



Dirichlet and Neumann boundary conditions in a lattice Boltzmann method for elastodynamics

Erik Faust¹ · Alexander Schlüter¹ · Henning Müller² · Felix Steinmetz¹ · Ralf Müller²

Received: 21 July 2022 / Accepted: 6 July 2023 / Published online: 26 July 2023
© The Author(s) 2023

Abstract

Recently, Murthy et al. (Commun Comput Phys 2:23, 2017. <http://dx.doi.org/10.4208/cicp.OA-2016-0259>) and Escande et al. (Lattice Boltzmann method for wave propagation in elastic solids with a regular lattice: theoretical analysis and validation, 2020. [arXiv:2009.06404](https://arxiv.org/abs/2009.06404)) adopted the Lattice Boltzmann Method (LBM) to model the linear elastodynamic behaviour of isotropic solids. The LBM is attractive as an elastodynamic solver because it can be parallelised readily and lends itself to finely discretised simulations of dynamic effects in continua, allowing transient phenomena such as wave propagation to be modeled efficiently. This work proposes simple local boundary rules which approximate the behaviour of Dirichlet and Neumann boundary conditions with an LBM for elastic solids. The boundary rules are shown to be consistent with the target boundary values in the first order. An empirical convergence study is performed for the transient tension loading of a rectangular plate, with a Finite Element (FE) simulation being used as a reference. Additionally, we compare results produced by the LBM for the sudden loading of a stationary crack with an analytical solution from Freund (Dynamic fracture mechanics. Cambridge Monographs on Mechanics. Cambridge University Press, Cambridge, 1990. <https://doi.org/10.1017/CBO9780511546761>).

Keywords Lattice Boltzmann method · Elastodynamics, boundary conditions · Transient solid simulation

1 Introduction

The Lattice Boltzmann Method (LBM) was first established as a viable simulation method in the context of fluid mechanics—in which the LBMs’ distribution functions are most directly subject to physical interpretation, and tech-

nologically relevant transient phenomena are commonplace [2]. However, solid mechanics also features its share of such phenomena—stress wave superposition and dynamic overshoots in the displacement under dynamic loading can result in significantly higher stresses than those predicted by quasistatic calculations—with far-reaching consequences for mining and crash-proofing, for example [4]. Since analytical calculations are often impossible and efficiency is a bottleneck for numerical computation in these cases, researchers have recently begun developing LBM algorithms for the simulation of solids: Marconi et al. [5] approximated crack propagation with one such algorithm, and O’Brien et al. [6] based another on a wave equation for Poisson solids to model wave propagation. More recently, Murthy et al. [1] and Escande et al. [2] extended the solid LBM to the more general class of isotropic, linear elastic solids. In the latter work, a first-order convergence rate was observed [2]. For the static case, second-order convergent methods for the modelling of the bulk and boundary behaviour of linear elastic solids were recently proposed in [7] and [8]. Given further developments, it may become feasible to perform coupled fluid–structure simulations using the LBM as both a fluid and a solid solver.

✉ Erik Faust
erik.faust@mv.rptu.de

Alexander Schlüter
alexander.schlueter@mv.rptu.de

Henning Müller
henning.mueller@tu-darmstadt.de

Felix Steinmetz
felix.steinmetz@mv.rptu.de

Ralf Müller
ralf.mueller@mechanik.tu-darmstadt.de

¹ Lehrstuhl für Technische Mechanik, University of Kaiserslautern-Landau, Postfach 3049, 67653 Kaiserslautern, Rhineland-Palatinate, Germany

² Institut für Mechanik, Technische Universität Darmstadt, Franziska-Braun-Straße 7, 64287 Darmstadt, Hessa, Germany

Thus, the groundwork for LBM-based transient solid simulations has been laid, but much work remains to be done if LBM algorithms are to become viable tools for applied solid mechanics. While efficient, these algorithms are not yet sufficiently stable [2]. Additionally, only rudimentary boundary conditions—for periodic, free, and fixed boundaries—may be modelled in the dynamic case at this point [2].

This work takes aim at the latter obstacle. Prototypes for simple local bounce-back-type boundary rules which may be used to model arbitrarily valued Dirichlet and Neumann boundaries in the solid case are presented. While only first order accuracy can be obtained with these prototypes at present, they highlight a possible pathway to the consistent implementation of boundary conditions for solid simulations with the LBM. Improvements to the stability of the algorithms outlined in [1] and [2], while ultimately necessary for their viability, are beyond the scope of this work.

