A Reproducible D3Q19 Multiple-Relaxation-Time Lattice Boltzmann Benchmark and Quantum-Operator Audit for Forced Wall-Bounded Flow

Citation for the original published paper (version of record):
Khan, M., Yao, H. (2026). A Reproducible D3Q19 Multiple-Relaxation-Time Lattice Boltzmann Benchmark and Quantum-Operator
Audit for Forced Wall-Bounded Flow Simulations. Fluids, 11(7).
http://dx.doi.org/10.3390/fluids11070175

(article starts on next page)

fluids

MDPI

Article

A Reproducible D3Q19 Multiple-Relaxation-Time Lattice Boltzmann Benchmark and Quantum-Operator Audit for Forced Wall-Bounded Flow Simulations

Muhammad Idrees Khan * and Hua-Dong Yao

Department of Mechanical Engineering, Chalmers University of Technology, SE-41296 Gothenburg, Sweden; huadong.yao@chalmers.se
* Correspondence: idreesm@chalmers.se

Abstract

Quantum algorithms for flow simulation are advancing rapidly, but reproducible wall-bounded benchmarks with classical reference data are still needed to evaluate future quantum and hybrid quantum-classical solvers. This work presents a forced D3Q19 multiple-relaxation-time (MRT) lattice-Boltzmann method (LBM) benchmark for body-force-driven Poiseuille flow in a three-dimensional channel. The solver combines periodic streamwise and spanwise boundaries, halfway bounce-back walls, moment-space relaxation, and body-force forcing with the half-force velocity correction. The solution is verified against the analytical parabolic profile using relative L_2 and maximum profile errors, mass conservation, extrapolated wall slip, and wall-normal leakage. A verification study over grid resolution, relaxation time, forcing strength, and initialization demonstrates second-order grid convergence and robust conservation behavior. The verified timestep is then decomposed into quantum-relevant primitives, including streaming, wall reflection, moment transformation, MRT relaxation, equilibrium evaluation, forcing, macroscopic recovery, and measurement. The resulting benchmark connects flow-solver accuracy metrics with operator-level requirements for quantum implementation, providing a compact reference problem for future quantum processing unit (QPU)-assisted, hybrid quantum-classical, and quantum-linear-solver-based computational fluid dynamics (CFD) studies. Performance gains over classical LBM execution are not assessed here.

Keywords: quantum computing; computational fluid dynamics; scientific computing; lattice Boltzmann method; D3Q19; multiple-relaxation-time collision; quantum linear solvers; wall-bounded flow; quantum computational fluid dynamics

Academic Editors: Surya Pratap Vanka and Ramesh Agarwal

Received: 4 June 2026
Revised: 4 July 2026
Accepted: 8 July 2026
Published: 10 July 2026

1. Introduction

Computational fluid dynamics (CFD) relies on large-scale numerical solvers whose cost grows rapidly with spatial resolution, physical complexity, and Reynolds number. Quantum computing has therefore attracted interest as a possible route toward new simulation strategies for transport and flow problems. The field is still young. Recent reviews describe quantum CFD as a promising direction that requires careful benchmark design, rigorous validation, and a realistic treatment of nonlinear and dissipative operators [1]. These points are especially relevant for lattice-based kinetic methods, where the structure of the algorithm is close to a sequence of local transformations and lattice shifts.

The lattice Boltzmann method (LBM) evolves particle populations on a discrete velocity set and recovers hydrodynamic behavior through a kinetic update [2–4]. Its streaming and collision structure has made it attractive for parallel classical computing and for early quantum-fluid ideas. Quantum lattice-gas and quantum lattice-Boltzmann approaches have been explored for more than two decades [5]. Recent work has renewed this interest by proposing quantum lattice-Boltzmann, dynamic-circuit LBM, and quantum Carleman lattice-Boltzmann formulations for transport and nonlinear flow simulation [6–9]. Quantum linear-system algorithms also motivate hybrid approaches in which nonlinear dynamics are lifted or linearized before solution on quantum subroutines [10,11].

Beyond its use as a classical CFD solver, LBM is increasingly used as a flexible platform for modern flow-simulation strategies. Recent examples include physics-constrained machine-learning closures for LBM large-eddy simulation, reinforcement-learning control of coarse-grained lattice-Boltzmann closures, high-performance graphics processing unit (GPU)-oriented LBM formulations for turbulent flows, and complex-geometry applications ranging from engineering and biological flows to turbulence studies [12–17]. This breadth makes LBM a useful setting in which to define reproducible benchmarks that connect classical verification metrics with quantum-operator requirements.

A practical obstacle remains. Many quantum CFD proposals require clean classical reference problems that include boundary conditions, forcing, and reproducible error metrics. Periodic cases are useful, yet wall-bounded flows are more representative of engineering CFD. The present work addresses this benchmark need through a reproducible forced wall-bounded D3Q19 multiple-relaxation-time (MRT) LBM reference problem with clear diagnostics for future quantum, hybrid quantum-classical, and quantum-linear-solver-based flow studies.

Existing quantum-flow studies have made important progress in quantum lattice-gas models, quantum lattice-Boltzmann algorithms, Carleman-linearized LBM, and non-linear quantum LBM simulations [5,6,9,18,19]. Dynamic-circuit quantum lattice Boltzmann method (QLBM) work advances timestep circuits for linearized advection–diffusion models [7,8], Carleman-based studies provide lifted collision circuits at moderate Reynolds number [9,11,19], and recent nonlinear QLBM formulations target strongly nonlinear fluid dynamics at the algorithm level [6,18]. Those lines of work center on circuit construction, lifting, or proof-of-principle quantum timestep designs. The present study complements them with a reproducible forced wall-bounded D3Q19 MRT benchmark, classical validation metrics, and an operator-level quantum audit tied to the same timestep used in the verified classical solver.

The solver uses a verified D3Q19 MRT-LBM formulation for body-force-driven channel flow with a consistent forcing treatment. The collision operator follows the MRT formulation, which provides separate relaxation rates for kinetic moments and has been widely used for improved stability relative to single-relaxation-time models [20,21]. Halfway bounce-back is used to impose the two channel walls, following common LBM practice for no-slip wall-bounded LBM simulations [3,22,23].

The contribution of this paper is threefold. First, it defines a reproducible forced D3Q19 MRT-LBM benchmark for wall-bounded Poiseuille flow, including analytical validation and machine-readable error metrics. Second, it verifies the benchmark through grid refinement, relaxation-time variation, forcing-strength variation, and initial-condition tests, thereby providing performance-relevant accuracy and robustness data. Third, it translates the verified timestep into quantum-relevant operator primitives, identifying which parts map to reversible streaming and bounce-back circuits and which parts require block encoding, dissipative-channel treatment, finite-precision arithmetic, or measurement-heavy readout.

Relative to the QLBM, Carleman, and dynamic-circuit studies cited above, the missing benchmark elements addressed here are a forced three-dimensional D3Q19 MRT wall-bounded channel, halfway bounce-back walls, body-force forcing, reproducible classical verification data, and diagnostics for profile error, mass conservation, wall slip, and wall-normal leakage. Existing quantum-LBM studies have mainly emphasized circuit construction, linearized transport, advection–diffusion models, lifted formulations, or proof-of-principle nonlinear updates, rather than a fully specified forced D3Q19 MRT wall-bounded reference case with parameter sweeps and classical error targets. The present benchmark is therefore designed to support and test several circuit-level components separately: reversible streaming circuits, boundary-controlled bounce-back/reflection circuits, local block-encoded moment-transform circuits, dissipative or channel-based MRT-relaxation circuits, finite-precision equilibrium/forcing subroutines, and measurement strategies for macroscopic observables. Thus, the contribution is not a quantum speedup claim, but a validated classical reference and operator-level circuit target for future quantum processing unit (QPU)-assisted and hybrid quantum-classical CFD studies.

2. Numerical Method

2.1. D3Q19 Lattice and Macroscopic Fields

The simulations use the D3Q19 velocity set with $Q = 19$ particle populations $f_i(\mathbf{x}, t)$, indexed by $i \in \{0, \ldots, 18\}$ in the order used by the reference solver. The lattice velocities are

$$\mathbf{c}_i = \begin{cases} (\pm 1, 0, 0),\ (0, \pm 1, 0),\ (0, 0, \pm 1), & i = 0, \ldots, 5, \\ (\pm 1, \pm 1, 0),\ (0, \pm 1, \pm 1),\ (\pm 1, 0, \pm 1), & i = 6, \ldots, 17, \\ (0, 0, 0), & i = 18, \end{cases} \tag{1}$$

with weights $w_i = 1/18$ for $i = 0, \ldots, 5$, $w_i = 1/36$ for $i = 6, \ldots, 17$, and $w_{18} = 1/3$. Table A1 in Appendix A lists each link explicitly. The discrete sound speed is

$$c_s^2 = \frac{1}{3}. \tag{2}$$

The density is computed as

$$\rho = \sum_{i=0}^{18} f_i. \tag{3}$$

The velocity includes the standard half-force correction used in consistent forcing schemes,

$$\mathbf{u} = \frac{\sum_{i=0}^{18} \mathbf{c}_i f_i + \frac{1}{2}\mathbf{F}}{\rho}. \tag{4}$$

For the present channel case the body force is spatially uniform and acts in the streamwise direction,

$$\mathbf{F} = (F_x, 0, 0). \tag{5}$$

2.2. Equilibrium Populations

The low-Mach D3Q19 equilibrium distribution is

$$f_i^{eq} = \rho w_i \left[1 + \frac{\mathbf{c}_i \cdot \mathbf{u}}{c_s^2} + \frac{(\mathbf{c}_i \cdot \mathbf{u})^2}{2c_s^4} - \frac{\mathbf{u} \cdot \mathbf{u}}{2c_s^2} \right]. \tag{6}$$

With $c_s^2 = 1/3$, Equation (6) gives the familiar coefficients 3, 4.5, and 1.5.

