Semi-Lagrangian lattice Boltzmann method for compressible flows

Dominik Wilde [1,2,*] Andreas Krämer [3] Dirk Reith [2,4] and Holger Foysi[1]

[1]*Department of Mechanical Engineering, University of Siegen, Paul-Bonatz-Straße 9-11, D-57076 Siegen-Weidenau, Germany*

[2]*Institute of Technology, Resource and Energy-efficient Engineering (TREE), Bonn-Rhein-Sieg University of Applied Sciences, Grantham-Allee 20, D-53757 Sankt Augustin, Germany*

[3]*National Heart, Lung, and Blood Institute, National Institutes of Health, Bethesda, Maryland 20892, USA*

[4]*Fraunhofer Institute for Algorithms and Scientific Computing (SCAI), Schloss Birlinghoven, D-53754 Sankt Augustin, Germany*

This work thoroughly investigates a semi-Lagrangian lattice Boltzmann (SLLBM) solver for compressible flows. In contrast to other LBM for compressible flows, the vertices are organized in cells, and interpolation polynomials up to fourth order are used to attain the off-vertex distribution function values. Differing from the recently introduced Particles on Demand (PoD) method [Dorschner, Bösch, and Karlin, Phys. Rev. Lett. **121**, 130602 (2018)], the method operates in a static, nonmoving reference frame. Yet the SLLBM in the present formulation grants supersonic flows and exhibits a high degree of Galilean invariance. The SLLBM solver allows for an independent time step size due to the integration along characteristics and for the use of unusual velocity sets, like the D2Q25, which is constructed by the roots of the fifth-order Hermite polynomial. The properties of the present model are shown in diverse example simulations of a two-dimensional Taylor-Green vortex, a Sod shock tube, a two-dimensional Riemann problem, and a shock-vortex interaction. It is shown that the cell-based interpolation and the use of Gauss-Lobatto-Chebyshev support points allow for spatially high-order solutions and minimize the mass loss caused by the interpolation. Transformed grids in the shock-vortex interaction show the general applicability to nonuniform grids.

I. INTRODUCTION

In the field of weakly compressible and isothermal flows, the lattice Boltzmann method (LBM) [1–3] has emerged as an efficient numerical solver that suits modern, highly parallel computing architectures. Consequently, many attempts have been undertaken to extend the method to energy conserving flows [4]. Despite considerable progress in this field, the research on robust and efficient lattice Boltzmann models for compressible flows is still ongoing.

Existing approaches can generally be categorized in two ways [5]: on- versus off-lattice, and single- versus double-population. While on-lattice approaches inherit the LBM's exact, space-filling streaming step [6–9], off-lattice methods discretize the Boltzmann equation through finite volume [10–13] or finite difference schemes [14–16]. Orthogonal to this classification, single-population models [4,17] express all physical moments, including energy and heat flux, by a single distribution function. In order to represent all moments, the particle velocity sets have to be extended beyond the lattices typically used for weakly compressible flows. In contrast, double-population models [8,18,19] represent the local internal energy through a separate distribution function that is coupled with the density and momentum coming from the first distribution function.

The works of Frapolli [17] and Coreixas [20] present an excellent overview of compressible extensions of the LBM. They show how previous approaches have traditionally suffered from instability, numerical dissipation, large computational cost, and undesirable couplings between the physical and numerical parameters.

The use of an efficient off-lattice method is key, since compressible on-lattice Boltzmann methods suffer from decisive disadvantages. First, the time step size is set to unity and cannot be changed independently from the physical parameters. In the weakly compressible regime, the Mach number is merely a numerical parameter following the law of similarity, so that the time step size can be changed via the flow velocity [5]. In compressible flows, however, the Mach number regains its meaning as a physical quantity with only limited remaining options to change the time step, e.g., by increasing or decreasing the spatial resolution, which is still coupled to the time step size. Alternatively, the speed of sound can be varied, at the price of an inferior numerical stability [4]. Second, the simulation of energy conserving flows requires disproportionately large velocity sets since integer-based values of the particle velocities lead to leading-order errors in the relevant moments of the Maxwell-Boltzmann distribution function [21]. A common way to reduce these errors is to increase the number of abscissae, leading to velocity sets like D1Q7, D2Q49, or D3Q343 [22]. For weakly compressible flows, the D1Q3 velocity set and its tensor products D2Q9 and D3Q27 in combination with a second-order truncation of the Maxwell-Boltzmann distribution mostly match the desired properties of

the Navier-Stokes equations in the hydrodynamic limit, except the "famous $O(\text{Ma}^3)$ error term" [23], which breaks Galilean invariance and restricts simulations to low Mach numbers.

The physical and numerical parameters can be decoupled in an off-lattice framework, where the velocity set does not need to match the computational grid. In the field of off-lattice Boltzmann methods the semi-Lagrangian lattice Boltzmann method (SLLBM) was recently introduced [24] and further investigated in a number of subsequent works [25–29]. To formulate the streaming step the algorithm follows the trajectories of the lattice Boltzmann equation along their characteristics to find the cells of the corresponding departure points. In each cell, a finite element formulation reconstructs the distribution function values from the local degrees of freedom by interpolation polynomials. The SLLBM overcomes the aforementioned drawbacks of the standard lattice Boltzmann method. It decouples the time step from the grid spacing and allows for high-order spatial accuracy on irregular grids. In common with Eulerian-based off-lattice Boltzmann solvers the SLLBM enables the use of non-integer-based velocity sets. Yet, compared to Eulerian-based off-lattice Boltzmann schemes, the streaming step is computationally more efficient [28], reduces numerical dissipation, and is still close to the original formulation of the LBM, as it also follows the trajectories of the discrete Boltzmann equation back in time [24]. The problems that arise with interpolation-based schemes have been discussed in earlier works, e.g., by Chen [10], like the issue that interpolation-based LBM do not conserve mass, momentum, and energy. However, this effect can be at least reduced by using high-order interpolation polynomials, which we showed in a recent work [28].

The present work thoroughly investigates an SLLBM solver for compressible flows. The methodology is related to the Particles on Demand (PoD) method [25], which attained considerable interest [29]. The PoD made use of a semi-Lagrangian streaming step to attain the off-lattice distribution function values in a dynamically shifting reference frame, similar to other adaptive LBM models for compressible flows [30–32]. In contrast to PoD, we restrict the present method to static, nonmoving reference frames at a fixed reference temperature and show that the method still remains competitive in terms of Galilean invariance up to a certain Mach number. In that sense, the present method is closer to the original lattice Boltzmann method formulation, which also operates in a static, nonmoving reference frame.

In addition to the achievements of PoD, we use Hermite-based equilibria [33,34] and high-order interpolation polynomials. We apply the double-population off-lattice approach that mitigates the aforementioned issues of previous solvers and provides a more comprehensive solution for simulating compressible flows with the LBM.

The semi-Lagrangian streaming step also offers more flexibility in constructing velocity sets from Hermite polynomials [35,36], which is particularly relevant for compressible flow. Originating in Grad's work on the moment system in kinetic theory [37], a number of subsequent works showed that consistent equilibrium distribution functions can be derived by projecting the Boltzmann equation onto a low-dimensional Hilbert subspace through Hermite expansion [21,34,36]. This method is used to construct high-order velocity sets that

recover the compressible Fourier-Navier-Stokes equations in the hydrodynamic limit [23,38].

This article is structured as follows. Section II derives the equilibria and velocity sets and recapitulates the semi-Lagrangian advection scheme. Section III validates the method by simulations of a moving incompressible Taylor-Green vortex, a smooth density propagation, a Sod shock tube, a two-dimensional Riemann problem, and a shock-vortex interaction. Section IV discusses the properties of the compressible SLLBM and highlights the advantages over other compressible LBM solvers. Section V provides conclusions.

II. METHODOLOGY

The Boltzmann equation reads

$$\partial_t f + \boldsymbol{\xi} \cdot \boldsymbol{\nabla} f = \Omega(f), \tag{1}$$

with the particle distribution f, the particle velocities ξ, and the collision operator Ω. In this work, only the linearized single relaxation time collision operator proposed by Bhatnagar-Gross-Krook (BGK) [39] will be used:

$$\Omega(f) = -\frac{1}{\lambda}(f - f^{\text{eq}}), \tag{2}$$

where f^{eq} denotes the equilibrium distribution function, while the relaxation time $\lambda = \nu/c_s^2$ reflects the kinematic viscosity ν and the speed of sound c_s. The macroscopic density (ρ), velocity (u), and energy (E) are represented by the moments of the distribution function:

$$\rho = \int f \, d\boldsymbol{\xi}, \tag{3}$$

$$\rho \boldsymbol{u} = \int f \boldsymbol{\xi} \, d\boldsymbol{\xi}, \tag{4}$$

$$2\rho E = \int (f|\boldsymbol{\xi}|^2 + g) \, d\boldsymbol{\xi}. \tag{5}$$

The second distribution function g represents the rotational energy of the polyatomic molecules and is described in Sec. II D.