Before the formulation of the boundary rules are addressed, however, Sect. 2 discusses the reformulation of the underlying solid mechanical equations required by the LBM. The Lamé-Navier equation is restated as a moment chain and boundary conditions are adjusted consistently with this formulation. Section 3 outlines the Lattice Boltzmann Method for solids briefly, and Sect. 4 presents the novel boundary rules. In Sect. 5, we perform an empirical convergence study for a simple example problem, and evaluate the performance of the LBM algorithm and the boundary rules on one analytical and one numerical benchmark problem in Sect. 6. Consistency studies for the LBM algorithm by Murthy et al. [1] and the boundary rules presented here are included in the appendix. While the modified boundary rules are easily generalised to the three-dimensional case, for ease of visualisation and validation we limit the discussion to two-dimensional (plane strain) problems here.

2 A moment chain formulation for linear elastodynamics

For sufficiently small displacements and displacement derivatives (see e.g. [9, p.3]), the Lamé-Navier equation

$$\rho_0 d_t^2 u_\alpha = (\lambda + \mu) d_\alpha d_\beta u_\beta + \mu d_\beta d_\beta u_\alpha + \rho_0 b_\alpha \quad (1)$$

approximates the temporal and spatial evolution of the displacement \mathbf{u} in isotropic, elastic continua [10, p.39]. Here, ρ_0 denotes the (reference) density, λ and μ are the Lamé parameters, and \mathbf{b} is a bulk force per unit mass. The d symbols indicate the material time derivative ($d_t(\cdot) = d(\cdot)/dt$) and the spatial derivative in the reference configuration ($d_\alpha(\cdot) = d(\cdot)/dx_\alpha$), respectively. Furthermore, Einstein summation convention is used. Note that, in contrast to [1], we choose a presentation in the reference configuration—as is common

in solid mechanics—which simplifies derivations in the following. Of course, the current and reference configurations coincide in the small strain limit, but it proves convenient to approach this limit from the reference configuration here. Furthermore, elasticity implies reversibility, which has implications for solid modelling via the LBM. These are discussed in Sect. 3.

For a solution to be determined on a domain Ω_0 with boundary $\partial\Omega_0$, initial conditions as well as boundary conditions (BCs) for the displacement (Dirichlet boundary conditions)

$$u_\alpha(\mathbf{x}, t) = u_\alpha^*(\mathbf{x}, t), \quad \mathbf{x} \in \partial\Omega_{0,u}, \quad (2)$$

with displacement boundary values \mathbf{u}^* and/or for the surface traction (Neumann boundary conditions)

$$\sigma_{\alpha\beta}(\mathbf{x}, t) n_\beta = t_\alpha^*(\mathbf{x}, t), \quad \mathbf{x} \in \partial\Omega_{0,t}, \quad (3)$$

with traction boundary values \mathbf{t}^* and surface normal \mathbf{n} on regions $\partial\Omega_{0,u}$ and $\partial\Omega_{0,t}$, respectively, must further be defined, where $\partial\Omega_{0,u} \cup \partial\Omega_{0,t} = \partial\Omega$ [11, p.7].

While boundary conditions are easily specified, accounting for them in a simulation method is often non-trivial, and the LBM makes no exception [12, p.155]. Moreover, because boundary conditions are central to engineering problems—much of the information relevant to an engineering design process is contained in boundary conditions—their importance is hard to overstate.

Note that, where explicit reference is made to the space- and time dependence of variables, it will be assumed to be of the form (\mathbf{x}, t) . For the sake of conciseness, this reference is omitted unless necessary.