2.3. MRT Collision

The MRT collision step is performed in moment space. Let $\mathbf{f} = (f_0, \ldots, f_{18})^T$ be the population vector at a lattice node and let M be the D3Q19 moment matrix defined in Appendix A. The moment vector is

$$\mathbf{m} = M\mathbf{f}, \qquad \mathbf{m} = (\delta\rho, e, \epsilon, j_x, q_x, j_y, q_y, j_z, q_z, 3p_{xx}, 3\pi_{xx}, p_{ww}, \pi_{ww}, p_{xy}, p_{yz}, p_{xz}, m_x, m_y, m_z)^T, \tag{7}$$

where $\delta\rho$ is the density moment, e and ϵ are energy moments, j_x, j_y, and j_z are momentum moments, q_x, q_y, and q_z are heat-flux moments, $p_{\alpha\beta}$ and $\pi_{\alpha\beta}$ are stress moments, and m_x, m_y, and m_z are higher-order kinetic moments. The diagonal relaxation matrix is

$$S = \mathrm{diag}(s_0, s_1, \ldots, s_{18}), \tag{8}$$

with the entries given in Table A2. The conserved density and momentum moments have zero relaxation. The shear-related moments use

$$\nu = \frac{\tau - 1/2}{3}, \qquad s_\nu = \frac{1}{3\nu + 1/2}, \tag{9}$$

for moments $3p_{xx}$, p_{ww}, p_{xy}, p_{yz}, and p_{xz}. The remaining kinetic relaxation rates follow the standard D3Q19 MRT values used in the reference implementation. The equilibrium moment vector $\mathbf{m}^{eq}$ is computed from ρ and $\mathbf{u}$.

2.4. Forcing Scheme

Body forces are incorporated through the Guo forcing scheme [24], applied here in moment space; the text below refers to this as body-force forcing. Here $\mathbf{F}$ denotes the macroscopic body-force vector and Φ_i denotes the discrete Guo source term at lattice link i. The discrete source term is

$$\Phi_i = w_i \left[\frac{\mathbf{c}_i - \mathbf{u}}{c_s^2} + \frac{(\mathbf{c}_i \cdot \mathbf{u})\mathbf{c}_i}{c_s^4} \right] \cdot \mathbf{F}. \tag{10}$$

The source term is transformed to moment space as

$$\mathbf{\Phi}_m = M\mathbf{\Phi}, \tag{11}$$

where $\mathbf{\Phi} = (\Phi_0, \ldots, \Phi_{18})^\mathsf{T}$. The post-collision moment vector is then computed as

$$\mathbf{m}^{post} = \mathbf{m} - S(\mathbf{m} - \mathbf{m}^{eq}) + \left(I - \frac{S}{2}\right)\mathbf{\Phi}_m. \tag{12}$$

For conserved moments, the forcing increment is applied directly. The post-collision populations are recovered through

$$\mathbf{f}^{post} = M^{-1}\mathbf{m}^{post}. \tag{13}$$

2.5. Streaming and Wall Treatment

The channel is periodic in x and z directions. Streaming shifts every population along its lattice direction. At the lower and upper walls in y direction, halfway bounce-back reflects outgoing populations into their opposite directions, in line with standard no-slip wall treatments used in lattice-Boltzmann channel flows [3,22,23,25]. The opposite direction index is denoted by $\bar{i}$. For a link crossing a wall, the reflected population is assigned at the same boundary fluid node as

$$f_i(\mathbf{x}, t+1) = f_{\bar{i}}^{post}(\mathbf{x}, t). \tag{14}$$

This gives a simple wall model suitable for the present Poiseuille verification problem.

2.6. Analytical Reference Solution

The flow is driven by a constant streamwise body force between two stationary walls. With halfway bounce-back, the physical walls are placed at $y = 0$ and $y = H$, while fluid nodes lie at $y_j = j + 1/2$ for $j = 0, 1, \ldots, N_y - 1$. The analytical velocity profile in lattice units is

$$u_x^{ana}(y) = \frac{F_x}{2\rho_0 \nu} y(H - y), \tag{15}$$

where $H = N_y$ and $\rho_0 = 1$. The target maximum velocity is used to select the forcing strength,

$$F_x = \frac{8\nu u_{\max}}{H^2}. \tag{16}$$

Figure 1 summarizes the timestep order used in the solver: macroscopic recovery, moment transform, equilibrium evaluation, MRT collision with forcing, inverse transform, streaming, and halfway bounce-back at the channel walls.

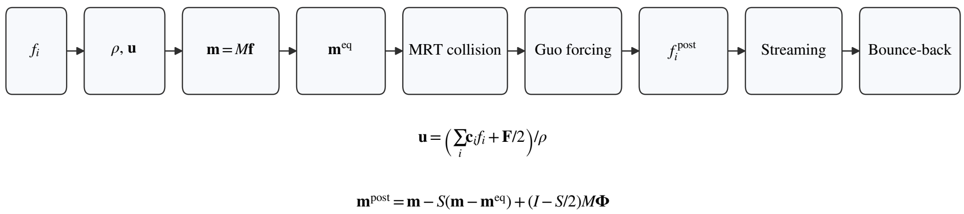

Figure 1. Schematic of the forced D3Q19 multiple-relaxation-time lattice-Boltzmann method (MRT-LBM) timestep used in the channel solver.

3. Verification and Performance Metrics

The verification study comprises eighteen simulations grouped as A1 (baseline), B1–B4 (grid refinement), C1–C6 (relaxation-time sweep), D1–D5 (forcing sweep), and E1–E2 (initialization). Table 1 summarizes the design. The baseline case A1 starts from rest on the reference grid. The four B cases refine the wall-normal resolution with proportional $N_x = 2N_y$ and $N_z = N_y/2$, fixed $\tau = 0.8$, fixed $u_{\max} = 0.02$, and a parabolic start to shorten the transient. The six C cases vary τ on the baseline grid while keeping $u_{\max} = 0.02$ and a parabolic start. The five D cases vary $u_{\max}$ with $\tau = 0.8$ and a parabolic start. Cases E1 and E2 repeat the baseline physical settings but compare rest and parabolic initialization (with different time-marching lengths, as noted in Section 4). In every case the body force follows Equation (16) from the chosen ν, H, and $u_{\max}$.

Table 1. Verification study design (eighteen cases A1–E2). The body force in each case is set from Equation (16).

Study	Cases	Purpose	Grid	Main Parameter	Initial Condition
Baseline	1	Reference validation	$64 \times 32 \times 16$	$\tau = 0.8$, $u_{\max} = 0.02$	rest
Grid refinement	4	Spatial convergence ($N_y = 16, 32, 48, 64$)	$32 \times 16 \times 8$ to $128 \times 64 \times 32$	$\tau = 0.8$, $u_{\max} = 0.02$	parabolic
Relaxation-time sweep	6	Collision-time sensitivity	$64 \times 32 \times 16$	$\tau = 0.55$–1.60, $u_{\max} = 0.02$	parabolic
Forcing sweep	5	Forcing-strength sensitivity	$64 \times 32 \times 16$	$\tau = 0.8$, $u_{\max} = 0.005$–0.080	parabolic
Initialization	2	Initial-condition effect	$64 \times 32 \times 16$	$\tau = 0.8$, $u_{\max} = 0.02$	rest or parabolic

The study is designed to provide reference metrics for future quantum or hybrid quantum-classical LBM implementations. The reported quantities include the analytical velocity-profile error, grid-convergence trend, mass-conservation level, wall-slip diagnostic, wall-normal leakage diagnostic, and parameter-sensitivity behavior.

To clarify the physical regime, we also report Reynolds and Mach numbers for the verification campaign. We define

$$Re_{\max} = \frac{u_{\max} H}{\nu}, \qquad Ma_{\max} = \frac{u_{\max}}{c_s} = \sqrt{3}\, u_{\max}, \tag{17}$$

where $H = N_y$, $\nu = (\tau - 1/2)/3$, and $c_s = 1/\sqrt{3}$. For the parabolic Poiseuille profile, the bulk Reynolds number is $Re_b = (2/3)Re_{\max}$. Table 2 summarizes the resulting ranges. The full campaign spans $Re_{\max} = 1.6$–38.4 and $Ma_{\max} = 0.0087$–0.139. Thus, the benchmark is a laminar wall-bounded test case, with the main validation cases in the low-Mach regime and the highest forcing case included as an upper-Mach sensitivity case. The baseline, grid-refinement, relaxation-time, and initialization cases use $Ma_{\max} \approx 0.035$.

Table 2. Dimensionless parameter ranges for the verification campaign. Here $Re_{\max} = u_{\max} H/\nu$ uses the centerline target velocity, and $Ma_{\max} = u_{\max}/c_s$.

