Figure 1b. The simulation results demonstrate satisfactory agreement with the reference experimental data, particularly in the clear fluid region.

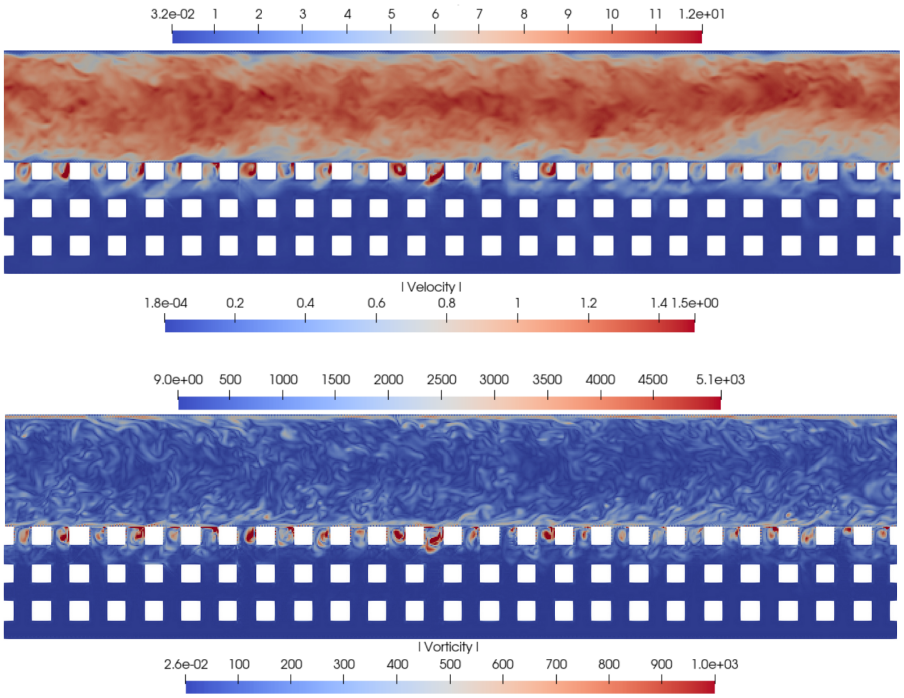

Figure 5: Contour of instantaneous velocity vector magnitude (top), vorticity vector magnitude (bottom) at X-Z plane.

It should be noted that the current computational model does not account for conjugate heat transfer effects. This simplification contributed to the observed discrepancies in the porous region, where thermal effects are most pronounced. However, in the clear fluid region, the model shows reasonable agreement, since the reference work [3] similarly neglected buoyancy effects.

3.3 Present test problem

This section deals with the preliminary investigation of turbulent flow through a porous medium composed of periodic square bars with dimensional characteristics that correspond to the configuration of ongoing collaborative experimental work. The porous structure features square bars of side length $D = 12.7$ mm arranged with a centre-to-centre spacing of $\ell = 2D = 25.4$ mm, yielding a porosity $\phi = 0.486$. The computational domain replicates the experimental dimensions with a clear fluid height $H = 76.2$ mm, streamwise length $L = 608$ mm, and spanwise width of 304 mm.

For the half-blockage configuration, the Reynolds number based on porous block height reaches $40,371$ at the maximum bulk velocity of 8 m/s. The permeability is estimated as $K = 4.67 \times 10^{-7}$ mm^2 using the Ergun equation $K = \frac{d_p^2 \phi^3}{150(1-\phi)^2}$, where d_p represents the porous particle diameter equivalent to side length. The fluid properties correspond to air at 293 K with a fixed Prandtl number $Pr = 0.71$. To characterize the turbulent flow dynamics, Figure 5 illustrates contour maps of instantaneous velocity and vorticity vector magnitudes while using separate data range scaling for the clear