2.1 A moment chain of conservation laws

As shown by [1] and as demonstrated below, the Lamé-Navier equation can be restated as a moment chain. This allows us to use the LBM—which acts as a solver for moment chains, as detailed in appendix A—to model small-strain elastodynamics. Here, in contrast to [1], a presentation in the reference configuration is chosen

$$\begin{aligned} d_t \rho + d_\alpha j_\alpha &= 0, \\ d_t j_\alpha + d_\beta P_{\alpha\beta} &= \rho_0 b_\alpha + \frac{\mu - \lambda}{\rho_0} d_\alpha \rho, \\ d_t P_{\alpha\beta} + d_\gamma \left(\frac{\mu}{\rho_0} (j_\alpha \delta_{\beta\gamma} + j_\beta \delta_{\alpha\gamma} + j_\gamma \delta_{\beta\alpha}) \right) &= 0. \end{aligned} \quad (4)$$

In the above, ρ is a linearised approximation of the density in the current configuration, \mathbf{j} is the linear momentum density, and \mathbf{P} a stress tensor (though not the established Cauchy stress). These variables are outlined in more detail

below, in equations (5) to (8). In keeping with the formulation in the reference configuration, the spatial derivatives d_α are taken with respect to the reference configuration, and the time derivatives d_t denote material derivatives. However, ρ is a linearised approximation of the density in the *current* configuration, defined (in terms of quantities in the reference configuration) as

$$\rho = \rho_0(1 - d_\alpha u_\alpha). \tag{5}$$

Meanwhile, \mathbf{j} is the linear momentum density in the reference configuration,

$$j_\alpha = \rho_0 v_\alpha = \rho_0 d_t u_\alpha. \tag{6}$$

As is readily verified by differentiation, the definition of these first two moments is consistent with the first equation of the moment chain,

$$\begin{aligned} d_\alpha j_\alpha &= d_\alpha(\rho_0 v_\alpha) = d_\alpha(\rho_0 d_t u_\alpha) \\ &= d_t d_\alpha(\rho_0 u_\alpha) = d_t(\rho_0 d_\alpha u_\alpha) = -d_t \rho, \end{aligned}$$

which models mass conservation. \mathbf{P} , meanwhile, denotes a stress tensor which is generally not equivalent to the established Cauchy stress $\boldsymbol{\sigma}$. Via the third equation in (4),

$$-d_t P_{\alpha\beta} = d_\gamma \left(\frac{\mu}{\rho_0} (j_\alpha \delta_{\beta\gamma} + j_\beta \delta_{\alpha\gamma} + j_\gamma \delta_{\beta\alpha}) \right),$$

the stress tensors are related by

$$\begin{aligned} -d_t P_{\alpha\beta} &= d_t \left(\mu (d_\beta u_\alpha + d_\alpha u_\beta + d_\gamma u_\gamma \delta_{\beta\alpha}) \right) \\ &= d_t \left(\lambda d_\gamma u_\gamma \delta_{\alpha\beta} + \mu (d_\beta u_\alpha + d_\alpha u_\beta) \right. \\ &\quad \left. - (\lambda - \mu) d_\gamma u_\gamma \delta_{\alpha\beta} \right) \\ &= d_t \left(\sigma_{\alpha\beta} - (\lambda - \mu) d_\gamma u_\gamma \delta_{\alpha\beta} \right), \end{aligned} \tag{7}$$

and, assuming an adequate initialisation with $P_{\alpha\beta}(\mathbf{x}, 0) = -\sigma_{\alpha\beta}(\mathbf{x}, 0) + (\lambda - \mu) d_\gamma u_\gamma(\mathbf{x}, 0) \delta_{\alpha\beta}$,

$$\begin{aligned} P_{\alpha\beta} &= -\mu (d_\beta u_\alpha + d_\alpha u_\beta + d_\gamma u_\gamma \delta_{\beta\alpha}) \\ &= -\sigma_{\alpha\beta} + (\lambda - \mu) d_\gamma u_\gamma \delta_{\alpha\beta}, \end{aligned} \tag{8}$$

i.e. the stress tensor \mathbf{P} is equivalent to a negated Cauchy stress tensor $\boldsymbol{\sigma}$, with an additional displacement derivative term with coefficient $\lambda - \mu$. The latter disappears for Poisson solids with $\mu = \lambda$ and Poisson’s ratio $\nu = 1/4$, which is why \mathbf{P} is termed the *Poisson stress tensor* in the following.

With (5), the gradient of ρ can be written as

$$d_\alpha \rho = d_\alpha(\rho_0(1 - d_\beta u_\beta)) = -\rho_0 d_\alpha d_\beta u_\beta. \tag{9}$$

Note that ρ , as defined in (5), can be viewed as a scalar variable encoding information about dilatational strains. This allows information about these dilatational strains to be introduced into the linear momentum balance in (4) via the gradient of ρ , without the displacement \mathbf{u} being required.