Study	H	ν	$Re_{\max}$	$Ma_{\max}$
Baseline/initialization	32	0.1000	6.4	0.0346
Grid refinement	16–64	0.1000	3.2–12.8	0.0346
Relaxation-time sweep	32	0.0167–0.3667	1.75–38.4	0.0346
Forcing sweep	32	0.1000	1.6–25.6	0.0087–0.139
Full campaign	16–64	0.0167–0.3667	1.6–38.4	0.0087–0.139

The primary error metric is the relative L_2 velocity-profile error,

$$E_{L_2} = \frac{\left[\frac{1}{N_y}\sum_{j=0}^{N_y-1}\left(u_x^{num}(y_j) - u_x^{ana}(y_j)\right)^2\right]^{1/2}}{\left[\frac{1}{N_y}\sum_{j=0}^{N_y-1}\left(u_x^{ana}(y_j)\right)^2\right]^{1/2}}. \tag{18}$$

The maximum profile error is

$$E_\infty = \max_j \left| u_x^{num}(y_j) - u_x^{ana}(y_j) \right|. \tag{19}$$

Mass conservation is reported through the maximum sampled relative mass error,

$$E_m = \max_t \left| \frac{M(t) - M(0)}{M(0)} \right|. \tag{20}$$

Wall slip is estimated by linear extrapolation of the mean streamwise velocity to the physical wall locations,

$$u_{w,b} = 1.5u_x(y_0) - 0.5u_x(y_1), \qquad u_{w,t} = 1.5u_x(y_{N_y-1}) - 0.5u_x(y_{N_y-2}). \tag{21}$$

The reported wall-slip diagnostic is

$$E_{slip} = \max(|u_{w,b}|, |u_{w,t}|). \tag{22}$$

For a streamwise Poiseuille flow the target wall-normal velocity is zero. The wall-normal leakage diagnostic uses the x–z mean of the y velocity component at the near-wall fluid rows,

$$\bar{u}_y(y_j) = \frac{1}{N_x N_z}\sum_{i=0}^{N_x-1}\sum_{k=0}^{N_z-1} u_y(x_i, y_j, z_k), \qquad j = 0, \ldots, N_y - 1, \tag{23}$$

and reports the maximum magnitude at the bottom and top fluid nodes,

$$E_{\text{leak}} = \max\left(\left|\bar{u}_y(y_0)\right|, \left|\bar{u}_y(y_{N_y-1})\right|\right). \tag{24}$$

This quantity measures spurious cross-channel flow at the wall-adjacent lattice rows.

4. Results

4.1. Baseline Validation

Case A1 is the baseline validation simulation in Table 1. It uses the grid $N_x \times N_y \times N_z = 64 \times 32 \times 16$, collision time $\tau = 0.8$ (so $\nu = 0.1$ from Equation (9)), target speed $u_{\max} = 0.02$, and a rest start for 20,000 timesteps. Equation (16) gives $F_x = 1.5625 \times 10^{-5}$. Figure 2 compares the final streamwise velocity profile with the analytical Poiseuille solution. The numerical and analytical profiles overlap on the plotted scale and confirm the expected parabolic shape. Table 3 quantifies the remaining small profile error and conservation diagnostics. The relative L_2 error E_{L_2} from Equation (18) and the maximum profile error E_∞ quantify the streamwise profile. Mass error E_m, wall slip E_{slip}, and leakage E_{leak} from Equation (24) measure conservation and wall artifacts. The final centerline and peak speeds are both 0.01996, close to the target $u_{\max} = 0.02$.

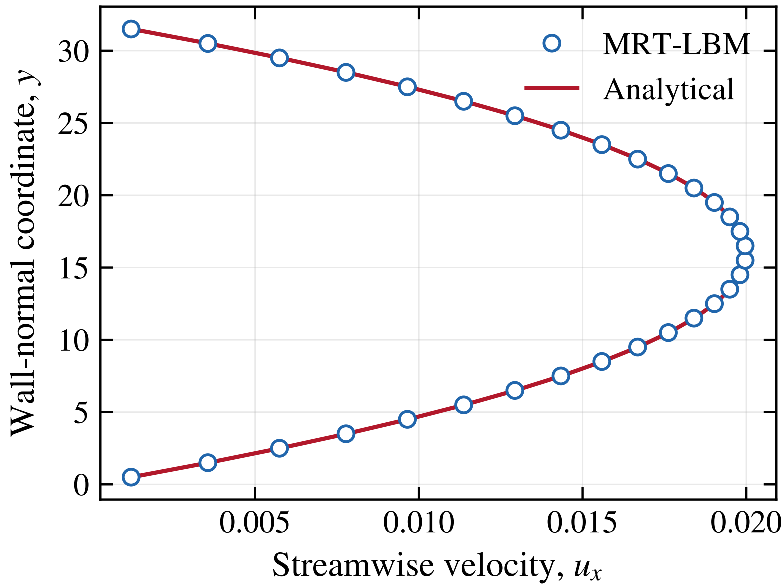

Figure 2. Baseline streamwise velocity profile compared with the analytical Poiseuille solution.

Table 3. Baseline validation metrics for case A1 (rest start) at $64 \times 32 \times 16$, $\tau = 0.8$, and $u_{\max} = 0.02$.

Metric	**Value**
E_{L_2} (Equation (18))	1.15×10^{-3}
E_∞	1.68×10^{-5}
E_m	2.18×10^{-12}
E_{slip}	4.18×10^{-5}
E_{leak} (Equation (24))	1.11×10^{-16}
Final centerline u_x	0.01996
Final maximum u_x	0.01996

4.2. Grid Convergence

The grid-refinement study keeps $\tau = 0.8$ and $u_{\max} = 0.02$ while varying the wall-normal resolution $N_y = 16, 32, 48, 64$. The streamwise and spanwise sizes scale proportionally as $N_x = 2N_y$ and $N_z = N_y/2$ (Table 1). In each case the channel height is

$H = N_y$ and the body force follows Equation (16), so the analytical Poiseuille solution keeps the same peak speed $u_{\max}$ but spans a different wall-normal extent in lattice units. All four grid-refinement cases initialize from the analytical profile to shorten the transient. Figure 3 shows the final x–z mean numerical streamwise velocity $u_x^{num}(y_j)$ at the fluid nodes $y_j = j + 1/2$ for the four wall-normal resolutions. Because H changes with N_y, the four curves occupy different y-intervals rather than a single shared axis. They share approximately the same centerline speed ($u_x \approx u_{\max}$ at $y \approx H/2$) and become smoother parabolic profiles as N_y increases. The coarsest case ($N_y = 16$) has the largest profile error in Table 4, while the refined cases show progressively smaller errors.

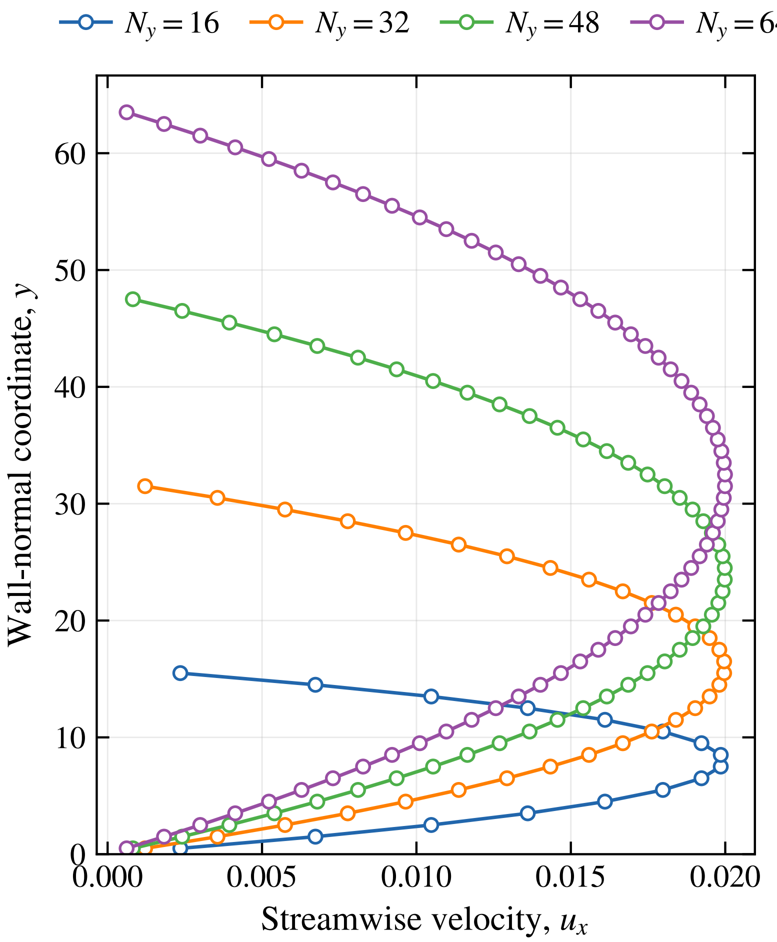

Figure 3. Final numerical streamwise velocity profiles for the four wall-normal resolutions N_y = 16, 32, 48, 64 with $H = N_y$ and fixed $u_{\max} = 0.02$. The profiles use the absolute lattice coordinate y at fluid nodes $y_j = j + 1/2$.

Table 4. Grid-refinement results for cases B1–B4 (parabolic start) with fixed $\tau = 0.8$ and $u_{\max} = 0.02$. Grids use $N_x = 2N_y$ and $N_z = N_y/2$.