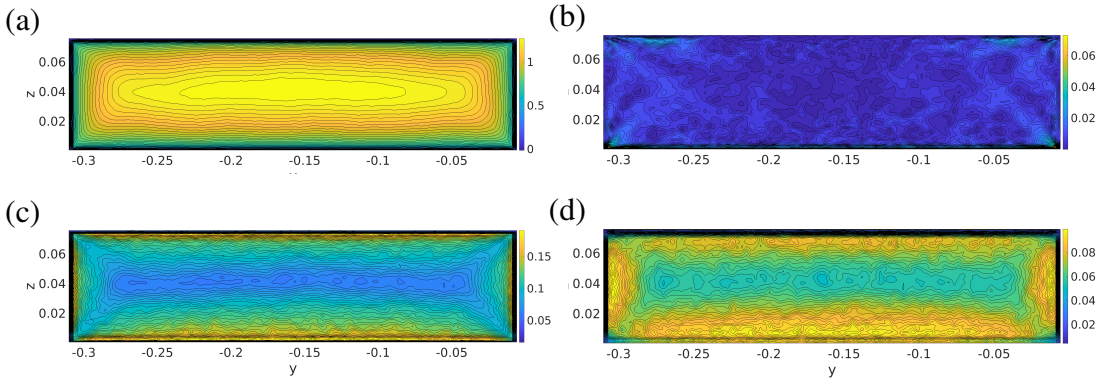

Figure 6: Contour of (a) mean streamwise velocity, (b) secondary flow intensity, (c) streamwise velocity fluctuations, and (d) vertical velocity fluctuations.

fluid and porous regions. The unobstructed fluid domain exhibits developed turbulent flow features, evidenced by high-intensity velocity fluctuations and vortex structures. Limited flow penetration is observed in the first layer of the porous matrix, attributable to the low porosity ($\phi = 0.486$) and consequent high flow resistance.

Quantitative analysis reveals a peak velocity magnitude of 12.0 ± 0.5 m/s in the clear fluid region, representing a 50% overshoot relative to the prescribed bulk velocity 8 m/s. This significant velocity enhancement arises from flow acceleration in the core region, driven by wall confinement effects of the sidewalls and top wall to maintain mass conservation. Within the porous medium, the velocity field decays sharply to a maximum of 1.5 ± 0.2 m/s. The vorticity field shows analogous behaviour, with peak magnitudes of 5100 ± 200 s^{-1} in the clear fluid region, compared to attenuated values of 1000 ± 50 s^{-1} within the porous matrix. The presence of vortex structures along the porous boundary layer indicates partial preservation of turbulent features, although at significantly reduced intensity.

Figure 6 portrays the contours of the normalized streamwise mean velocity $\overline{u}/U_b$ and the secondary flow intensity $\sqrt{\overline{v}^2 + \overline{w}^2}/U_b$ along with the turbulence intensities. The turbulence intensities are quantified through the root-mean-square (rms) of velocity fluctuations: $u'_{\text{rms}} = \sqrt{\overline{u'^2}}$ (streamwise) and $v'_{\text{rms}} = \sqrt{\overline{v'^2}}$ (vertical), where u' and v' represent the fluctuating velocity components. Similar to Suga's case, a large single primary vortex is observed in clear fluid section without any secondary vortex flow as seen in Figure 6a. Figure 6b shows a key difference in secondary flow topology between solid and porous walls. The classic eight-vortex pattern in solid-wall ducts [15] disappears entirely under porous-wall conditions. The peak secondary flow intensities were approximately 8.75% of the bulk velocity (8 m/s). This indicates enhanced momentum transport through the porous interface in comparison to the solid-wall reference case.

Figures 6c and 6d demonstrate the three-dimensional wall effects on turbulent fluctuations, presenting the rms of both streamwise (u'_{rms}) and vertical (v'_{rms}) velocity

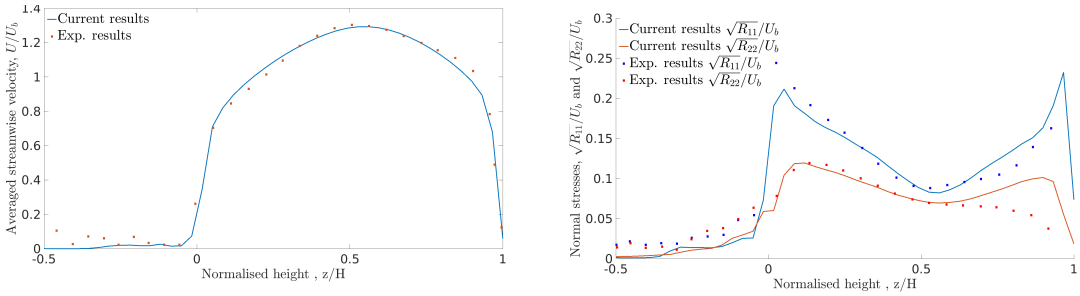

Figure 7: Comparison of turbulent flow statistics: time and spatial averaged velocity profile (left), and normal Reynolds stress components (right).