Finally, substitution of (6), (8), and (9) into the second equation of the moment chain (4)—which accounts for the linear momentum balance

$$\begin{aligned} d_t j_\alpha + d_\beta P_{\alpha\beta} &= \rho_0 b_\alpha + \frac{\mu - \lambda}{\rho_0} d_\alpha \rho, \\ \Leftrightarrow d_t(\rho_0 d_t u_\alpha) - \mu d_\beta (d_\beta u_\alpha + d_\alpha u_\beta + d_\gamma u_\gamma \delta_{\beta\alpha}) \\ &= \rho_0 b_\alpha + \frac{\mu - \lambda}{\rho_0} (-\rho_0 d_\alpha d_\beta u_\beta), \\ \Leftrightarrow \rho_0 d_t^2 u_\alpha - \mu (d_\beta d_\beta u_\alpha + d_\alpha d_\beta u_\beta + d_\alpha d_\gamma u_\gamma) \\ &= \rho_0 b_\alpha - (\mu - \lambda) d_\alpha d_\beta u_\beta, \\ \Leftrightarrow \rho_0 d_t^2 u_\alpha &= (\lambda + \mu) d_\alpha d_\beta u_\beta + \mu d_\beta d_\beta u_\alpha + \rho_0 b_\alpha, \end{aligned}$$

yields the Lamé-Navier equation

$$\rho_0 d_t^2 u_\alpha = (\lambda + \mu) d_\alpha d_\beta u_\beta + \mu d_\beta d_\beta u_\alpha + \rho_0 b_\alpha.$$

The presentation in the reference configuration simplifies the derivations (and the analysis in the following) somewhat: in contrast to the calculations in [1], there are no convective terms which need to be eliminated with small-strain assumptions. Smallness assumptions are only required in the formulation of the material law, which neglects material and geometric nonlinearities at the outset.

The first equation of the moment chain (4) is a mass conservation equation which accounts for the evolution of the linearised density in the current configuration, ρ . As discussed in the context of (9), ρ is used to embed information about dilatational strains without reference to the displacements. The second equation accounts for the linear momentum balance, with the contribution due to the Cauchy stress $\boldsymbol{\sigma}$ partially modeled via the Poisson stress \mathbf{P} and partially via the gradient of ρ . ρ introduces information about the non-Poisson contribution to the dilatational strain into the linear momentum balance. The final equation determines the evolution of the Poisson stress tensor \mathbf{P} . Together, these equations were shown to be equivalent to the Lamé-Navier equation, which is commonly used to model the dynamic behaviour of isotropic solids subject to small strains. As the LBM acts as a solver for moment chains like (4), it may be used to tackle isotropic, small-displacement elastodynamics.

The introduction of the Poisson tensor \mathbf{P} as in [1] is necessary as the LBM can not account for the Cauchy stress $\boldsymbol{\sigma}$ easily: the reasons for this are discussed in Sect. 3.

2.2 Reformulating the boundary conditions

In accordance with the reformulation of the Lamé-Navier equation as a moment chain in ρ , \mathbf{j} , and \mathbf{P} , boundary conditions (BCs) must of course also be rephrased in terms of these variables. For Dirichlet boundary conditions, this is comparatively straightforward: with the definition of the linear momentum density $\mathbf{j} = \rho_0 d_t \mathbf{u}$, boundary values in the linear momentum density \mathbf{j}^* can be obtained from boundary values in the displacement

$$j_\alpha(\mathbf{x}, t) = j_\alpha^*(\mathbf{x}, t) = \rho_0 d_t u_\alpha^*(\mathbf{x}, t), \quad \mathbf{x} \in \partial\Omega_{0,u}. \quad (10)$$

With (8), boundary values in the Poisson stress tensor \mathbf{P}^* could be computed from boundary values in the Cauchy stress tensor $\boldsymbol{\sigma}^*$ and from values of the divergence of the displacement at the boundary, via

$$\begin{aligned} P_{\alpha\beta}(\mathbf{x}, t) &= P_{\alpha\beta}^*(\mathbf{x}, t) \\ &= -\sigma_{\alpha\beta}^*(\mathbf{x}, t) + (\lambda - \mu) d_\gamma u_\gamma(\mathbf{x}, t) \delta_{\alpha\beta}, \quad \mathbf{x} \in \partial\Omega_{0,t}. \end{aligned}$$

However, Neumann boundary conditions enforce boundary values in the Cauchy stress vector $\mathbf{t}^* = \boldsymbol{\sigma}^* \mathbf{n}$ rather than the Cauchy stress tensor $\boldsymbol{\sigma}^*$ directly. Furthermore, the displacement \mathbf{u} does not appear in the moment chain (4), and is thus not directly available in the LBM algorithm.