Case	Grid (N_y)	F_x	E_{L_2}	E_∞	E_{slip}
B1	$32 \times 16 \times 8$ (16)	6.2500×10^{-5}	4.6×10^{-3}	6.72×10^{-5}	1.67×10^{-4}
B2	$64 \times 32 \times 16$ (32)	1.5625×10^{-5}	1.15×10^{-3}	1.68×10^{-5}	4.18×10^{-5}
B3	$96 \times 48 \times 24$ (48)	6.9444×10^{-6}	5.11×10^{-4}	7.47×10^{-6}	1.86×10^{-5}
B4	$128 \times 64 \times 32$ (64)	3.90625×10^{-6}	2.86×10^{-4}	4.2×10^{-6}	1.04×10^{-5}

Figure 4 reports the quantitative trend. Table 4 lists cases B1–B4. The F_x column shows the body force from Equation (16) at each $H = N_y$, so F_x drops by roughly a factor of four when the channel doubles in height while $u_{\max}$ stays fixed. For each refinement level, E_{L_2} from Equation (18) compares the final numerical profile with the analytical solution on that grid's fluid nodes. All three error metrics decrease as the grid is refined. On log–log axes, E_{L_2} versus N_y is close to a power law,

$$E_{L_2} \approx C N_y^{-p}. \tag{25}$$

Figure 4 adds a grey dashed reference for ideal second-order convergence on the log–log plot. That line follows $E_{L_2} = C_{\mathrm{ref}} N_y^{-2}$ with C_{ref} set by the measured point at the coarsest grid ($N_y = 16$, case B1). The simulation values appear as blue symbols, plotted separately from the reference line so both remain visible when the data lie close to it. A point on that reference line would imply a factor-of-four error reduction each time N_y doubles. The measured points track the reference closely, consistent with $p \approx 2$. A least-squares fit of $\log E_{L_2}$ against $\log N_y$ has slope approximately -2.00, which corresponds to $p = 2.00$ in Equation (25). This second-order wall-normal trend matters for benchmarking because it separates the expected classical discretization error from extra error that may appear in future quantum or hybrid implementations.

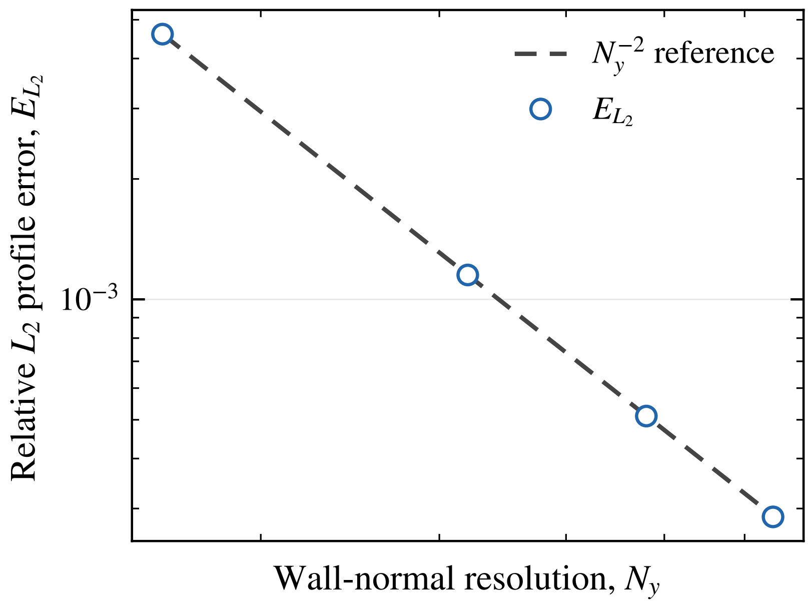

Figure 4. Grid convergence of the relative L_2 velocity-profile error E_{L_2} from Equation (18) versus wall-normal resolution N_y for the four refinement cases in Table 4. Blue circles are the simulation data. The grey dashed line is an N_y^{-2} reference anchored at $N_y = 16$.

Figure 5 summarizes E_{slip}, E_m, and E_{leak} from Equation (24) for all eighteen verification cases in Table 1. The abscissa follows campaign order: baseline A1, grid-refinement cases B1–B4 (labeled by N_y), relaxation-time sweep C1–C6, forcing sweep D1–D5, and initialization cases E1–E2. Lines connect markers to guide the eye across these separate computations. Each tick marks one verification case rather than a point along a continuous parameter sweep. The wall-slip metric E_{slip} (blue) is the largest of the three and reflects the halfway bounce-back treatment. It falls from about 1.7×10^{-4} at $N_y = 16$ to about 10^{-5} on the finer grid and baseline-resolution cases, in line with the wall-normal convergence of the streamwise profile error. Across the fixed-grid τ and $u_{\max}$ sweeps it stays near 10^{-5}. Mass error E_m (green) stays near 10^{-12} and wall-normal leakage E_{leak} (red) near 10^{-16}, so profile and slip error dominate this campaign. Relative to the target velocity scale, these levels are small for a laminar verification benchmark. For the grid-refinement cases with $u_{\max} = 0.02$, the largest wall-slip value, 1.67×10^{-4} at $N_y = 16$, is about 0.8% of $u_{\max}$, while the baseline value 4.18×10^{-5} is about 0.2% of $u_{\max}$. These values are below the normalized profile error on the coarsest grid and comparable to, or smaller than, typical sub-percent verification tolerances used for simple laminar benchmark flows. The mass error is near round-off level and the wall-normal leakage is effectively zero on the plotted scale. Therefore, the dominant numerical uncertainty in this benchmark is the resolved velocity-profile and wall-slip error, rather than conservation drift or spurious cross-channel flow.

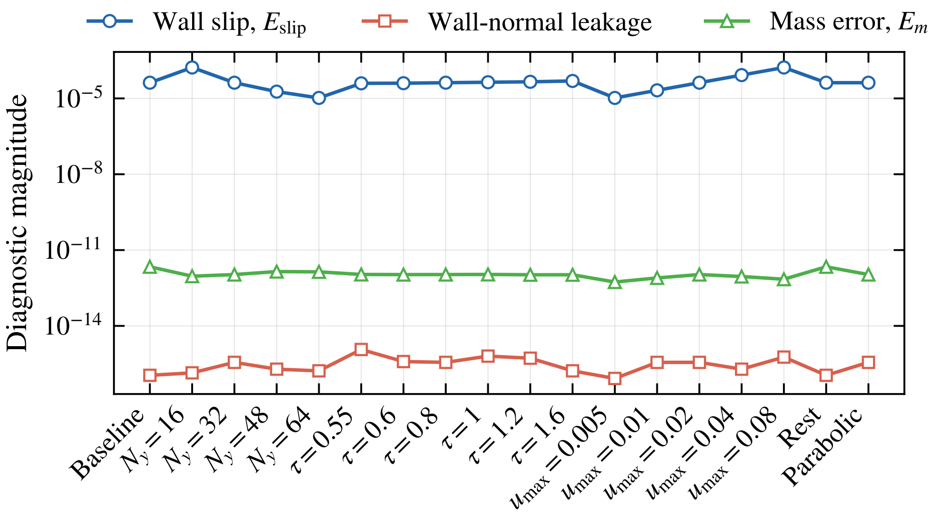

Figure 5. Wall-slip E_{slip}, wall-normal leakage E_{leak} from Equation (24), and mass error E_m for the eighteen-case verification campaign (Table 1). The common case ordering and log-scale vertical axis provide a compact comparison of all three diagnostics across the same simulations.

4.3. Relaxation-Time Sensitivity

The relaxation-time sweep uses cases C1–C6 in Table 1. The grid stays at $64 \times 32 \times 16$, $u_{\max} = 0.02$, and the flow starts from a parabolic profile. The collision time ranges from $\tau = 0.55$ to 1.6. Table 5 lists the paired values of $\nu = (\tau - \frac{1}{2})/3$ from Equation (9) and F_x from Equation (16) at fixed $H = 32$. Because $F_x \propto \nu$ for fixed $u_{\max}$ and H, both ν and F_x increase down the table. Figure 6 plots E_{L_2} from Equation (18) against τ on semilog axes. The error stays near 10^{-3} for $\tau \leq 0.8$ and then falls to about 6.5×10^{-4} at $\tau = 1.6$. Between $\tau = 0.55$ and 0.60 there is a small non-monotonic increase at the lowest viscosities. Table 5 reports E_m near 10^{-12} for every row. Wall-slip values for the C cases appear in Figure 5 and vary only weakly near 4×10^{-5} across τ.

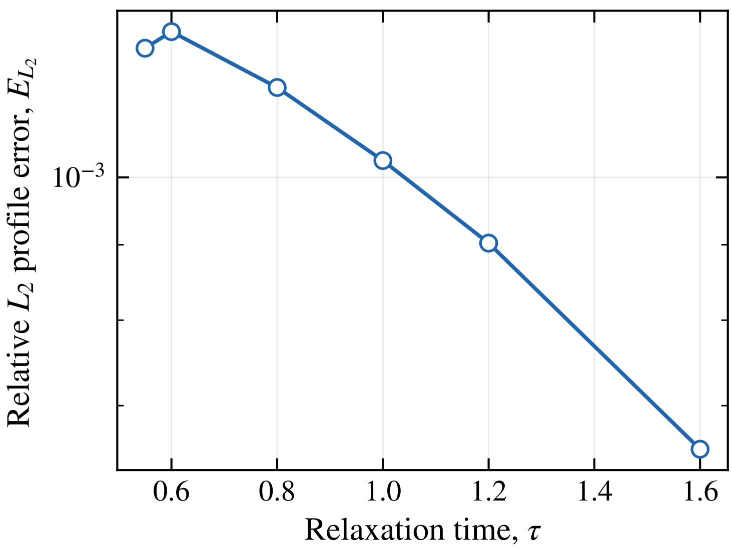

Figure 6. Relaxation-time sensitivity of the relative L_2 profile error E_{L_2} from Equation (18) on the baseline grid $64 \times 32 \times 16$ with $u_{\max} = 0.02$ and a parabolic start. Each point uses F_x from Equation (16) at the corresponding ν.

https://doi.org/10.3390/fluids11070175

Table 5. Relaxation-time sensitivity for cases C1–C6 (parabolic start) on the baseline grid $64 \times 32 \times 16$ with fixed $u_{\max} = 0.02$. The column ν follows Equation (9) and F_x follows Equation (16).