components normalized by the bulk mean velocity (U_b). The results reveal a significant enhancement of turbulent characteristics above the porous wall, in agreement with previous findings for porous-walled turbulent channel flows [3, 15]. This intensification is attributed to increased flow penetration and unsteady momentum exchange at the porous interface. Interestingly, the vortex structure exhibits some similarities with that of solid duct flows [15], due to the low porosity and the orderly configuration of the solid blocks.

Figure 7 presents the time and spatially averaged velocity and normal Reynolds stress components. Spatial averaging was performed over the slices as shown in Figure 1d. The comparison is made with [14] to demonstrate that the current flow characteristics follow similar trends while exhibiting distinct features due to differences in Reynolds number, porous structure, and flow geometry.

4 Scalability test

A scalability analysis was also performed to evaluate the parallel performance of the current implementation, as presented in Figure 8. The study examined two distinct computational approaches: a uniform mesh configuration and an adaptive mesh refinement (AMR) scheme with three levels of hierarchical refinement. The dynamic mesh adaptation employed a vorticity magnitude threshold of $400.0\ \mathrm{s}^{-1}$ for the domain configuration described in Section 3.2. Strong scaling benchmarks conducted on the ARCHER2 system utilized a uniform mesh of $1392 \times 160 \times 320$ cells (7.127×10^7 total cells), with performance metrics derived from averaged wall-clock times per timestep during the initial 80 iterations.

The adaptive simulations initiated from a base grid of $174 \times 20 \times 40$ cells, with dynamic addition of up to three refinement levels while refined in all spatial directions by factor 2. Although the speed-up achieved by AMROC decreases with increasing CPU count compared to the uniform mesh case, the adaptive approach provides net computational savings in terms of the wall-clock time. Figure 8(b) presents a detailed breakdown of computational costs, showing the averaged wall-clock times per time

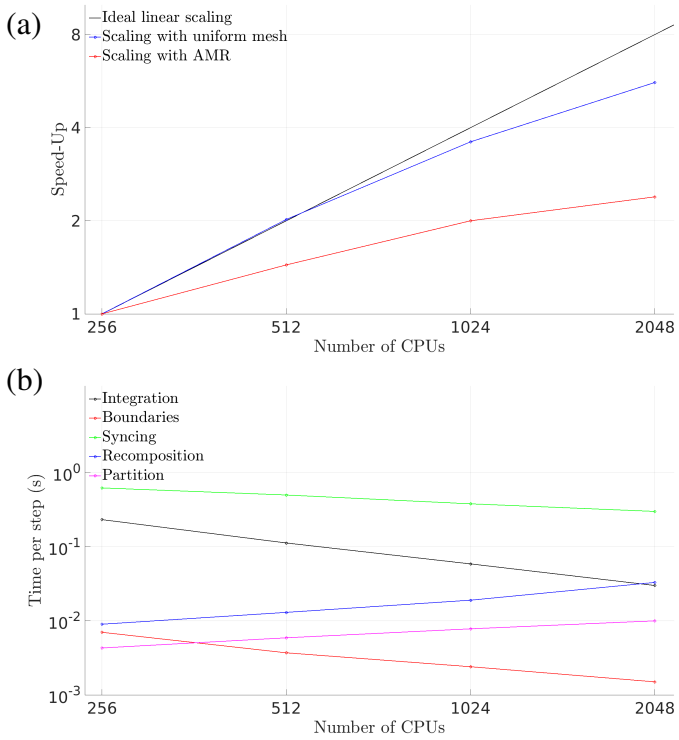

Figure 8: (a) Scalability of AMROC with or without refinement comparing with ideal linear scaling, (b) scaling of main operations.

step for different operations. The analysis reveals that, with the exception of partitioning and grid recomposition, the cost of the majority of operations decreases almost linearly as the number of CPU cores increases. Interestingly, with increasing core counts, the recomposition phase becomes a major computing barrier and one of the main causes of the simulation's total cost.