The latter of these issues could be circumvented by integrating the linear momentum density $\mathbf{j} = \rho_0 d_t \mathbf{u}$ to obtain the displacement \mathbf{u} . However, the linearised density ρ embeds information about the displacement divergence via (5), which can be exploited

$$d_\alpha u_\alpha = \frac{\rho_0 - \rho}{\rho_0}, \quad (11)$$

to compute $d_\alpha u_\alpha$ more efficiently.

Meanwhile, Neumann boundary conditions can be transformed into a coordinate system normal to the boundary with a rotation matrix \mathbf{T}

$$t_\alpha^{*n}(\mathbf{x}, t) = T_{\beta\alpha} t_\beta^*(\mathbf{x}, t) \quad \mathbf{x} \in \partial\Omega_{0,t}.$$

The entries of the orthonormal transformation tensor \mathbf{T} are simply given by the normal and tangential vectors \mathbf{e}^n and \mathbf{e}^t of the boundary, i.e. $T_{1\alpha} = e_\alpha^n$ and $T_{2\alpha} = e_\alpha^t$ for the two-dimensional case [13, p.28].

For the two-dimensional case, (3) becomes, in this normal coordinate system

$$\begin{pmatrix} \sigma_{nn}^n & \sigma_{nt}^n \\ \sigma_{nt}^n & \sigma_{tt}^n \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} t_n^{*n} \\ t_t^{*n} \end{pmatrix}.$$

As is apparent, the first column of the Cauchy stress tensor $\boldsymbol{\sigma}^n$ is fully determined by the Cauchy stress vector that is

prescribed via the boundary condition, i.e.

$$\begin{aligned} \sigma_{nn}^n &= t_n^{*n} := \sigma_{nn}^{*n}, \\ \sigma_{nt}^n &= t_t^{*n} := \sigma_{nt}^{*n}, \end{aligned}$$

meaning that, with the symmetry of the Cauchy tensor, the Neumann boundary condition determines all entries of $\boldsymbol{\sigma}^n$ save for σ_{tt}^n . This remaining entry should remain unaffected by the Neumann boundary condition. In the boundary rules suggested in Sect. 4, we extrapolate from (known) values of $\sigma_{tt}^n(\mathbf{x}, t)$, $\mathbf{x} \in \Omega_0 \setminus \partial\Omega_0$ within the material domain to the boundary to obtain an approximation of $\sigma_{tt}^n(\mathbf{x}, t)$, $\mathbf{x} \in \partial\Omega_{0,t}$ —see (23) for details. Thus, we obtain boundary values for the Cauchy stress $\boldsymbol{\sigma}^n$ in the normal coordinate system. A subsequent inverse transformation yields boundary values for $\boldsymbol{\sigma}^*$ in the original (global) coordinate system,

$$\sigma_{\alpha\beta}^*(\mathbf{x}, t) = T_{\alpha\gamma} \sigma_{\gamma\zeta}^{*n}(\mathbf{x}, t) T_{\beta\zeta} \quad \mathbf{x} \in \partial\Omega_{0,t}. \quad (12)$$

Using (8), we finally obtain the desired boundary values for the Poisson stress tensor \mathbf{P}

$$\begin{aligned} P_{\alpha\beta}(\mathbf{x}, t) &= P_{\alpha\beta}^*(\mathbf{x}, t) \\ &= -\sigma_{\alpha\beta}^*(\mathbf{x}, t) + (\lambda - \mu) d_\gamma u_\gamma(\mathbf{x}, t) \delta_{\alpha\beta}, \quad \mathbf{x} \in \partial\Omega_{0,t}. \end{aligned} \quad (13)$$

where $\boldsymbol{\sigma}^*$ is determined via (12) and the displacement divergence may be obtained from (11). With (10) and (13), Dirichlet and Neumann boundary conditions for elastodynamic problems described by the Lamé-Navier equation can now be stated consistently with the moment chain form in (4). This completes the reformulation required to make elastodynamic problems amenable to solutions with the LBM.