A. Equilibrium distribution function

One key component to correctly calculate thermal and compressible flows is the construction of the equilibrium distribution function f^{eq}. The present section is based on the comprehensive derivation of the equilibrium, which was detailed by Shan *et al.* [34]. The equilibrium is found via scaled Hermite polynomials $\mathcal{H}^{(n)}(\hat{\boldsymbol{\xi}})$, which are formulated in terms of the normalized particle velocities $\hat{\boldsymbol{\xi}} = \boldsymbol{\xi}/c_s$. The Hermite polynomials are defined by the generating function,

$$\omega(\hat{\boldsymbol{\xi}}) = \frac{1}{(2\pi)^{(D/2)}} \exp\left(-\frac{|\hat{\boldsymbol{\xi}}|^2}{2}\right), \tag{6}$$

as

$$\mathcal{H}^{(n)}(\hat{\boldsymbol{\xi}}) = (-1)^n \frac{1}{\omega(\hat{\boldsymbol{\xi}})} \frac{d^n}{d\hat{\boldsymbol{\xi}}^n} \omega(\hat{\boldsymbol{\xi}}), \tag{7}$$

where D is the dimension and $c_s^2 = RT_0$ is the product of gas constant and reference temperature. (Note that the generating function is differentiated with respect to $\boldsymbol{\xi}$ rather than $\hat{\boldsymbol{\xi}}$.)

The scaled Hermite polynomials up to fourth order are

$$\boldsymbol{\mathcal{H}}^{(0)} = 1,$$

$$\boldsymbol{\mathcal{H}}^{(1)} = \frac{\hat{\boldsymbol{\xi}}}{c_s},$$

$$\boldsymbol{\mathcal{H}}^{(2)} = \frac{\hat{\boldsymbol{\xi}}^2 - \boldsymbol{I}}{c_s^2},$$

$$\boldsymbol{\mathcal{H}}^{(3)} = \frac{\hat{\boldsymbol{\xi}}^3 - 3\,\widehat{\hat{\boldsymbol{\xi}}\boldsymbol{I}}}{c_s^3},$$

$$\boldsymbol{\mathcal{H}}^{(4)} = \frac{\hat{\boldsymbol{\xi}}^4 - 6\,\widehat{\hat{\boldsymbol{\xi}}^2\boldsymbol{I}} + 3\,\widehat{\boldsymbol{I}\boldsymbol{\delta}}}{c_s^4}.$$

Here, $\boldsymbol{I}$ is the identity matrix, $\boldsymbol{\xi}^n := \boldsymbol{\xi} \otimes \boldsymbol{\xi} \otimes \cdots \otimes \boldsymbol{\xi}$ (n times) and for arbitrary vectors, for example, we write $\boldsymbol{ab} := \boldsymbol{a} \otimes \boldsymbol{b}$. Furthermore, we defined for an nth order tensor $\boldsymbol{T}^{(n)}$, according to [40], the relation,

$$\widehat{\boldsymbol{T}^{(n)}} = \frac{1}{\text{number of } \pi} \sum_{\pi} \boldsymbol{T}^{(n)},$$

with π indicating the significant permutations of either the base vectors or indices. For arbitrary vectors $\boldsymbol{v}$ and $\hat{\boldsymbol{\xi}}$ one has, for example,

$$\widehat{\boldsymbol{v}\hat{\boldsymbol{\xi}}^2} = \tfrac{1}{3}(\boldsymbol{v}\hat{\boldsymbol{\xi}}\hat{\boldsymbol{\xi}} + \hat{\boldsymbol{\xi}}\boldsymbol{v}\hat{\boldsymbol{\xi}} + \hat{\boldsymbol{\xi}}\hat{\boldsymbol{\xi}}\boldsymbol{v}).$$

The (equilibrium) distribution functions can be expressed as a Hermite series [23] of order N,

$$f^N(\mathbf{x}, \hat{\boldsymbol{\xi}}, t) \approx \omega(\hat{\boldsymbol{\xi}}) \sum_{n=0}^{N} \frac{1}{n!} \boldsymbol{a}^{(n)}(\mathbf{x}, t) : \boldsymbol{\mathcal{H}}^{(n)}(\hat{\boldsymbol{\xi}}), \quad (8)$$

$$f^{\text{eq},N}(\mathbf{x}, \hat{\boldsymbol{\xi}}, t) \approx \omega(\hat{\boldsymbol{\xi}}) \sum_{n=0}^{N} \frac{1}{n!} \boldsymbol{a}_{\text{eq}}^{(n)}(\mathbf{x}, t) : \boldsymbol{\mathcal{H}}^{(n)}(\hat{\boldsymbol{\xi}}), \quad (9)$$

with : indicating full contraction. The order N determines the physics of the lattice Boltzmann model. The Hermite coefficients $\boldsymbol{a}$ are obtained via projection of f on the orthogonal basis of Hermite tensors. The equilibrium coefficients $\boldsymbol{a}_{\text{eq}}^{(n)}$ are therefore directly related to the moments of the Maxwell-Boltzmann equilibrium, as [23]

$$a_{\text{eq}}^{(0)} = \rho, \quad (10a)$$

$$a_{\alpha,eq}^{(1)} = \rho u_\alpha, \quad (10b)$$

$$a_{\alpha\beta,eq}^{(2)} = \Pi_{\alpha\beta}^{\text{eq}} = \rho(u_\alpha u_\beta + T_0(\theta - 1)\delta_{\alpha\beta}), \quad (10c)$$

$$a_{\alpha\beta\gamma,eq}^{(3)} = \mathcal{Q}_{\alpha\beta\gamma}^{\text{eq}} = \rho[u_\alpha u_\beta u_\gamma + T_0(\theta - 1)(\delta_{\alpha\beta} u_\gamma \cdot \\ + \delta_{\alpha\gamma} u_\beta + \delta_{\beta\gamma} u_\alpha)], \quad (10d)$$

$$a_{\alpha\beta\gamma\delta,eq}^{(4)} = \mathcal{R}_{\alpha\beta\gamma\delta}^{\text{eq}}$$
$$= \rho\big[u_\alpha u_\beta u_\gamma u_\delta + T_0(\theta - 1)((\delta_{\alpha\beta}\delta_{\gamma\delta} \\ + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma})T_0(\theta - 1) \\ + \delta_{\alpha\beta} u_\gamma u_\delta + \delta_{\alpha\gamma} u_\beta u_\delta + \delta_{\alpha\delta} u_\beta u_\gamma \\ + \delta_{\beta\gamma} u_\alpha u_\delta + \delta_{\beta\delta} u_\alpha u_\gamma + \delta_{\gamma\delta} u_\alpha u_\beta)\big]. \quad (10e)$$

TABLE I. Abscissae ξ_i and weights w_i for the D1Q5 lattice. The D2Q25 velocity set is obtained by a tensor multiplication.

i	ξ_i	w_i
0	0	8/15
1,3	$\pm\sqrt{5 - \sqrt{10}}$	$(7 + 2\sqrt{10})/60$
2,4	$\pm\sqrt{5 + \sqrt{10}}$	$(7 - 2\sqrt{10})/60$

Note that in case of the established weakly compressible lattice Boltzmann simulations, the isothermal assumption leads to a local temperature of $\theta = T/T_0 = 1$ and to vanishing terms in Eq. (10).

Truncating at second order covers the well-known lattice Boltzmann model for weakly compressible flows at low velocity. Truncating at third order compromises the nonequilibrium parts of $\mathcal{Q}_{\alpha\beta\gamma}$ due to deviations in $\mathcal{R}_{\alpha\beta\gamma\delta}^{\text{eq}}$. Thus, to represent the heat flux correctly, a fourth-order expansion is required [23,34]. Consequently, to recover the full Fourier-Navier-Stokes equations, the approximation in Eqs. (8) and (9) is of fourth order in the present work.