Case	τ	ν	F_x	E_{L_2}	E_m
C1	0.55	0.016667	2.6042×10^{-6}	1.22×10^{-3}	1.08×10^{-12}
C2	0.60	0.033333	5.2083×10^{-6}	1.26×10^{-3}	1.07×10^{-12}
C3	0.80	0.100000	1.5625×10^{-5}	1.15×10^{-3}	1.07×10^{-12}
C4	1.00	0.166667	2.6042×10^{-5}	1.03×10^{-3}	1.09×10^{-12}
C5	1.20	0.233333	3.6458×10^{-5}	9.02×10^{-4}	1.05×10^{-12}
C6	1.60	0.366667	5.7292×10^{-5}	6.54×10^{-4}	1.05×10^{-12}

4.4. Forcing Sensitivity

The forcing sweep uses cases D1–D5 in Table 1. The grid stays at $64 \times 32 \times 16$ with $\tau = 0.8$ (so $\nu = 0.1$) and a parabolic start. The target maximum velocity ranges from $u_{\max} = 0.005$ to 0.080. Table 6 gives the matching F_x from Equation (16) at fixed $H = 32$. Because $F_x \propto u_{\max}$ at fixed ν and H, each doubling of $u_{\max}$ doubles F_x in the table. This sweep checks whether the second-order equilibrium and body-force forcing remain well behaved as the flow speed increases within a low-Mach band. Figure 7 plots E_{L_2} from Equation (18) against $u_{\max}$ on semilog axes (logarithmic $u_{\max}$, linear E_{L_2}) so the nearly constant error level is visible. The relative error stays near 1.15×10^{-3} for every row in Table 6, which points to a discretization-dominated normalized profile error at these forcing levels. The pointwise error E_∞ grows in proportion to $u_{\max}$, as expected when the velocity scale increases. Wall slip and mass conservation for the D cases appear in Figure 5.

Table 6. Forcing-strength sensitivity for cases D1–D5 (parabolic start) on the baseline grid $64 \times 32 \times 16$ with fixed $\tau = 0.8$. The column F_x follows Equation (16).

Case	$u_{\max}$	F_x	E_{L_2}	E_∞
D1	0.005	3.90625×10^{-6}	1.15×10^{-3}	4.2×10^{-6}
D2	0.010	7.8125×10^{-6}	1.15×10^{-3}	8.4×10^{-6}
D3	0.020	1.5625×10^{-5}	1.15×10^{-3}	1.68×10^{-5}
D4	0.040	3.1250×10^{-5}	1.15×10^{-3}	3.36×10^{-5}
D5	0.080	6.2500×10^{-5}	1.15×10^{-3}	6.72×10^{-5}

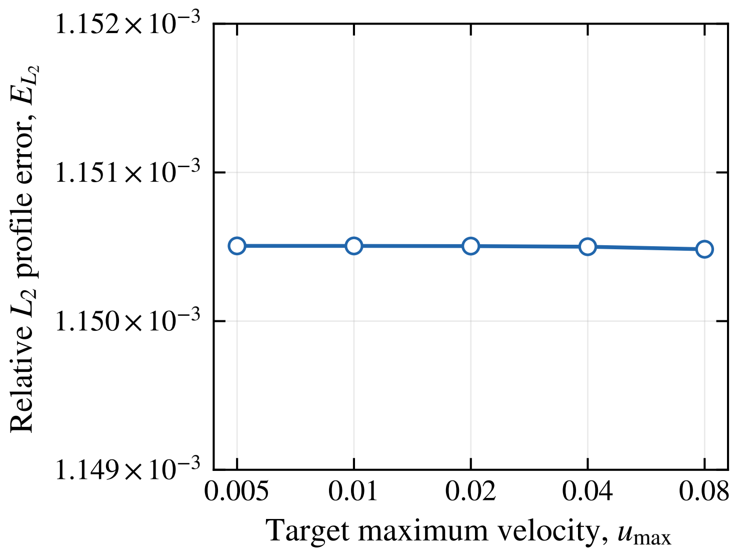

Figure 7. Sensitivity of the relative L_2 profile error E_{L_2} from Equation (18) to the target maximum velocity $u_{\max}$ on the baseline grid ($\tau = 0.8$, parabolic start). Each point uses F_x from Equation (16). The horizontal axis is logarithmic in $u_{\max}$; the vertical axis is linear in E_{L_2}.

4.5. Initial-Condition Effect

The initialization study uses cases E1 and E2 in Table 1. Both simulations use the baseline grid $64 \times 32 \times 16$ with $\tau = 0.8$, $u_{\max} = 0.02$, and F_x from Equation (16). Case E1 starts from rest and is time-marched for 20,000 timesteps. Case E2 starts from the analytical parabolic profile and is integrated for 10,000 timesteps because the flow is already near the forced steady state. Figure 8 tracks the streamwise centerline velocity u_x at the mid-channel fluid row ($y_{N_y/2}$), averaged over x and z, at every 200 timesteps. The rest initialization rises from zero and reaches about 0.020 by roughly n = 10,000, then stays flat. The parabolic initialization begins near that same level and remains nearly constant over the shorter time march, indicating that the forced MRT scheme with halfway bounce-back walls preserves a correctly initialized Poiseuille state. Table 7 summarizes the final metrics. Both cases reach the same E_{L_2} and the same centerline speed 0.01996, close to the target $u_{\max} = 0.02$. Case E1 matches the baseline rest simulation A1 at these reported precisions. Figure 8 highlights the transient path, while the final steady-state accuracy is the same for both initializations.

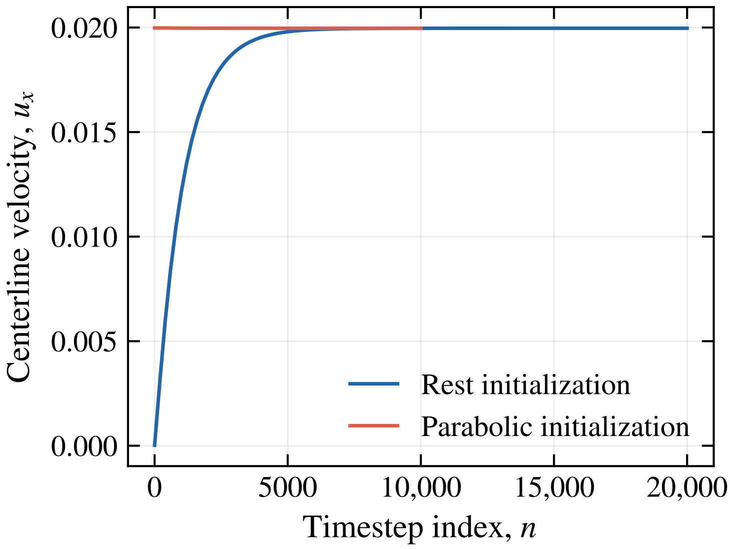

Figure 8. Centerline streamwise velocity u_x at the mid-channel row versus timestep index n for rest (E1, 20,000 steps) and parabolic (E2, 10,000 steps) initialization on the baseline grid.

Table 7. Initial-condition comparison for cases E1–E2 on the baseline grid $64 \times 32 \times 16$ with $\tau = 0.8$ and $u_{\max} = 0.02$. The column E_{L_2} follows Equation (18).

Case	Initialization	Timesteps	E_{L_2}	Final Centerline u_x
E1	Rest	20,000	1.15×10^{-3}	0.01996
E2	Parabolic	10,000	1.15×10^{-3}	0.01996

5. Quantum-Algorithmic Audit and Resource Implications

This section interprets the verified forced MRT-LBM timestep as a sequence of operations relevant to future quantum or hybrid quantum-classical solvers. The aim is to clarify the algorithmic structure of the benchmark and to identify the parts that are naturally reversible, linear, dissipative, nonlinear, or measurement-intensive. This distinction is important because equilibrium evaluation, dissipative relaxation, and measurement are central challenges for quantum flow simulation [6,26].

5.1. Operator-Level Decomposition and Register Maps

For the streaming step, a position and velocity basis state may be written as

$$|x, y, z, i\rangle,$$

where $i \in \{0, \ldots, 18\}$ indexes the D3Q19 population. The ideal streaming map has the reversible form

$$|x, y, z, i\rangle \mapsto |x + c_{ix}, y + c_{iy}, z + c_{iz}, i\rangle,$$

with periodic wrapping in the streamwise and spanwise directions. At a wall-crossing link, halfway bounce-back can be represented at the operator level as a boundary-controlled velocity reflection,

$$|x, y, z, i\rangle \mapsto |x, y, z, \bar{i}\rangle,$$

where $\bar{i}$ is the opposite D3Q19 direction. These mappings show why streaming and bounce-back are the most directly quantum-compatible parts of the timestep.

Table 8 summarizes the quantum status of each operator in the present forced D3Q19 MRT benchmark. The audit is operator-level and literature-supported: it maps each timestep primitive for forced D3Q19 MRT with body-force forcing and halfway bounce-back walls to quantum implementation requirements. To address implementation feasibility without claiming measured quantum performance, the final column gives indicative logical qubit counts, circuit-depth scaling, and gate-count scales for the main timestep operators before hardware transpilation, error correction, and device-specific routing.

Table 8. Quantum-operator audit and indicative logical circuit-complexity estimate for the forced D3Q19 MRT-LBM timestep. The final column reports qubit, depth, and gate-scale estimates before hardware transpilation, error correction, and device-specific routing.