3 The solid LBM in a nutshell

For the sake of compactness, only a brief overview of the LBMs key features will be given here. An in-depth discussion can be found in [12]. In keeping with [14], we treat the Lattice Boltzmann algorithm as a numerical solver for PDEs described by moment chains, independently from any associations with kinetic theory.¹

The algorithm operates on a lattice L consisting of regularly spaced lattice sites $\mathbf{x} \in L \subset \Omega$ (the dark grey points in Fig. 1) which are in turn connected by lattice links (the light

¹ In this view, the distribution functions \bar{f}_i with which the LBM works are merely supplementary variables containing information about the tensors to be modelled (the linearised density ρ in the current configuration, the linear momentum density \mathbf{j} , and the Poisson stress \mathbf{P} , in this case).

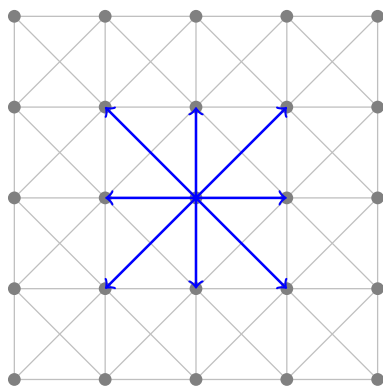


Fig. 1 (Part of) a D2Q9 lattice for the LBM

grey lines in the same diagram) [12, p.94]. Furthermore, several lattice velocity vectors $c_i, i \in 0, \dots, q - 1$ (indicated by the blue arrows in Fig. 1) cover the distance between any lattice site x and its neighbours $x + c_i \Delta t$ along a lattice link i in one time step Δt [12, p.94]. The simplicity of this spatial discretisation makes for easy pre- and post-processing.

Generally, LBM discretisations are classed by the dimension d and the number of lattice velocities q , as DdQq lattices. For ease of visualisation and validation, we only consider D2Q9 lattices - exemplified in Fig. 1 - here. Note that among the $q = 9$ lattice velocities, there is a zero velocity $c_0 = \mathbf{0}$.

The LBM encapsulates information about the tensors to be modelled (the linearised density in the current configuration ρ , the linear momentum density j , and the Poisson stress P in our case) in distribution functions. On each site of the lattice, a distribution function vector with one entry \bar{f}_i per lattice velocity c_i is introduced [12, p.63]. Here and in the following, we utilise distribution functions resulting from the second-order accurate discretisation by He et al. [15], which is denoted by the overbar $\bar{\cdot}$.

In each iteration, the distribution functions \bar{f}_i at each lattice site are locally relaxed toward the value of the equilibrium distribution function f_i^{eq} [12, p.64], and a contribution due to a source term is added [12, p.239]. The choice of equilibrium distribution function determines the PDE modelled by the LBM algorithm - as shown in appendix A, the equilibrium distribution function by Murthy et al. [1]

$$f_i^{eq} = w_i \left(\rho + \frac{1}{c_s^2} c_{i\alpha} j_\alpha + \frac{1}{2c_s^4} (P_{\alpha\beta} - \rho c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} - c_s^2 \delta_{\alpha\beta}) \right) \quad (14)$$

may be used to recover the Lamé-Navier equation. Here, $c_s = \sqrt{\mu/\rho_0}$ denotes the speed of shear waves, and w_i is a lattice weight associated with lattice link i [2].

In the solid LBM algorithm by Murthy et al. [1] and Escande et al. [2], a BGK collision operator with relaxation

time $\bar{\tau}$ is utilised. Together with the contribution due to the source term ψ_i , this results in the following expression for the post-collision (i.e., post-relaxation) distribution functions [12, p.239]

$$\bar{f}_i^{col} = \bar{f}_i - \frac{\Delta t}{\bar{\tau}} (\bar{f}_i - f_i^{eq}) + \Delta t \left(1 - \frac{\Delta t}{2\bar{\tau}} \right) \psi_i. \quad (15)$$

The locality of this collision step makes parallelisation relatively easy [12, p.579], contributing to the efficiency which makes the LBM attractive as a PDE solver.