B. Hermite polynomial velocity sets

To recover the Mth order moment $\boldsymbol{a}^M$, the velocity space is discretized by a (weighted) Gauss-Hermite quadrature,

$$\boldsymbol{a}^M = \int f^N \boldsymbol{\mathcal{H}}^M d\hat{\boldsymbol{\xi}} = \sum_{i=1}^{V} f_i^N \boldsymbol{\mathcal{H}}^M(\hat{\boldsymbol{\xi}}_i), \quad (11)$$

$$\boldsymbol{a}_{\text{eq}}^M = \int f^{\text{eq},N} \boldsymbol{\mathcal{H}}^M d\hat{\boldsymbol{\xi}} = \sum_{i=1}^{V} f_i^{\text{eq},N} \boldsymbol{\mathcal{H}}^M(\hat{\boldsymbol{\xi}}_i), \quad (12)$$

provided $N \geqslant M$ and with the discrete equilibrium,

$$f_i^{\text{eq},N} = w_i/w(\hat{\boldsymbol{\xi}}_i) f^{\text{eq},N}(\hat{\boldsymbol{\xi}}_i).$$

Here, w_i represents the weights coming from a Gauss-Hermite quadrature, and V is the number of velocities in the velocity set. In general, the number of possible velocity sets is several times larger than in on-lattice Boltzmann methods, since the abscissae of the velocity sets are not required to match the grid points. The present work uses a D2Q25 velocity set, which is derived by a tensor multiplication from the one-dimensional D1Q5 velocity set [36]. The one-dimensional abscissae and weights are listed in Table I, while the two-dimensional D2Q25 velocity set is depicted in Fig. 1. Unless scaled, the characteristic lattice speed of sound equals $c_s = 1$.

This off-lattice velocity set is of quadrature order $Q = 9$ and consequently recovers moments up to order $N = 4$, fulfilling the requirement $(M + N)/2 \leqslant Q$ to recover the Fourier-Navier-Stokes equations [41].

C. Variable Prandtl number

In the context of thermal and compressible flows, using the BGK operator limits the Prandtl number Pr, i.e., the ratio of kinematic viscosity ν and thermal diffusivity α, to Pr $= \nu/\alpha = 1$. This drawback can be circumvented by using a multirelaxation time collision operator to decouple both properties of the fluid [42], a quasiequilibrium [6,43] or by
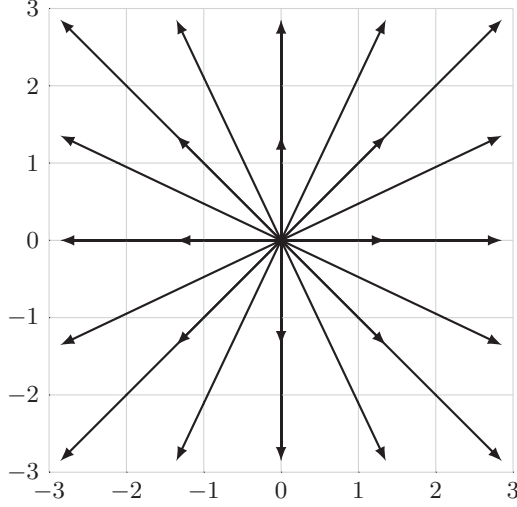
FIG. 1. D2Q25 velocity set based on Hermite polynomials of order 5. Although the velocity can be scaled, matching the regular grid is not possible.

using a BGK-Shakhov model [44]. Although we observed that for the test cases in this work there is no significant difference, the SLLBM solver is capable to adapt the Prandtl number by the quasiequilibrium f_i^* and g_i^* [6,43], which is

$$f_i^* = w_i \bar{\mathbf{Q}}^{\mathrm{neq}} : \left(\hat{\boldsymbol{\xi}}_i^3 - 3 \, \widetilde{\hat{\boldsymbol{\xi}}_i \boldsymbol{I}} \right) / (c_s^3), \tag{13}$$

with the nonequilibrium part of the centered heat flux tensor $\bar{\mathbf{Q}}^{\mathrm{neq}} = \bar{\mathbf{Q}} - \bar{\mathbf{Q}}^{\mathrm{eq}}$,

$$\bar{\mathbf{Q}} = \sum_{i=1}^{n} f_i (\hat{\boldsymbol{\xi}}_i - \boldsymbol{u}) \otimes (\hat{\boldsymbol{\xi}}_i - \boldsymbol{u}) \otimes (\hat{\boldsymbol{\xi}}_i - \boldsymbol{u}), \tag{14}$$

with the same proceeding for g_i^*.

D. Variable heat capacity ratio

As the Boltzmann equation and the Maxwell-Boltzmann distribution are valid in the assumption of monoatomic flows, the heat capacity ratio $\gamma = C_p/C_v$ is fixed to $\gamma = (D + 2)/2$, with the dimension D and the heat capacities at constant pressure and volume C_p and C_v, respectively. Nonetheless, several attempts in the literature solve this issue by using a second set of distribution functions [17,45], inspired by the Rykov model in kinetic theory for polyatomic gases [46], or by adapting the equilibrium distribution to the heat capacity of the fluid [47]. We adopt the ansatz by Frapolli *et al.* for the LBM by incorporating an additional set of distributions g and its equilibrium g^{eq}.

The equilibrium of g^{eq} can easily be determined as

$$g_i^{\mathrm{eq}} = (2C_v - D)\theta f_i^{\mathrm{eq}}, \tag{15}$$

with $\gamma = R/C_v + 1$, where the gas constant R is usually set to unity (without loss of generality). The distribution function g_i follows the same stream and collide algorithm as the distribution function f_i and is ultimately used in Eq. (5).

The lattice Boltzmann equations of the present model read

$$f_i(x, t) = f_i(x - \delta_t \xi_i, t - \delta_t)$$
$$- \frac{1}{\tau} \Big[f_i(x - \delta_t \xi_i, t - \delta_t) - f_i^{\mathrm{eq}}(x - \delta_t \xi_i, t - \delta_t) \Big]$$
$$+ \left(\frac{1}{\tau} - \frac{1}{\tau_{\mathrm{Pr}}} \right) f_i^*(x - \delta_t \xi_i, t - \delta_t), \tag{16a}$$

$$g_i(x, t) = g_i(x - \delta_t \xi_i, t - \delta_t)$$
$$- \frac{1}{\tau} \Big[g_i(x - \delta_t \xi_i, t - \delta_t) - g_i^{\mathrm{eq}}(x - \delta_t \xi_i, t - \delta_t) \Big]$$
$$+ \left(\frac{1}{\tau} - \frac{1}{\tau_{\mathrm{Pr}}} \right) g_i^*(x - \delta_t \xi_i, t - \delta_t), \tag{16b}$$

with the relaxation time $\tau = \nu/(c_s^2 \delta_t) + 0.5$, where the term of 0.5 is a consequence of the second-order time integration in terms of the trapezoidal rule [48–50]. The relaxation time $\tau_{\mathrm{Pr}} = (\tau - 0.5)/\mathrm{Pr} + 0.5$ is related to the Prandtl number Pr. In contrast to the usual lattice Boltzmann algorithm, neither the time step size δ_t, nor the expression $\delta_t \xi_i$ are required to be integers due to the use of a semi-Lagrangian streaming step.

E. Semi-Lagrangian streaming step

To obtain the off-vertex distribution function values f_i (and likewise g_i), the SLLBM uses a cell-wise polynomial interpolation. Therefore, the computational domain is divided into a finite number of cells N_Ξ with Lagrange polynomials ψ_Ξ of order p as shape functions. The distribution function values at the support points in cell Ξ are denoted as $\hat{f}_{i\Xi j}$, so that

$$f_i(\mathbf{x}, t) = \sum_{j=1}^{N_j} \hat{f}_{i\Xi j}(t) \psi_{\Xi j}(\mathbf{x}), \tag{17}$$

for all points $\mathbf{x}$ in cell Ξ.

The distributions $\hat{f}_{i\Xi j}$ are updated via Eq. (16), where the value at the departure point $f_i(x - \delta_t \xi_i)$ is obtained via the interpolation defined in Eq. (17). An equidistant interpolation is stable for low-order interpolation polynomials only. Therefore, at higher interpolation orders this work applies Gauß-Lobatto-Chebyshev support points to dampen the Runge phenomenon and to increase the stability [24,51]. Gauß-Lobatto-Chebyshev support points in one dimension read

$$x_k = \frac{1}{2} \left[-\cos \left(\frac{k - 1}{n - 1} \pi \right) + 1 \right], \quad k = 1, 2, \ldots, n.$$

The two-dimensional unit cell used in this work is depicted in Fig. 2.

The interpolation error of semi-Lagrangian advection schemes has been investigated, for instance, by Einkemmer and Ostermann [52] or Falcone and Ferretti [53]. For the SLLBM used in the present work, we have previously discussed the nontrivial behavior of the temporal discretization error [28], which is of order,

$$\mathcal{O} \left(\min \left(\frac{\delta_x^{p+1}}{\delta_t}, \delta_x^p \right) + \delta_t^2 \right). \tag{18}$$
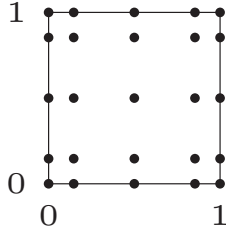
FIG. 2. Distribution of Gauß-Lobatto-Chebyshev support points on a two-dimensional reference cell with $p = 4$.