Operator in This Benchmark	Quantum Interpretation	Literature-Supported Assessment	Indicative Qubit/Depth/Gate Metric
Position–velocity encoding	Encode lattice position and the D3Q19 population index as quantum registers.	Feasible lower-bound register model. D3Q19 velocity index: five qubits ($\lceil \log_2 19 \rceil = 5$; Equation (26)). Excludes finite-precision, work, boundary-flag, block-encoding, loading, and measurement registers.	$q_{\min} = q_x + q_y + q_z + 5$. Baseline grid: $q_x = 6$, $q_y = 5$, $q_z = 4$, hence $q_{\min} = 20$ logical index qubits. This excludes precision, work, loading, collision, and measurement registers.
Streaming	Reversible lattice permutation conditioned on the velocity index.	Most quantum-compatible LBM timestep stage. QLBM work commonly treats streaming as a shift or permutation; collision and readout need separate handling. Dynamic-circuit QLBM supports this mainly for linear advection–diffusion LBM, not full forced nonlinear D3Q19 MRT with body-force forcing [7,8].	Reversible controlled permutation. Depth/gate scale $O(q_x + q_y + q_z)$ per streamed update. Baseline: $q_x + q_y + q_z = 15$, so the transport part requires controlled-increment/decrement subcircuits of order 10^1–10^2 elementary logical operations, depending on the chosen adder and control decomposition, before routing and error correction.
Bounce-back walls	Boundary-controlled map $i \to \bar{i}$ on the velocity register.	Controlled opposite-link map is plausible. Wall detection, valid-link logic, and boundary flags add overhead. QLBM and quantum Carleman lattice Boltzmann method (QCLBM) boundary studies include bounce-back-type treatments mainly in simplified, low-dimensional, linearized, low-*Re*, or Carleman-lifted settings, not yet in full forced D3Q19 MRT channels [9,27,28].	y-register wall comparison plus fixed five-qubit opposite-link permutation. Depth/gate scale $O(q_y) + O(1)$. Baseline: order 10^1 controlled/Toffoli-class operations plus a small number of flag/work qubits.

Table 8. *Cont.*

Operator in This Benchmark	Quantum Interpretation	Literature-Supported Assessment	Indicative Qubit/Depth/Gate Metric
Moment transforms (M, M^{-1})	Local linear maps between population and moment space.	Dense local maps are generally nonunitary. Require unitary embedding, block encoding, linear-combination-of-unitaries (LCU)/singular-value decomposition (SVD), or hybrid classical treatment [7,28,29]. QLBM studies use matrix decompositions for linearized or low-*Re* cases; quantum singular value transformation (QSVT) and block encoding provide the broader framework.	Local 19×19 maps embedded in the 2^5 velocity register. A generic dense five-qubit unitary embedding has synthesis scale $O(4^5) \approx 10^3$ at the audit level; the actual cost depends on the block-encoding construction, normalization, ancillas, postselection or amplification, and compiler choices [29,30].
MRT relaxation	Dissipative collision in moment space.	Implementation-critical nonunitary operator: damps nonconserved moments; not a reversible lattice permutation. No direct unitary update without embedding, channel treatment, postselection, or hybrid processing. Signed two-rail completely positive trace-preserving (CPTP) construction realizes the diagonal dissipative MRT relaxation block [31]; the remaining timestep operators require separate treatment.	Diagonal but nonunitary 19-mode map. No unique fixed gate count without choosing a dilation, block-encoding, CPTP-channel, or hybrid realization. Resource scale is dominated by ancillas, rail/block encoding, precision, and amplitude-management overhead.
Equilibrium and body-force forcing	Local nonlinear/affine update depending on ρ, $\mathbf{u}$, and $\mathbf{F}$.	Among the hardest operators: equilibrium is nonlinear in recovered $\mathbf{u}$; forcing in Equation (10) couples the body force $\mathbf{F}$ to the local macroscopic state through Φ_i. Nonlinear QLBM, Carleman LBM, and related schemes advance the field but do not yet implement this forced D3Q19 MRT benchmark [6,9,11].	Not a fixed 19×19 unitary. Requires reversible or hybrid evaluation of ρ, $\mathbf{u}$, quadratic equilibrium terms, and forcing. For p-bit arithmetic, depth is governed by fixed-point multiplications/divisions, typically polynomial in p, in addition to loading and uncomputation overhead.
Macroscopic recovery and measurement	Estimate ρ, $\mathbf{u}$, and diagnostic observables from the quantum state.	Measurement-intensive: repeated full-field readout and reinitialization reduce quantum encoding benefits. Dynamic-circuit QLBM lowers overhead for linear advection–diffusion LBM [7,8]; extension to forced nonlinear MRT-LBM remains open.	Sampling-limited full-profile recovery over many observables. Direct sampling scales as $O(1/\epsilon^2)$ shots per estimated observable for accuracy ϵ. Amplitude estimation can improve sampling scaling but increases coherent circuit depth.

5.2. Minimum Logical-Qubit Estimate

A minimal position-velocity encoding requires qubits for x, y, z, and the D3Q19 population index. A lower bound is

$$q_{min} = \lceil \log_2 N_x \rceil + \lceil \log_2 N_y \rceil + \lceil \log_2 N_z \rceil + \lceil \log_2 19 \rceil. \tag{26}$$

Since $\lceil \log_2 19 \rceil = 5$, each spatial direction uses $q_x = \lceil \log_2 N_x \rceil$, $q_y = \lceil \log_2 N_y \rceil$, and $q_z = \lceil \log_2 N_z \rceil$. Table 9 lists q_{min} for the channel grids used in the verification study and one finer illustrative grid. The first four rows match cases B1–B4 ($N_x = 2N_y$, $N_z = N_y/2$). The row $256 \times 128 \times 64$ extends the B-grid sequence one refinement level beyond B4 and lies outside the eighteen-case campaign. The entries $96 \times 48 \times 24$ and $128 \times 64 \times 32$ share $q_{min} = 23$ because $\lceil \log_2 N_x \rceil$ and $\lceil \log_2 N_z \rceil$ reach the same ceilings even though the physical sizes differ. These counts are lower bounds for the index registers in the basis state of Equation (27). Amplitude registers for all nineteen populations at every node, finite-precision arithmetic, wall flags, block-encoding ancillas, and measurement overhead lie outside this estimate. Section 5.4 gives a practical streaming–bounce-back estimate q_{SB} on top of q_{min}.

Table 9. Lower-bound logical-qubit estimates for position–velocity encoding of D3Q19 channel grids from Equation (26). Rows 1–4 are the B-grid refinement levels in Table 4. Row 5 is an illustrative finer grid.

Grid	q_x	q_y	q_z	Velocity Qubits	q_{min}
$32 \times 16 \times 8$ (B1)	5	4	3	5	17
$64 \times 32 \times 16$ (B2)	6	5	4	5	20
$96 \times 48 \times 24$ (B3)	7	6	5	5	23
$128 \times 64 \times 32$ (B4)	7	6	5	5	23
$256 \times 128 \times 64$	8	7	6	5	26

5.3. Implications for Future Quantum LBM

The audit suggests that a practical quantum implementation should begin with the parts that have clear reversible structure. Streaming and bounce-back can be treated as permutations on position and velocity registers. Local moment transforms are also suitable targets for block-encoding or small-register linear maps. The main difficulty lies in the collision and forcing stage. MRT relaxation is dissipative, while the equilibrium and forcing source term depend on ρ and $\mathbf{u}$. Recent work shows that the diagonal dissipative MRT relaxation block can be realized deterministically using a signed two-rail encoding and completely positive trace-preserving (CPTP) amplitude-damping channels [31]; however, that result targets the relaxation subblock and does not remove the separate costs of constructing $\mathbf{m}^{eq}$, applying M and M^{-1}, enforcing boundaries, or measuring macroscopic fields. Hybrid schemes may therefore compute selected macroscopic fields classically while preserving quantum treatment for linear transport, or may use lifting and linearization ideas such as Carleman embedding [9,11]. The present benchmark gives a compact wall-bounded test case for comparing such choices.

5.4. Explicit Reversible Kernel and Block-Encoding Pathway

To make the operator audit concrete, this subsection specifies an operator-level quantum target aligned with the classical D3Q19 MRT channel solver. The scope is the transport and wall-treatment part of the timestep, together with a standard block-encoding route for the local moment transforms M and M^{-1}. It is an audit-level circuit specification for transport, wall treatment, and local moment embedding. That focus matches recent quantum-LBM work, in which streaming is the most naturally unitary component, while collision, forcing, equilibrium evaluation, and measurement remain the more difficult parts and are commonly treated in related studies using strategies such as Hamiltonian simulation, Carleman lifting, dynamic circuits, linearization, singular value decomposition (SVD), block encodings, or hybrid classical–quantum processing [6–8,18,19,28].

Let the computational basis state be

$$|\psi\rangle = |x\rangle|y\rangle|z\rangle|i\rangle, \tag{27}$$

where $x \in \{0, \ldots, N_x - 1\}$, $y \in \{0, \ldots, N_y - 1\}$, $z \in \{0, \ldots, N_z - 1\}$, and $i \in \{0, \ldots, 18\}$ is the D3Q19 velocity index. The velocity register uses five qubits; basis states $19, \ldots, 31$ are unused in the present encoding and may be flagged as invalid. Based on the halfway bounce-back rule used in the classical channel solver, we define the following operator-level wall-crossing flag and reversible streaming–bounce-back permutation in the $|x, y, z, i\rangle$ basis (not a standard equation from the quantum-LBM literature). The flag applies at the bottom and top fluid rows,

$$b(x, y, z, i) = \begin{cases} 1, & y = 0 \text{ and } c_{iy} = -1, \\ 1, & y = N_y - 1 \text{ and } c_{iy} = +1, \\ 0, & \text{otherwise}, \end{cases} \tag{28}$$

where c_{iy} is the y component of the D3Q19 velocity $\mathbf{c}_i$. The reversible streaming–bounce-back kernel is the permutation

$$U_{\text{SB}} |x, y, z, i\rangle = \begin{cases} |x + c_{ix} \pmod{N_x},\ y + c_{iy},\ z + c_{iz} \pmod{N_z},\ i\rangle, & b = 0, \\ |x, y, z, \bar{i}\rangle, & b = 1, \end{cases} \tag{29}$$

where $\bar{i}$ is the opposite D3Q19 direction. Ordinary streaming is therefore a velocity-controlled modular shift in x and z and a shift in y when $b = 0$; halfway bounce-back is a boundary-controlled permutation of the velocity register when $b = 1$. Both branches are one-to-one on the computational basis, so U_{SB} is unitary. Algorithm 1 summarizes the reversible streaming–bounce-back kernel used for the D3Q19 channel flow.