5 Conclusions

This study demonstrates the effectiveness and robustness of the adaptive AMROC-LBM solver integrated with large eddy simulation for modelling turbulent flows in porous media. By validating the numerical framework against benchmark experimental and numerical data, the solver has shown good accuracy in capturing both mean flow structures and turbulence statistics within porous configurations. The results reveal that there are an accelerated streamwise velocity and vortex structures in the clear fluid region and attenuated turbulence penetration within the porous matrix. The scalability analysis confirms the solver's capability to perform large-scale simulations efficiently on parallel architectures, particularly when leveraging adaptive mesh strategies. Although the current model does not incorporate conjugate heat transfer effects, its ability to resolve key hydrodynamic features of porous flows positions it as a valuable tool for future investigations. Further work will extend the solver to include

thermal coupling and explore broader classes of porous geometries.

References

[1] B. D. Wood, X. He, S.V. Apte, "Modeling Turbulent Flows in Porous Media", Annual Review of Fluid Mechanics, 52, 171-203, 2020.

[2] Y. Jin, A. Kuznetsov, "Turbulence modeling for flows in wall bounded porous media: an analysis based on direct numerical simulations", Physics of Fluids, 29, 045102, 2017.

[3] Y. Kuwata, K. Tsuda, K. Suga, "Direct numerical simulation of turbulent conjugate heat transfer in a porous-walled duct flow", Journal of Fluid Mechanics, 904, A9, 2020.

[4] S. Hou, J. Sterling, S. Chen, G. D. Doolen, "A lattice Boltzmann subgrid model for high Reynolds number flows", Fields Institute Communications, 6, 1994.

[5] M. Grondeau, R. Deiterding., "Direct prediction of flow noise around airfoils using an adaptive lattice Boltzmann method", Handbook of Wind Energy Aerodynamics, 25 pages, Springer, (B. Stoevesandt, G. Schepers, P. Fuglsang, Y. Sun, ed.), 2021.

[6] O. Malaspinas, "Increasing stability and accuracy of the lattice Boltzmann scheme: recursivity and regularization", 2015.

[7] R. Deiterding, "Block-structured adaptive mesh refinement - theory, implementation and application", European Series in Applied and Industrial Mathematics: Proceedings, 34, 97–150, 2011.

[8] R. Deiterding, S.L. Wood, "An adaptive lattice Boltzmann method for predicting wake fields behind wind turbines", in A. Dillmann, G. Heller, E. Krämer, C. Wagner, C. Breitsamter (Editors), New Results in Numerical and Experimental Fluid Mechanics X, Volume 132, pages 845–857. Springer, 2016.

[9] A. Dupuis, B. Chopard, "Theory and applications of an alternative lattice Boltzmann grid refinement algorithm", Physical Review E, 67(6), 066707, 2003.

[10] J. Smagorinsky, "General circulation experiments with the primitive equations", Monthly Weather Review, 91(3), 99–163, 1963.

[11] O. Malaspinas, P. Sagaut, "Consistent subgrid scale modelling for lattice Boltzmann methods", Journal of Fluid Mechanics, 700, 514–542, 2012.

[12] M. Bouzidi, M. Firdaouss, P. Lallemand, "Momentum transfer of a Boltzmann-lattice fluid with boundaries", Physics of Fluids, 13, 3452, 2001.

[13] Z. Guo, B.Shi, C. Zheng, "A coupled lattice BGK model for the Boussinesq equations", Int. J. Numerical Methods in Fluids, 39, 325–342, 2002.

[14] K. Suga, Y. Okazaki, Y. Kuwata, "Characteristics of turbulent square duct flows over porous media", Journal of Fluid Mechanics, 884, A7, 2020.

[15] A. Samanta, R. Vinuesa, I. Lashgari, P. Schlatter, L. Brandt , "Enhanced secondary motion of the turbulent flow through a porous square duct", Journal of Fluid Mechanics, 784, 681–693, 2015.