This means in practice that the error has a minimum for intermediate CFL numbers and is bounded by the spatial discretization error in the limit of small time steps.

All SLLBM simulations in this manuscript have been performed with the NATRIUM library [28], which makes use of the finite element library DEAL.II [54]. See [28] for further details concerning the implementation.

F. Thermo-hydrodynamic limit

The thermo-hydrodynamic limit of the presented equilibrium distribution function was presented and thoroughly discussed by Malaspinas *et al.* [38] and by Coreixas *et al.* [23], who found

$$\partial_t(\rho) + \partial_\alpha(\rho u_\alpha) = 0, \tag{19}$$

$$\partial_t(\rho u_\alpha) + \partial_\beta(\rho u_\alpha u_\beta) = \partial_\alpha \sigma_{\alpha\beta},$$
$$\partial_t(\rho E) + \partial_\alpha(\rho E u_\alpha) = \partial_\alpha(\kappa \partial_\alpha T) + \partial_\alpha(\sigma_{\alpha\beta} u_\beta), \tag{20}$$

with the stress tensor,

$$\sigma_{\alpha\beta} = -P\delta_{\alpha\beta} + \nu \partial_\beta [\rho(\partial_\beta u_\alpha + \partial_\alpha u_\beta)] - \zeta \partial_\gamma u_\gamma \delta_{\alpha\beta}, \tag{21}$$

with the kinematic viscosity,

$$\nu = \left(\tau - \tfrac{1}{2}\right) c_s^2 \theta \delta_t, \tag{22}$$

and with the heat conductivity,

$$\kappa = \left(\tau_{\mathrm{Pr}} - \tfrac{1}{2}\right) C_p c_s^2 \theta \delta_t. \tag{23}$$

The bulk viscosity equals [22]

$$\zeta = \rho \nu \left(\frac{1}{C_v} - \frac{2}{D}\right), \tag{24}$$

with C_v and C_p being the heat capacity at constant volume and pressure, respectively.

G. On-lattice Boltzmann solver for comparison

Section III C compares the SLLBM with an on-lattice Boltzmann solver. For the latter, we chose the D2V37 velocity set proposed by Philippi *et al.* [55], which is of the same quadrature order $Q = 9$ as the D2Q25 in this work. Additionally, the equilibrium in Sec. II A and the second distribution function in Sec. II D were used to obtain a variable adiabatic exponent γ. The on-lattice simulations were performed by the solver LETTUCE [56], which is based on the machine learning framework PYTORCH [57]. The code is written in PYTHON and makes use of the PYTORCH routines to enable GPU simulations.

III. RESULTS

In this section, a variety of test cases will be studied. First, the errors of the method are quantified by simulations of the two-dimensional incompressible Taylor-Green vortex. Then the accuracy will be confirmed by a smooth density propagation. The Sod shock tube shows the general capability of the SLLBM to deal with shocks at the original density ratio presented by Sod [58]. Then the two-dimensional Riemann problem will be studied, followed by a simulation of a shock-vortex interaction on nonuniform grids.

A. Moving two-dimensional Taylor-Green vortex

The SLLBM was validated by a moving incompressible two-dimensional Taylor-Green vortex. As it comes along with a reference solution it is suited to quantify the interpolation-caused errors and to show the high degree of Galilean invariance if an appropriate velocity set is used. The initial and reference solution is

$$u_x^{\mathrm{ref}}(x, y, t) = +u_v \sin(x - u_h t)\cos(y)\exp(-2\nu t),$$
$$u_y^{\mathrm{ref}}(x, y, t) = -u_v \cos(x - u_h t)\sin(y)\exp(-2\nu t),$$
$$P^{\mathrm{ref}}(x, y, t) = \tfrac{1}{4}(\cos(2(x - u_h t)) + \cos(2y))\exp(-4\nu t),$$

with the constant horizontal velocity u_h and the vortex velocity u_v. The Reynolds number was set to $\mathrm{Re} = u_v l/\nu = 10$, with an initial maximum speed $u_v = 1$ on a domain of $l = 2\pi$. All units were transferred to lattice units; the initial Mach number of the vortex flow was constantly $\mathrm{Ma}_v = 0.01$, while the horizontal Mach number was either disabled for case (a) or set to $\mathrm{Ma}_h = 0.05$ for case (b). The results were compared to the reference at $t = 1.83$, corresponding to a decay of the vortex velocity to $u_{\max} = 0.1 u_v$. The grid consisted of 4, 8, 16, and 32 cells per direction times the interpolation order $p = 4$, yielding 16, 32, 64, and 128 grid points per direction. The time step was set to $\delta_t = 0.0008$ for the coarsest test case and halved for each refinement step, without changing the Mach numbers.

We compare two velocity sets: the standard D2Q9 velocity set and the D2Q25 velocity set, both with the SLLBM advection step. The fourth-order equilibrium specified in Sec. II A was used in an isothermal configuration, i.e., $\theta = 1$.
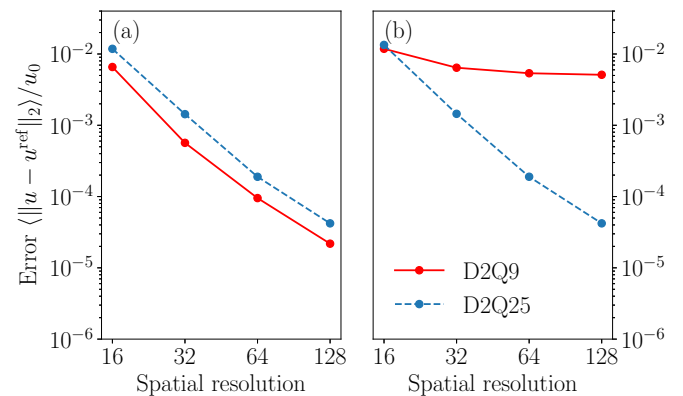

FIG. 3. Numerical errors of incompressible 2D Taylor-Green vortex benchmarks with horizontal Mach numbers $\mathrm{Ma}_h = 0$ (a) and $\mathrm{Ma}_h = 0.05$ (b).
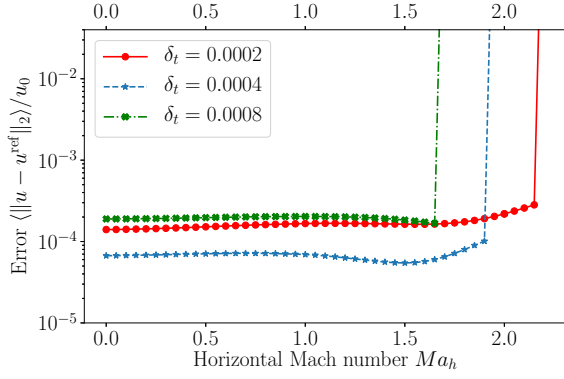
FIG. 4. Numerical errors of incompressible 2D Taylor-Green vortex benchmarks over horizontal Mach numbers Ma_h with the D2Q25 velocity set for three different time step sizes. The resolution was 64 grid points per direction.

Figure 3 illustrates the errors of the two-dimensional Taylor-Green vortex flow. Figure 3(a) shows the nonmoving case. As detailed in Sec. II E the interpolation provoked nonnegligible errors, which were in total two to three times larger for the D2Q25 velocity set in comparison to the D2Q9, caused by the different amount of interpolations needed to perform one time step. However, the larger velocity set pays off for constantly moving vortices in case Fig. 3(b). Here the D2Q9 simulation converged towards an error, which is caused by the lacking Galilean invariance of this particular velocity set. In contrast, the D2Q25's errors were nearly unchanged compared to simulations [Fig. 3(a)] with $u_h = 0$.

This test case also served to evaluate the maximum Mach number ranges of the D2Q25 velocity set. To achieve this, the horizontal velocity Ma_h was increased until the simulations became unstable, while the initial vortex velocity remained at $Ma_v = 0.01$. Figure 4 depicts the numerical errors over the horizontal Mach number Ma_h for three different time step sizes with two main findings. First, there is a local minimum of the numerical error in case of the time step size $\delta_t = 0.0004$. Second, for all three time step sizes the numerical errors remain more or less constant over the range of horizontal Mach numbers, while the Mach number limit was abruptly exceeded. Obviously, the maximum Mach number of semi-Lagrangian schemes is not solely determined by equilibrium and velocity set, but also by the time step size.

TABLE II. Numerical errors of the smooth density propagation benchmark.