The source term

$$\psi_i = \frac{1}{c_s^2} w_i c_{i\alpha} S_\alpha^{FD}, \quad (16)$$

meanwhile, is used to model bulk forces, where

$$S_\alpha^{FD} = \rho_0 b_\alpha + \frac{\mu - \lambda}{\rho_0} d_\alpha^{FD} \rho, \quad (17)$$

accounts both for the volumetric loads b and the part of the material law not contained in the Poisson stress tensor [2] - recall the moment chain (4). The finite difference stencil used to evaluate the gradient of ρ - denoted by $d_\alpha^{FD} \rho$ above - must at least be second-order accurate for the LB algorithm to be second-order consistent with the target moment chain in the bulk, as demonstrated in appendix A. The source term is crucial for P-wave modelling in non-Poisson solids: the Lamé constant λ does not appear in the equilibrium distribution function by Murthy et al. [1]; the source term ensures that the correct dilatational wave speed $c_d = \sqrt{(\lambda + 2\mu)/\rho_0}$ is recovered.

The post-collision distribution functions are subsequently propagated to neighbouring lattice sites along the associated lattice velocities c_i in the streaming step [12, p.66]

$$\bar{f}_i(\mathbf{x} + c_i \Delta t, t + \Delta t) = \bar{f}_i^{col}. \quad (18)$$

Together, the collision and streaming steps define the Lattice Boltzmann equation

$$\bar{f}_i(\mathbf{x} + c_i \Delta t, t + \Delta t) = \bar{f}_i - \frac{\Delta t}{\bar{\tau}} (\bar{f}_i - f_i^{eq}) + \Delta t \left(1 - \frac{\Delta t}{2\bar{\tau}} \right) \psi_i, \quad (19)$$

which describes the evolution of the distribution functions on the lattice [12, p.239]. From the distribution functions, the desired tensors may finally be post-processed as moments

$$\rho = \sum_i \bar{f}_i, \\ j_\alpha = \sum_i \bar{f}_i c_{i\alpha} + \frac{1}{2} \Delta t S_\alpha^{FD},$$

$$P_{\alpha\beta} = \sum_i \bar{f}_i c_{i\alpha} c_{i\beta}. \tag{20}$$

Appendix A uses the method suggested by Farag et al. [14] to demonstrate that the resulting algorithm is consistent with the target equations (4) up to the second order. Note that, to obtain second-order consistency, the bulk force term S^{FD} in (20) must be evaluated at time $t + \Delta t$ for an iteration starting at time t . Despite ρ influencing S^{FD} via (17), this is unproblematic. $\rho(\mathbf{x}, t + \Delta t)$ may be computed via the first equation in (4), and subsequently used to compute $S^{FD}(\mathbf{x}, t + \Delta t)$ for use in the second moment.

In contrast to the fluid LBM (see e.g. [14]), the solid LBM algorithm outlined above models the stress tensor σ (see equation (8)) entirely via the equilibrium moments. This is because the momentum transfer described by the Cauchy stress σ is non-dissipative and reversible in elastic solids, in contrast to the Navier–Stokes fluid case, where viscous terms play a role [12, p.6].

On account of this reversibility, the non-dissipative, non-viscous limit $\bar{\tau} \rightarrow \Delta t/2_+$ (from above) of the LBM (see e.g. [12, p.65,101]) is of interest for linear elastic solid modelling. As simulations with $\bar{\tau} = \Delta t/2$ (and no damping) are unstable, $\bar{\tau} = 0.55\Delta t$ is chosen in keeping with [2]. In contrast to the fluid case, $\bar{\tau}$ is treated as a numerical parameter and does not determine the evolution of the stress tensor.