Algorithm 1 Reversible streaming–bounce-back kernel for D3Q19 channel flow

Step	**Operation**
1	*Input:* initialize registers X, Y, Z, and V; set all scratch flags to zero.
2	*Decode:* from the velocity register V, set the direction flags $d_x^\pm$, $d_y^\pm$, and $d_z^\pm$.
3	*Wall flag:* compute $b = \big[(Y = 0) \wedge d_y^-\big] \vee \big[(Y = N_y - 1) \wedge d_y^+\big].$
4	*Streaming branch:* if $b = 0$, apply controlled ± 1 shifts on X, Y, and Z, with periodic wrapping in x and z.
5	*Bounce-back branch:* if $b = 1$, apply the opposite-link permutation Π_{opp} on V: $\Pi_{\text{opp}} |i\rangle = |\bar{i}\rangle, \quad i = 0, \ldots, 18.$ The map uses the classical D3Q19 opposite-link table.
6	*Uncompute:* clear all scratch flags.

For the position–velocity encoding, Equation (26) gives $q_{\min}$. The reversible kernel requires a modest number of additional scratch qubits. A practical register estimate is

$$q_{\text{SB}} = q_{\min} + q_{\text{flag}} + q_{\text{work}}, \tag{30}$$

where $q_{\text{flag}} \leq 7$ stores the six direction flags and the wall flag, and q_{work} denotes one or a few clean work qubits for controlled increment and decrement operations. For the baseline grid $64 \times 32 \times 16$, $q_{\min} = 20$, which yields $q_{\text{SB}} \approx 28$–30 logical qubits before precision registers, amplitude loading, collision ancillas, block-encoding ancillas, or measurement overhead. The asymptotic gate scaling of this kernel is

$$G_{\text{SB}} = O(q_x + q_y + q_z) + O(1), \tag{31}$$

because D3Q19 has a fixed number of velocity links and the velocity lookup and opposite-direction map have cost independent of the lattice size when the state is encoded in superposition. For the baseline grid, the dominant cost is controlled arithmetic on registers of sizes $q_x = 6$, $q_y = 5$, and $q_z = 4$.

At each lattice node, the MRT algorithm applies the dense local maps

$$\mathbf{m} = M\mathbf{f}, \qquad \mathbf{f} = M^{-1}\mathbf{m}. \tag{32}$$

Because M and M^{-1} are invertible linear maps with $\|A\|_2$ generally greater than one, they are embedded through block encoding [29]. For each local matrix $A \in \{M, M^{-1}\}$, choose a scale $\alpha_A \geq \|A\|_2$ and construct a unitary U_A such that

$$(\langle 0^a| \otimes I)\, U_A\, (|0^a\rangle \otimes I) = \frac{A}{\alpha_A}. \tag{33}$$

The corresponding lattice-wide operator is

$$\mathcal{U}_A = \sum_{x,y,z} |x,y,z\rangle\langle x,y,z| \otimes U_A, \tag{34}$$

so the same local block encoding is reused at every lattice node. Because the D3Q19 velocity space is fixed, A is a 19×19 matrix embedded in the 2^5-dimensional Hilbert space of the velocity register. The local synthesis cost is therefore independent of $N_x N_y N_z$, although the embedding is not free: a generic dense five-qubit unitary has $O(4^5)$ synthesis scale, with additional ancillas, normalization, postselection, possible amplitude amplification, and compilation overhead for a block-encoded implementation [7,28–30]. Exploiting the algebraic structure of the D3Q19 MRT matrix may reduce this cost relative to a generic dense synthesis.

This decomposition identifies two explicit quantum targets for the benchmark. First, U_{SB} in Equation (29) is an exact reversible permutation matched to the classical streaming–bounce-back step. Second, Equations (33) and (34) provide a standard embedding route for the local moment transforms. MRT relaxation, equilibrium evaluation, body-force forcing, macroscopic recovery, and measurement are documented separately in Table 8 as the demanding stages of a full quantum translation. The benchmark therefore supplies a verified classical reference problem together with explicit operator-level targets for future QPU-assisted or hybrid quantum-classical implementations [6,8,18,28].

6. Reproducibility

All simulations were performed with an in-house D3Q19 MRT channel solver with body-force forcing. The code is available at https://github.com/IdreesKhan3/mrt-lbm-quantum-audit, (accessed on 4 July 2026). The repository contains five main scripts: `1_mrt_guo_channel.py` implements the D3Q19 MRT channel solver, `2_run_campaign.py` defines and runs the eighteen verification cases, `3_collect_results.py` collects the case summaries, `4_make_figures.py` regenerates the paper figures, and `5_quantum_audit.py` regenerates the quantum-audit tables.

The verification cases are defined in `2_run_campaign.py` and consist of A1, B1–B4, C1–C6, D1–D5, and E1–E2. Running the campaign writes the case input, summary metrics, and profile/time-history data for each case. Appendix A specifies the D3Q19 link ordering, moment basis, full moment matrix M, and relaxation vector used by the solver.

7. Limitations

The study focuses on laminar, primarily low-Mach-number, body-force-driven channel flow with a single D3Q19 MRT formulation and the body-force forcing treatment in Section 2.4. The verification campaign spans $Re_{\max} = 1.6$–38.4 and $Ma_{\max} = 0.0087$–0.139, so turbulent or high-Reynolds-number regimes are outside the present scope. This scope isolates ingredients central to quantum flow-simulation benchmarks: streaming, wall reflection, MRT collision, body forcing, equilibrium evaluation, macroscopic recovery, and measurement. Future work will extend the benchmark set to more complex wall geometries, unsteady flows, three-dimensional vortical configurations, explicit quantum-circuit constructions for selected sub-operators, and quantum hardware execution benchmarks relative to the classical reference. Demonstrating runtime or resource superiority on a QPU is not attempted in the present study.

8. Conclusions

The eighteen-case verification campaign confirms second-order wall-normal convergence ($p \approx 2$ from Figure 4) and stable mass conservation for the forced D3Q19 MRT channel solver on the reported parameter sweeps. The operator audit ranks streaming and halfway bounce-back as the most directly reversible timestep parts, while MRT relaxation, equilibrium evaluation, body forcing, and measurement carry the larger quantum-implementation burden. Section 5.4 turns that audit into concrete targets: the reversible permutation U_{SB} in Equation (29) and block-encoded local moment maps.

The benchmark is intended for methodological operator analysis and reproducibility, not for performance comparison. It provides a verified classical reference problem, transparent error diagnostics, and an operator-by-operator map of quantum-implementation requirements for the same forced D3Q19 MRT timestep. The study does not claim quantum speed-up, QPU runtime advantage, or resource superiority over classical execution; those questions require future hardware implementations and lie outside the present scope. Scripts at the repository linked in Section 6 regenerate the summary tables, figures, and audit tables so the reference data and operator decomposition can be reused in subsequent quantum and hybrid studies.

Author Contributions: M.I.K.: methodology, code, validation, analysis, figures, and draft writing. H.-D.Y.: supervision. Both authors: conceptualization, review, editing, and approval. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Vinnova–Fordonsstrategisk forskning och innovation (FFI) in the project "3D Virtual Platform for Digitalization of Holistic Acoustic Environment in Cabs of Heavy-Duty Vehicles (OCTAVE)" under Grant No. P2024-01011.

Data Availability Statement: The solver, campaign scripts, figure-generation scripts, and quantum-audit scripts are available at https://github.com/IdreesKhan3/mrt-lbm-quantum-audit, (accessed on 4 July 2026). The eighteen verification datasets are defined in `2_run_campaign.py` and are regenerated by running the campaign script. Each run produces the case input, machine-readable summary, and profile/time-history output files.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. MRT Basis Specification

This appendix records the D3Q19 lattice link ordering, moment basis, moment matrix, and relaxation rates used by the reference D3Q19 MRT channel solver described in Section 6.

Appendix A.1. D3Q19 Population Ordering

Table A1 lists the velocity vectors and weights for each population index i. Columns of the moment matrix M in Equation (A1) follow this population order.