Resolution	Error $\|\rho - \rho_{\text{ref}}\|_\infty$	Order
16×16	2.10599×10^{-2}	–
32×32	1.32533×10^{-3}	4.0
64×64	5.36948×10^{-5}	4.6
128×128	4.68312×10^{-6}	3.5
256×256	7.90318×10^{-7}	2.6
512×512	2.14017×10^{-7}	1.9

B. Smooth density propagation

To measure the accuracy of the SLLBM for compressible flows, the smooth density propagation [59,60] was evaluated. The exact solution is

$$\rho(x, y, t) = 1 + 0.2 \sin(\pi(x - u_x t)) \sin(\pi y), \quad (25)$$

$$u_x(x, y, t) = 1, u_y(x, y, t) = 0, P(x, y, t) = 1. \quad (26)$$

The domain was $x, y \in [-1, 1]$ and the simulation end time was $t = 2$. For the coarsest grid of 16×16 grid points (i.e., 4×4 cells and $p = 4$), this time was reached in 1105 steps; the number of steps doubled for each finer simulation. The kinematic viscosity was set to $\nu = 10^{-10}$; the Mach number was set to $Ma = 0.2$.

Table II depicts the numerical errors of the density for six different resolutions. At coarse resolutions the numerical errors decreased with approximately fourth order. At the finest resolutions, the second-order temporal order became dominant.

C. Sod shock tube

In a first test case with discontinuities, the well-known Sod shock tube [58] was simulated. The one-dimensional domain $x \in [0, 1]$ was divided into two regions at $x = 0.5$ with the initial conditions,

$$\rho_0 = 8, u_0 = 0, P_0 = 10,$$

$$\rho_1 = 1, u_1 = 0, P_1 = 1,$$

with $P = \rho RT$ being the pressure according to the ideal gas law and $R = 1$. The periodic domain was discretized using
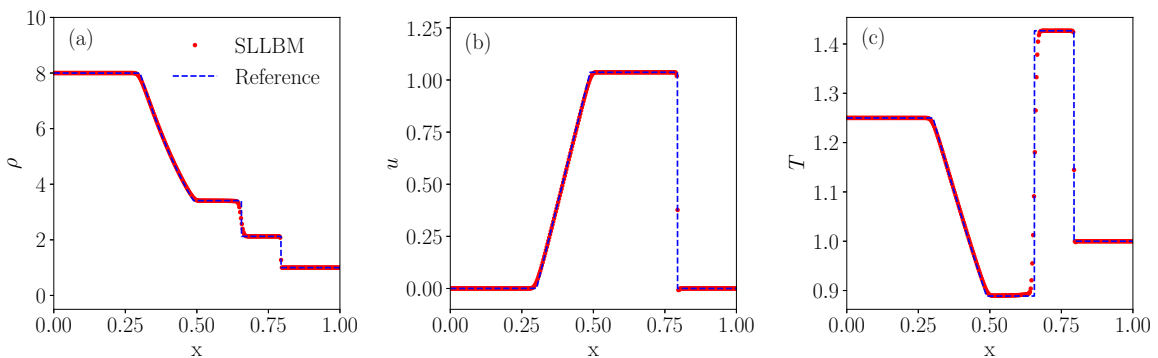

FIG. 5. Sod shock tube with $p = 4, N = 800 \, (200 \, \text{cells}), \delta t = 0.002$ after 735 iterations showing (a) density ρ, (b) velocity u, and (c) temperature T. The SLLBM is compared with Sod's reference results [58].
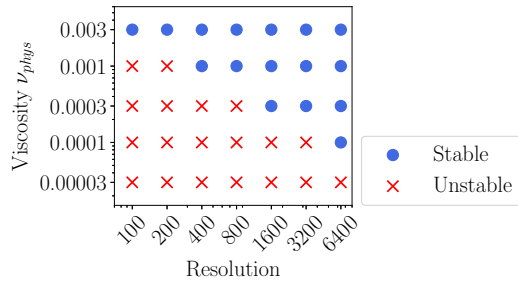
FIG. 6. Stable and unstable Sod shock tube simulations of the on-lattice Boltzmann solver detailed in Sec. II G.

TABLE III. Initial conditions of the 2D Riemann problem according to Lax and Liu [61].

$\rho = 1$	$\rho = 0.5313$
$u_x = 0.7276$	$u_x = 0$
$u_y = 0$	$u_y = 0$
$P = 1$	$P = 0.4$
$\rho = 0.8$	$\rho = 1$
$u_x = 0$	$u_x = 0$
$u_y = 0$	$u_y = 0.7276$
$P = 1$	$P = 1$

800 grid points; the number of cells was 200 for $p = 4$. The kinematic viscosity, Prandtl number, and adiabatic exponent were set to $\nu = 0.00012$, Pr $= 1.0$, and $\gamma = 1.4$, respectively. With a time step size of $\delta_t = 0.002$ this resulted in a relaxation time of $\tau = \nu_{\mathrm{LB}}/(c_s^2 \delta_t) + 0.5 = 1.518$. Figure 5 depicts the density, velocity, and temperature for the polynomial order $p = 4$ at $t = 0.15$, which is reached after 735 time steps.

The results accurately matched the reference solution [58] with only minor oscillations visible, comparable to other recently introduced methods [60].

The proposed method in our tests demonstrated clearly improved numerical stability compared to previous on-lattice methods. To compare the stability properties of the SLLBM and of the representative on-lattice Boltzmann solver, specified in Sec. II G, the Sod shock tube was again simulated for a variety of resolutions and physical viscosities, ν_{phys}, until $t = 1.5$. A simulation was rated as stable, if the simulation finished successfully and all macroscopic values were still bounded. Figure 6 shows that the on-lattice Boltzmann solver was unstable for most of the lower viscosities at low resolutions. A simulation at $\nu = 0.0001$, an example of an often used setting for Sod shock tubes in the literature, was only stable for the on-lattice LBM using an excessive resolution of $N = 6400$. Contrary to that, the flexible time step size of the SLLBM allowed for stable simulations even at low resolutions. Figure 7 depicts all stable simulations for three different time step sizes $\delta_t = 0.25\delta_t'$, $\delta_t = 0.50\delta_t'$, and $\delta_t = 1.00\delta_t'$, where δ_t' is the time step size of the respective on-lattice simulation. All of the representative resolution and viscosity scenarios were simulated stably with at least one time step size by the SLLBM.
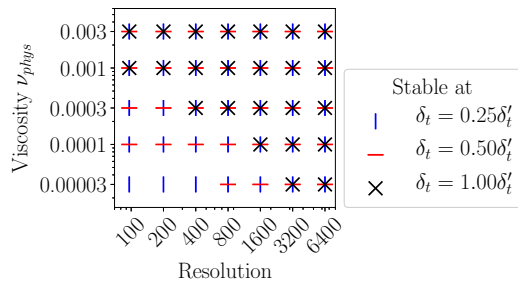
Taken together, the Sod shock tube shows that the proposed SLLBM solver is capable to accurately capture shocks and convinces by a larger stability region than the D2V37 on-lattice Boltzmann simulation.

D. 2D Riemann

The two-dimensional Riemann problem is another demanding test case for compressible inviscid flow solvers, which was deeply investigated for a variety of initial configurations by Lax and Liu [61] and Kurganov and Tadmor [62]. The original works made use of Riemann solvers and semidiscrete finite difference schemes, respectively, to obtain the reference solutions.

The domain $(x, y) \in [0, 1]$ was divided into four quadrants with the initial conditions shown in Table III. The simulation domain was discretized into $N_\Xi = 128 \times 128$ cells with an order of finite elements of $p = 4$, which results in $N = 512 \times 512$ grid points. The relaxation time $\tau = 1.94815$ was determined according to $\tau = \nu/(c_s^2 \delta_t) + 0.5$. The physical simulation end time was $t_{\mathrm{end}} = 0.25$, which was reached after 2091 iterations, i.e., the viscosity is $\nu = 0.000173$, assuming the reference temperature $T_0 = 1$. The heat capacity ratio was set to $\gamma = 1.4$. At the boundaries, a zero-gradient condition, i.e., $\partial f = 0$, was applied. From a practical point of view, to free the simulation domain from disturbing boundary interactions, the domain was set to twice the size and clipped during post-processing. At first, Gauss-Lobatto-Chebyshev nodes were used to distribute the support points within each cell. Figure 8 shows the density contours after 2091 iterations. In total, 45 equally distributed contours were applied. The results were in good agreement with the reference solution in [61,62], although minor oscillations were still visible near the shock fronts.

Figure 9 depicts the pressure, velocity, and temperature charts along the $y = x$ diagonal at the simulation end time. Although there is no reference solution available, the plot gives a good impression of the strong discontinuities that are also the main source of instabilities in this test case.

To emphasize the value of nonequidistant support points for the interpolation procedure, the same configuration was simulated again with equidistant nodes. The simulation became unstable shortly after 200 time steps. Figure 10 shows the pressure along the $y = x$ axis and reveals the heavy oscillations of the simulation with equidistant nodes shortly before the simulation diverges.

FIG. 7. Stable Sod shock tube simulations of the SLLBM at three different time step sizes, where δ_t' is the time step size of the on-lattice method in Fig. 6.
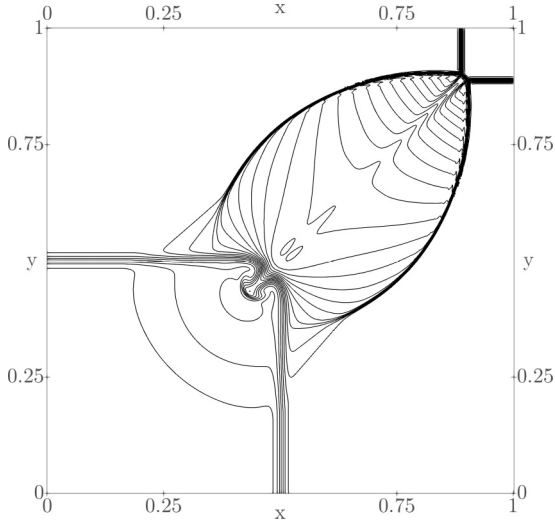
FIG. 8. Outline of 45 equally spaced density contours of the 2D Riemann problem with $N = 512 \times 512$ in the interval $\rho \in [0.412, 1.753]$. The order of finite elements was $p = 4$. The simulation end time was reached in 2091 iterations. Reference solutions obtained by two different solvers are available in [61] and [62].

E. Shock vortex interaction

The final test case was a 2D shock vortex interaction, which was originally presented by Inoue and Hattori [63]. For this simulation a transformed grid was used, which shows the applicability of the SLLBM to nonuniform grids. The simulation setup was as follows: The domain consists of two regions, which are separated by a steady shock. The Mach number of region A is $\mathrm{Ma}_A = -1.2$, while the Mach number in region B $\mathrm{Ma}_B \approx -0.842$ is determined by the Rankine-Hugoniot condition. A vortex with characteristic Mach number $\mathrm{Ma}_v = 0.5$ is advected in region A towards the shock (from right to left) and interacts with it throughout the simulation. The flow field of the vortex is given by

$$u_\theta(r) = \sqrt{\gamma T} \mathrm{Ma}_v r \exp((1 - r^2)/2). \qquad (27)$$

The initial pressure and density field are

$$P(r) = \frac{1}{\gamma} \left(1 - \frac{\gamma - 1}{2} \mathrm{Ma}_v \exp(1 - r^2) \right)^{\gamma/(\gamma-1)}, \qquad (28)$$

FIG. 10. Comparison of the pressure P with Gauß-Lobatto-Chebyshev nodes and conventional equidistant nodes at order of finite elements of $p = 4$ after 200 time steps. The test run with equidistant nodes provoked heavy oscillations, which resulted in a crash shortly afterwards.

and

$$\rho(r) = \left(1 - \frac{\gamma - 1}{2} \mathrm{Ma}_v \exp(1 - r^2) \right)^{1/(\gamma-1)}. \qquad (29)$$

The temperature field can be determined via the ideal gas law $P = \rho RT$.

The Reynolds number is defined by

$$Re = 400 = \frac{c_s R}{\nu}, \qquad (30)$$

with R being the resolution of the characteristic radius r. The Prandtl number is $\mathrm{Pr} = 0.75$.

The simulation was run on a physical domain of size 60×24, which was discretized by $N_\Xi = 256 \times 256$ cells and the polynomial order was $p = 4$, resulting in 1024 lattice nodes in each direction. However, to highly resolve the steady shock the grid was transformed in the x direction according to a simple transformation function:

$$x_{\mathrm{trans}} = \frac{30}{\pi} \left(\Lambda \sin \left(\frac{\pi}{30}(x) \right) + \frac{\pi}{30} x \right), \qquad (31)$$
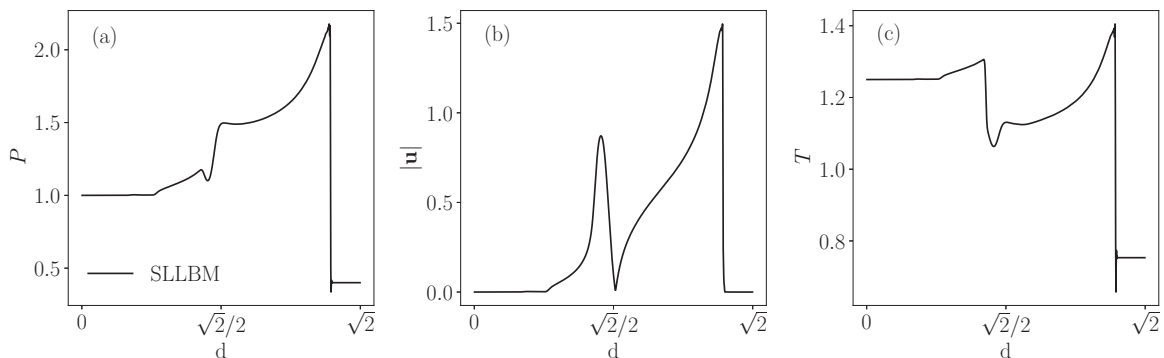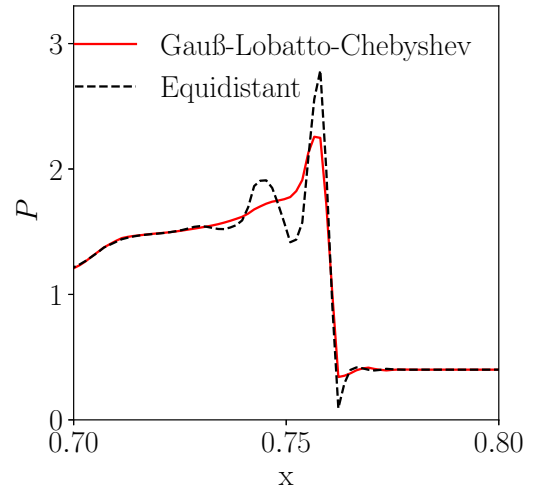

FIG. 9. Pressure, velocity, and temperature curves of the two-dimensional Riemann problem, shown in Fig. 8 along the $y = x$ diagonal d.
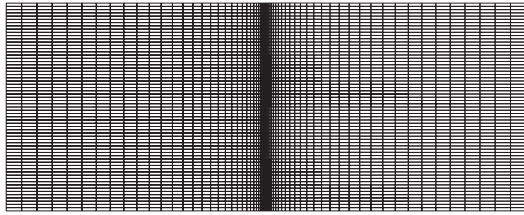
FIG. 11. Schematic grid for the shock vortex interaction. The actual grid used in the simulation was further refined by splitting each cell in half along each axis. Each cell contains 5×5 support points, whereby the outer support points are shared with the neighboring cells.

where Λ is a stretching parameter, which was set to $\Lambda = 0.95$. A similar refinement was done by Inoue and Hattori in the original work [63]. Although the shock slightly moved during the interaction with the shock, the vortex was still located in the refinement zone, which led to satisfactory results and reduced the amount of needed grid points. The shock was located at $x_{shock} = 30$ and the initial vortex center position was $x_{vortex} = 32$. We observed that the DNS results by Inoue and Hattori were only matched if the vortex was initialized exactly at this position although partly located in the shock. The schematic grid of the computation is shown in Fig. 11. The average resolution of the characteristic radius along the x axis r was $1024/60 \approx 17.07$ grid points and $1024/24 \approx 42.67$ grid points along the y axis. The large domain and periodic boundary conditions in all directions were sufficient to keep the domain of interest free from disturbing influences of the boundaries. The characteristic time is defined as $t' = R/c_s$. The time step size was $\delta_t = 7 \times 10^{-4}$ so that the final result at $t = 8$ was reached in 11 290 iterations.