Table A1. D3Q19 lattice links and weights used in the reference solver (population index i).

i	$\mathbf{c}_i = (c_{ix}, c_{iy}, c_{iz})$	w_i
0	$(1, 0, 0)$	$1/18$
1	$(-1, 0, 0)$	$1/18$
2	$(0, 1, 0)$	$1/18$
3	$(0, -1, 0)$	$1/18$
4	$(0, 0, 1)$	$1/18$
5	$(0, 0, -1)$	$1/18$
6	$(1, 1, 0)$	$1/36$
7	$(-1, -1, 0)$	$1/36$
8	$(1, -1, 0)$	$1/36$
9	$(-1, 1, 0)$	$1/36$
10	$(0, 1, 1)$	$1/36$
11	$(0, -1, -1)$	$1/36$
12	$(0, 1, -1)$	$1/36$
13	$(0, -1, 1)$	$1/36$
14	$(1, 0, 1)$	$1/36$
15	$(-1, 0, -1)$	$1/36$
16	$(1, 0, -1)$	$1/36$
17	$(-1, 0, 1)$	$1/36$
18	$(0, 0, 0)$	$1/3$

Appendix A.2. Moment Matrix

The moment matrix M maps populations to the moment vector in Equation (7). Its rows are moments and its columns are populations ordered as in Table A1.

$$M = \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
-11 & -11 & -11 & -11 & -11 & -11 & 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 & 8 & -30 \\
-4 & -4 & -4 & -4 & -4 & -4 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 12 \\
1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 0 \\
-4 & 4 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 0 \\
0 & 0 & 1 & -1 & 0 & 0 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -4 & 4 & 0 & 0 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 0 \\
0 & 0 & 0 & 0 & -4 & 4 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 0 \\
2 & 2 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 & 0 \\
-4 & -4 & 2 & 2 & 2 & 2 & 1 & 1 & 1 & 1 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & 0 \\
0 & 0 & -2 & -2 & 2 & 2 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & -1 & 1 & -1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & 0
\end{bmatrix} \tag{A1}$$

Appendix A.3. Relaxation Rates

Table A2 gives the diagonal MRT relaxation rates $s_0, \ldots, s_{18}$ used by the reference solver. For a chosen τ, the shear moments use $s_\nu = 1/(3\nu + 1/2)$ with $\nu = (\tau - 1/2)/3$ from Equation (9); all other nonconserved entries are fixed to the standard values in the table.

Table A2. MRT relaxation rates for the D3Q19 moment basis ($\tau = 0.8$, $\nu = 0.1$, $s_\nu = 1.25$).

Index	Moment	s_i
0	$\delta\rho$	0
1	e	1.19
2	ϵ	1.4
3	j_x	0
4	q_x	1.2
5	j_y	0
6	q_y	1.2
7	j_z	0
8	q_z	1.2
9	$3p_{xx}$	s_ν
10	$3\pi_{xx}$	1.4
11	p_{ww}	s_ν
12	π_{ww}	1.4
13	p_{xy}	s_ν
14	p_{yz}	s_ν
15	p_{xz}	s_ν
16	m_x	1.98
17	m_y	1.98
18	m_z	1.98

References

1. Malinverno, G.; Blasco Alberto, J. A Review of the Current State-of-the-Art of Quantum Computing for CFD: Approaches, Advantages, and Limitations. *Aerotec. Missili Spaz.* **2026**, *105*, 251–268. [CrossRef]
2. Succi, S. *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond*; Oxford University Press: Oxford, UK, 2001.
3. Krüger, T.; Kusumaatmaja, H.; Kuzmin, A.; Shardt, O.; Silva, G.; Viggen, E.M. *The Lattice Boltzmann Method: Principles and Practice*; Springer: Cham, Switzerland, 2017. [CrossRef]
4. Aidun, C.K.; Clausen, J.R. Lattice-Boltzmann method for complex flows. *Annu. Rev. Fluid Mech.* **2010**, *42*, 439–472. [CrossRef]
5. Yepez, J. Quantum lattice-gas model for computational fluid dynamics. *Phys. Rev. E* **2001**, *63*, 046702. [CrossRef] [PubMed]
6. Wang, B.; Meng, Z.; Zhao, Y.; Yang, Y. Quantum lattice Boltzmann method for simulating nonlinear fluid dynamics. *npj Quantum Inf.* **2025**, *11*, 196. [CrossRef]
7. Wawrzyniak, D.; Winter, J.; Schmidt, S.; Indinger, T.; Janßen, C.F.; Schramm, U.; Adams, N.A. A quantum algorithm for the lattice-Boltzmann method advection-diffusion equation. *Comput. Phys. Commun.* **2025**, *306*, 109373. [CrossRef]
8. Wawrzyniak, D.; Winter, J.; Schmidt, S.; Indinger, T.; Janßen, C.F.; Schramm, U.; Adams, N.A. Linearized quantum lattice-Boltzmann method for the advection-diffusion equation using dynamic circuits. *Comput. Phys. Commun.* **2025**, *317*, 109856. [CrossRef]
9. Turro, F.; Lignarolo, A.; Dragoni, D. Practical application of the quantum Carleman lattice Boltzmann method in industrial CFD simulations. *arXiv* **2025**, arXiv:2504.13033. [CrossRef]
10. Harrow, A.W.; Hassidim, A.; Lloyd, S. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.* **2009**, *103*, 150502. [CrossRef] [PubMed]
11. Costa, P.C.S.; Schleich, P.; Morales, M.E.S.; Berry, D.W. Further improving quantum algorithms for nonlinear differential equations via higher-order methods and rescaling. *npj Quantum Inf.* **2025**, *11*, 141. [CrossRef]
12. Khan, M.I.; Succi, S.; Yao, H.D.; Falcucci, G. Physics-Constrained Neural Closure for Lattice Boltzmann Large-Eddy Simulation. *arXiv* **2026**, arXiv:2603.15992. [CrossRef]
13. Fischer, P.; Kaltenbach, S.; Litvinov, S.; Succi, S.; Koumoutsakos, P. Optimal Lattice Boltzmann Closures through Multi-Agent Reinforcement Learning. *arXiv* **2025**, arXiv:2504.14422. [CrossRef]
14. Montessori, A.; La Rocca, M.; Amati, G.; Lauricella, M.; Tiribocchi, A.; Succi, S. High-order thread-safe lattice Boltzmann model for high performance computing turbulent flow simulations. *Phys. Fluids* **2024**, *36*, 035171. [CrossRef]
15. Falcucci, G.; Amati, G.; Fanelli, P.; Krastev, V.K.; Polverino, G.; Porfiri, M.; Succi, S. Extreme flow simulations reveal skeletal adaptations of deep-sea sponges. *Nature* **2021**, *595*, 537–541. [CrossRef] [PubMed]
16. Biferale, L.; Mantovani, F.; Sbragaglia, M.; Scagliarini, A.; Toschi, F.; Tripiccione, R. High resolution numerical study of Rayleigh–Taylor turbulence using a thermal lattice Boltzmann scheme. *Phys. Fluids* **2010**, *22*, 115112. [CrossRef]
17. Khan, M.I.; Succi, S.; Falcucci, G. Validating the Boltzmann Approach to the Large-Eddy Simulations of Forced Homogeneous Incompressible Turbulence. *Int. J. Mod. Phys. C* **2026**, 2750064. [CrossRef]

18. Itani, W.; Sreenivasan, K.R.; Succi, S. Quantum algorithm for lattice Boltzmann (QALB) simulation of incompressible fluids with a nonlinear collision term. *Phys. Fluids* **2024**, *36*, 017112. [CrossRef]
19. Sanavio, C.; Succi, S. Lattice Boltzmann–Carleman quantum algorithm and circuit for fluid flows at moderate Reynolds number. *AVS Quantum Sci.* **2024**, *6*, 023802. [CrossRef]
20. Lallemand, P.; Luo, L.S. Theory of the lattice Boltzmann method: Dispersion, dissipation, isotropy, Galilean invariance, and stability. *Phys. Rev. E* **2000**, *61*, 6546–6562. [CrossRef] [PubMed]
21. d'Humières, D.; Ginzburg, I.; Krafczyk, M.; Lallemand, P.; Luo, L.S. Multiple-relaxation-time lattice Boltzmann models in three dimensions. *Philos. Trans. R. Soc. Lond. Ser. A* **2002**, *360*, 437–451. [CrossRef] [PubMed]
22. Ladd, A.J.C. Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 1. Theoretical foundation. *J. Fluid Mech.* **1994**, *271*, 285–309. [CrossRef]
23. He, X.; Zou, Q.; Luo, L.S.; Dembo, M. Analytic solutions of simple flows and analysis of nonslip boundary conditions for the lattice Boltzmann BGK model. *J. Stat. Phys.* **1997**, *87*, 115–136. [CrossRef]
24. Guo, Z.; Zheng, C.; Shi, B. Discrete lattice effects on the forcing term in the lattice Boltzmann method. *Phys. Rev. E* **2002**, *65*, 046308. [CrossRef] [PubMed]
25. Zou, Q.; He, X. On pressure and velocity boundary conditions for the lattice Boltzmann BGK model. *Phys. Fluids* **1997**, *9*, 1591–1598. [CrossRef]
26. Lewis, D.; Eidenbenz, S.; Nadiga, B.; Subaşı, Y. Limitations for quantum algorithms to solve turbulent and chaotic systems. *Quantum* **2024**, *8*, 1509. [CrossRef]
27. Bakker, B.; Watts, T.W. Quantum Carleman Linearization of the Lattice Boltzmann Equation with Boundary Conditions. *arXiv* **2024**, arXiv:2312.04781. [CrossRef]
28. Kumar, H.; Frankel, S.H. Quantum unitary matrix representation of the lattice Boltzmann model for low Reynolds fluid flow simulation. *AVS Quantum Sci.* **2025**, *7*, 013802. [CrossRef]
29. Gilyén, A.; Su, Y.; Low, G.H.; Wiebe, N. Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, New York, NY, USA, 23–26 June 2019*; STOC 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 193–204. [CrossRef]
30. Shende, V.V.; Bullock, S.S.; Markov, I.L. Synthesis of quantum-logic circuits. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **2006**, *25*, 1000–1010. [CrossRef]
31. Khan, M.I.; Succi, S.; Yao, H.D. Deterministic Realization of Classical Dissipation on Quantum Computers. *arXiv* 2026, arXiv:2604.25429. [CrossRef]

https://doi.org/10.3390/fluids11070175