Figure 12 shows the density contours in the range $\rho \in [0.92, 1.55]$ with 119 contours. The contours were in good agreement with previous works, e.g., [17,63]. Both the static
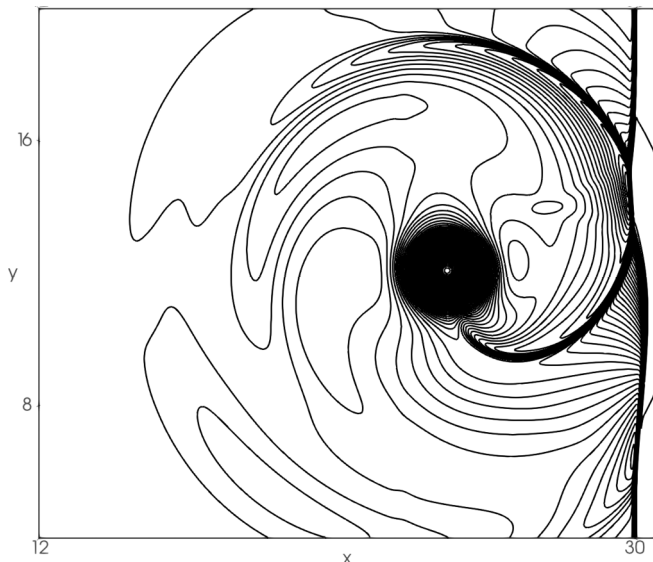

FIG. 12. Density contours of the shock-vortex interaction with $Ma_v = 0.5$ and $Re = 400$ in the range $\rho \in [0.92, 1.55]$ in 119 steps.
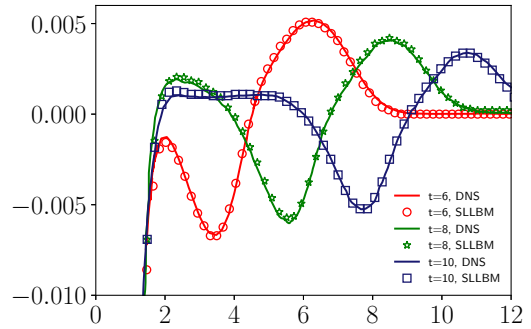

FIG. 13. Sound pressure $\Delta P = (P - P_B)/P_B$ of the shock-vortex interaction with $Re = 800$, $Ma_v = 0.25$, and $Ma_A = 1.2$ at times $t' = 6$, $t' = 8$, and $t' = 10$. The measurements were taken from the center of the vortex along the radius with $-45°$ with respect to the x axis. DNS from [63].

shock and the attached shock were well resolved and the density contours appeared without any oscillations.

To quantify the results, the radial sound pressure in a simulation with $Ma_v = 0.25$ and $Re = 800$ was evaluated and compared with DNS results from Inoue and Hattori [63]. Figure 13 shows the normalized pressure $\Delta P = (P - P_B)/P_B$), where P_B denotes the initial pressure in region B. The results of the present configuration were in excellent agreement with the reference solution obtained by the DNS.

The total mass in the previously presented well-resolved test cases was found to be more or less constant over the entire simulation time. However, since the SLLBM is not rigorously mass-conserving, Figure 14 presents the evolution of mass over 1500 simulation steps for a slightly under-resolved configuration of $Re = 800$, $Ma = 0.25$, and 512×512 grid points at two polynomial orders. In case of a low-order polynomial interpolation of $p = 2$, there was a minor but steady loss of mass, which amounted to approximately 0.05% of the initial mass after 1500 steps. The mass conservation was tighter for the high-order interpolation. For $p = 4$, the mass oscillated around the initial value as opposed to a systematic gain or loss.
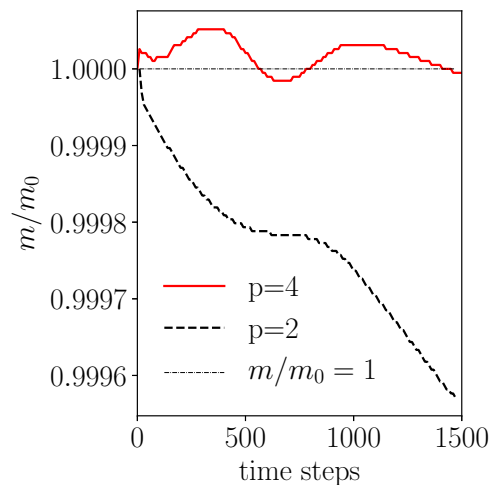

FIG. 14. Relative mass m/m_0 for polynomial orders $p = 4$ and $p = 2$ over the number of iterations for the test case shown in Fig. 12.

IV. DISCUSSION

As detailed in the introduction of this work, the research on lattice Boltmann methods for compressible flows is ongoing. We have shown how the use of an efficient off-lattice streaming step enables compressible simulations by relaxing the coupling between time step, computational grid, and particle velocities in the LBM.

Originating from the usual lattice Boltzmann equation the SLLBM algorithm follows the trajectories along the characteristics and recovers the off-lattice departure points using a finite element interpolation. In common with finite volume or finite difference LBMs, this procedure allows for a flexible time step size. This key feature is missing in usual on-lattice Boltzmann solvers for compressible flows. Contrary to the Eulerian methods, the SLLBM avoids costly and often high-dissipative time integration. The compressible SLLBM is related to the PoD method, but operates in a static, nonmoving reference frame. Despite this regression, the SLLBM was capable to simulate a variety of compressible test cases. As the weights in the present model are not temperature-dependent, the temperature ratio is—in principle—not limited. Simple temperature diffusion tests not shown in this article revealed a stable temperature ratio of at least 1:1000.

We tested the compressible SLLBM for five common test cases, in both inviscid and viscous regimes. The two-dimensional Taylor-Green vortex showed the high degree of Galilean invariance of the scheme. The accuracy test by a smooth density propagation confirmed the fourth order in space and second order in time of the scheme. The Sod shock tube in its original configuration and a pressure ratio of $P_1/P_2 = 10$ were successfully tested with the SLLBM. In contrast, on-lattice Boltzmann solvers like the one presented in [8] often restrict the pressure ratio to lower values on larger grids. In contrast, the SLLBM's time step size can be tuned to the needs of the simulation, even with the most common BGK model. Furthermore, we employed fourth-order interpolation polynomials, rendering the solution in the smooth areas spatially to the corresponding order [24], which is another unique property of this scheme. However, as shown in the two-dimensional Riemann problem in Sec. III D, leveraging the spatial order is only workable for an appropriate distribution of support points within each cell. While equidistant support points led to unstable simulations, the use of fourth-order Gauss-Lobatto-Chebyshev points enabled stable simulations. Moreover, the simulation of the shock-vortex interaction showed that a high-order interpolation improves mass conservation, which is not rigorously enforced in interpolation-based LBM [29]. The equilibrium distribution function based on a Hermite expansion of fourth order is computationally costlier than the usual second order expansion, but conserves energy as required for compressible flows [34]. Concerning the discretization of the velocity space, a D2Q25 velocity set based on the fifth-order Hermite polynomial was used, which is suited to recover the moments up to fourth order correctly [21]. In contrast, on-lattice velocity sets usually provoke deviations in the recovery of the high-order moments [21]. The combination of the equilibrium and velocity set allowed for a total Mach number of Ma = 1.7 in the simulation of the shock-vortex interaction, but the results of the 2D Taylor-Green vortex flow suggest that even higher Mach numbers are attainable, depending on the time step size. A stability analysis [64,65] could clarify the exact limits of the scheme. An obvious way to further increase the Mach number is to incorporate statically shifted [22,66] or dynamically shifted lattices [25]. Future works should thoroughly investigate the path to extend the velocity sets to higher velocities in three dimensions, while keeping the number of discrete velocities lower than 125. The effects of lattice pruning, which were thoroughly studied in the weakly compressible case, e.g., D3Q27 to D3Q19 [67], should also be examined for these high-order velocity sets. Additionally, we propose to pursue an approach that was originally supposed by Watari and Hattori [68] for finite difference LBMs. The authors combined platonic solids, dodecahedra, and icosahedra, to construct highly isotropic velocity sets at different velocity shells. It seems to be worth investigating if these velocity sets allow increasing the Mach number in SLLBM simulations. Corresponding results will be published in an upcoming article.

Another issue is the second distribution function value, needed to correctly represent the heat capacity ratio γ. This model was also successfully used by other authors [6,8,17]. A reduction of the velocity set size of g could possibly make the compressible LBM more attractive.

Taken together, the presented SLLBM opens an alternative and attractive field of solvers for compressible flows and, at the same time, combines many insights from previous LBM solvers for compressible flows.

V. CONCLUSIONS

In this article we showed that the semi-Lagrangian streaming step enables efficient lattice Boltzmann simulations of compressible flows even in a static, nonmoving reference frame. In contrast to other methods, the vertices are organized in cells, and interpolation polynomials up to fourth order are used to attain the off-vertex distribution function values.

The presented SLLBM solver for compressible flows convinces by six main advantages.

(1) Flexibility of the time step size.

(2) Absence of a dedicated time integrator as the integration is performed along characteristics.

(3) Applicability to stretched or unstructured meshes.

(4) Possibility to employ advantageous velocity sets.

(5) Limitation of mass loss by high-order polynomials.

(6) Fourth order of convergence in space and second order in time.

The test cases of a 2D Taylor Green vortex, a smooth density propagation, a Sod shock tube, a 2D Riemann problem, and a shock-vortex interaction on transformed grids show the general applicability to flows with discontinuities. Thereby, Gauss-Lobatto-Chebyshev support points reduce oscillations near the shock fronts.

In our future work, the presented framework will be applied to three-dimensional test cases with a focus on further increasing the time step size and on reducing the number of discrete velocities.

ACKNOWLEDGMENTS

[1] G. R. McNamara and G. Zanetti, Phys. Rev. Lett. **61**, 2332 (1988).

[2] F. J. Higuera and J. Jiménez, Europhys. Lett. **9**, 663 (1989).

[3] S. Chen, H. Chen, D. Martnez, and W. Matthaeus, Phys. Rev. Lett. **67**, 3776 (1991).

[4] F. J. Alexander, H. Chen, S. Chen, and G. D. Doolen, Phys. Rev. A **46**, 1967 (1992).

[5] T. Krüger, H. Kusumaatmaja, A. Kuzmin, O. Shardt, G. Silva, and E. M. Viggen, *The Lattice Boltzmann Method: Principles and Practice* (Springer International Publishing Switzerland, Basel, 2017).

[6] N. Frapolli, S. S. Chikatamarla, and I. V. Karlin, Phys. Rev. E **92**, 061301 (2015).

[7] M. Atif, M. Namburi, and S. Ansumali, Phys. Rev. E **98**, 053311 (2018).

[8] M. H. Saadat, F. Bösch, and I. V. Karlin, Phys. Rev. E **99**, 013306 (2019).

[9] J. Latt, C. Coreixas, J. Beny, and A. Parmigiani, arXiv:1910.13515.

[10] H. Chen, Phys. Rev. E **58**, 3955 (1998).

[11] R. Zhang, H. Chen, Y. Qian, and S. Chen, Phys. Rev. E **63**, 056705 (2001).

[12] Z. Guo, R. Wang, and K. Xu, Phys. Rev. E **91**, 033313 (2015).

[13] Y. Feng, P. Sagaut, and W. Q. Tao, Comput. Fluids **131**, 45 (2016).

[14] W. Shi, W. Shyy, and R. Mei, Numer. Heat Trans., Part B: Fundamentals **40**, 1 (2001).

[15] Y. Wang, Y. L. He, T. S. Zhao, G. H. Tang, and W. Q. Tao, Int. J. Mod. Phys. C **18**, 1961 (2007).

[16] X. Nie, X. Shan, and H. Chen, in *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition* (American Institute of Aeronautics and Astronautics, Reston, 2009).

[17] N. Frapolli, Ph.D thesis, ETH Zürich, Switzerland, 2017.

[18] T. Kataoka and M. Tsutahara, Phys. Rev. E **69**, 056702 (2004).

[19] Q. Li, Y. L. He, Y. Wang, and W. Q. Tao, Phys. Rev. E **76**, 056705 (2007).

[20] C. Coreixas, Ph.D thesis, INP Toulouse, France, 2018.

[21] S. S. Chikatamarla and I. V. Karlin, Phys. Rev. E **79**, 046701 (2009).

[22] N. Frapolli, S. S. Chikatamarla, and I. V. Karlin, Phys. Rev. E **93**, 063302 (2016).

[23] C. Coreixas, G. Wissocq, G. Puigt, J. F. Boussuge, and P. Sagaut, Phys. Rev. E **96**, 033306 (2017).

[24] A. Krämer, K. Küllmer, D. Reith, W. Joppich, and H. Foysi, Phys. Rev. E **95**, 023305 (2017).

[25] B. Dorschner, F. Bösch, and I. V. Karlin, Phys. Rev. Lett. **121**, 130602 (2018).

[26] A. Krämer, Ph.D thesis, University of Siegen, Germany, 2017.

[27] G. Di Ilio, B. Dorschner, G. Bella, S. Succi, and I. V. Karlin, J. Fluid Mech. **849**, 35 (2018).

[28] A. Krämer, D. Wilde, K. Küllmer, D. Reith, H. Foysi, and W. Joppich, Comput. Math. Appl. **79**, 34 (2020).

[29] A. V. Zakirov, B. A. Korneev, V. D. Levchenko, and A. Y. Perepelkina, Keldysh Inst. Prepr., 1 (2019), doi: 10.20948/prepr-2019-35-e.

[30] C. Sun, Phys. Rev. E **58**, 7283 (1998).

[31] C. Sun, J. Comput. Phys. **161**, 70 (2000).

[32] C. Sun and A. T. Hsu, Phys. Rev. E **68**, 016303 (2003).

[33] X. Shan and X. He, Phys. Rev. Lett. **80**, 65 (1998).

[34] X. Shan, X. F. Yuan, and H. Chen, J. Fluid Mech. **550**, 413 (2006).

[35] A. Bardow, I. V. Karlin, and A. A. Gusev, Phys. Rev. E **77**, 025701 (2008).

[36] X. Shan, J. Comput. Sci. **17**, 475 (2016).

[37] H. Grad, Commun. Pure Appl. Math. **2**, 331 (1949).

[38] O. Malaspinas, Ph.D thesis, EPFL Lausanne, Switzerland, 2009.

[39] P. L. Bhatnagar, E. P. Gross, and M. Krook, Phys. Rev. **94**, 511 (1954).

[40] K. Suchy, Australian J. Phys. **49**, 1135 (1996).

[41] X. B. Nie, X. Shan, and H. Chen, Europhys. Lett. **81**, 34005 (2008).

[42] X. Shan and H. Chen, Int. J. Mod. Phys. C **18**, 635 (2007).

[43] S. Ansumali, S. Arcidiacono, S. S. Chikatamarla, N. I. Prasianakis, A. N. Gorban, and I. V. Karlin, Eur. Phys. J. B **56**, 135 (2007).

[44] E. M. Shakhov, Fluid Dyn. **3**, 95 (1972).

[45] X. Nie, X. Shan, and H. Chen, Phys. Rev. E **77**, 035701 (2008).

[46] V. A. Rykov, Fluid Dyn. **10**, 959 (1976).

[47] P. J. Dellar, Prog. Comput. Fluid Dyn., An Inte. J. **8**, 1 (2008).

[48] X. He, X. Shan, and G. D. Doolen, Phys. Rev. E **57**, R13 (1998).

[49] M. Krause, Ph.D thesis, Karlsruhe Institute of Technology, Germany, 2010.

[50] D. Wilde, A. Krämer, K. Küllmer, H. Foysi, and D. Reith, Int. J. Numer. Methods Fluids **90**, 156 (2019).

[51] J. Hesthaven and T. Warburton, J. Comput. Phys. **181**, 186 (2002).

[52] L. Einkemmer and A. Ostermann, Comput. Math. Appli. **69**, 170 (2015).

[53] M. Falcone and R. Ferretti, *Semi-Lagrangian Approximation Schemes for Linear and Hamilton—Jacobi Equations* (Society for Industrial and Applied Mathematics, Philadelphia, 2013).

[54] W. Bangerth, R. Hartmann, and G. Kanschat, ACM Trans. Math. Softw. **33**, 24 (2007).

[55] P. C. Philippi, L. A. Hegele, L. O. E. Dos Santos, and R. Surmas, Phys. Rev. E **73**, 056702 (2006).

[56] A. Krämer, D. Wilde, and M. Bedrunka, *PyTorch-based Lattice Boltzmann Solver Version 0.2.0* (Zenodo, 2020), doi: 10.5281/zenodo.3757641.

[57] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, in *Advances in Neural Information Processing Systems 32*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett (Curran Associates, Red Hook, 2019), pp. 8024–8035.

[58] G. A. Sod, J. Comput. Phys. **27**, 1 (1978).

[59] X. Ji, F. Zhao, W. Shyy, and K. Xu, J. Comput. Phys. **356**, 150 (2018).

[60] M. H. Saadat, F. Bösch, and I. V. Karlin, Phys. Rev. E **101**, 023311 (2020).

[61] P. Lax and X. Liu, SIAM J. Sci. Comput. **19**, 319 (1998).

[62] A. Kurganov and E. Tadmor, Numer. Methods Partial Differ. Equ. **18**, 584 (2002).

[63] O. Inoue and Y. Hattori, J. Fluid Mech. **380**, 81 (1999).

[64] J. D. Sterling and S. Chen, J. Comput. Phys. **123**, 196 (1996).

[65] S. A. Hosseini, C. Coreixas, N. Darabiha, and D. Thévenin, Phys. Rev. E **99**, 063305 (2019).

[66] S. A. Hosseini, C. Coreixas, N. Darabiha, and D. Thévenin, Phys. Rev. E **100**, 063301 (2019).

[67] A. T. White and C. K. Chong, J. Comput. Phys. **230**, 6367 (2011).

[68] M. Watari and M. Tsutahara, Phys. A Stat. Mech. Appl. **364**, 129 (2006).