The introduction of the Poisson tensor \mathbf{P} as in [1] is necessary as the LBM can not account for the Cauchy stress σ directly without significant structural changes: the terms of the third-order moment \mathbf{Q} (which determines the evolution of the second moment) that are affected by the isotropy defect in a D2Q9 lattice are inconsistent with an evolution equation for the Cauchy stress, i.e.

$$\begin{aligned} Q_{AAA} &= \sum_i c_{iA} c_{iA} c_{iA} f_i^{eq} = \sum_i \left(\frac{\Delta x}{\Delta t}\right)^2 c_{iA} f_i^{eq} \\ &= 3c_s^2 \bar{j}_A = 3\frac{\mu}{\rho_0} j_A \neq \frac{\lambda + 2\mu}{\rho_0} j_A, \end{aligned}$$

where summation over the latin index A is not implied. However, $Q_{AAA} = j_A(\lambda + 2\mu)/\rho_0$ must hold for $d_t \sigma_{\alpha\beta} - d_\gamma Q_{\alpha\beta\gamma} = 0$ to hold for all α and β . Therefore, σ can not be modelled directly as a moment, unless modifications are made to the equilibrium distribution function and the wave speed c_s .

Note that the displacement \mathbf{u} is not required at any point in the LBM algorithm outlined above (it is also not required at any stage in the boundary rules discussed below). If displacements are required for postprocessing, they may be computed from the linear momentum density \mathbf{j} by numerical integration, e.g.

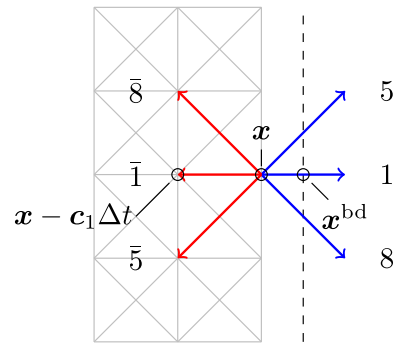


Fig. 2 Lattice-conforming boundary (dotted line) with lattice velocities leading out of the domain (blue) and lattice velocities for which distribution functions are missing (red) for a boundary lattice site at \mathbf{x}

$$\begin{aligned} u_\alpha(\mathbf{x}, t + \Delta t) &= \\ u_\alpha(\mathbf{x}, t) &+ \frac{1}{\rho_0} j_\alpha(\mathbf{x}, t + \Delta t) \Delta t + \mathcal{O}(\Delta t^2). \end{aligned}$$

Of course, higher-order integration is also possible.

The operations constituting the LB algorithm—the local relaxation in the collision step (15), the simple memory swap in the streaming step (18), and the local array operations in the moment computation (20) can be evaluated very efficiently, which makes the LBM attractive from a computational point of view.

4 Local boundary rules for the solid LBM

While the LBE accounts for the evolution of the distribution functions in the interior of the lattice, the collision and streaming steps leave some distribution functions in the vicinity of the boundary $\partial\Omega_0 \subset \Omega_0$ undefined. As illustrated in Fig. 2, these need to be ‘streamed’ from across the boundary to each boundary lattice site $\mathbf{x} \in \partial L$, i.e. each lattice site with lattice links crossing the boundary [12, p.165]. For convenience, let the set of boundary lattice sites $\partial L \subset L$ be divided into ∂L_u and ∂L_t depending on whether the nearby boundary is subject to Dirichlet or Neumann boundary values, respectively.²

Specifying the missing distribution functions in accordance with the boundary conditions of the underlying PDE is a non-trivial task: while boundary conditions are formulated in terms of the tensors appearing in the moment chain $(\rho, \mathbf{j}, \mathbf{P})$, a boundary rule specification with respect to \bar{f}_i is sought [12, p.155]. Furthermore, the number of unknown distribution functions \bar{f}_i at some $\mathbf{x} \in \partial L$ is sometimes higher than the number of boundary condition equations available to specify

² Distinctions between boundaries can also be made on a link-by-link basis (as is done in the numerical experiments in Sect. 6), but this text uses the simpler site-by-site approach to simplify explanations.

them, leaving boundary rules for the LBM under-determined. An example in the D2Q9 case are Dirichlet boundary conditions (10) at a straight boundary as in Fig. 2, with three unknown distribution functions but two entries of \mathbf{j}^* determined as boundary values. Conversely, boundary rules might also be over-determined, as in the case of a concave boundary with, for example, only one missing distribution function.

As a first step, Escande et al. [2] appropriated the popular bounce-back [12, p.175] and anti-bounce-back [12, p.200] rules from the fluid case, to model lattice-conforming fixed and free boundaries respectively. In the scope of this work, these rules are extended, to approximate the behaviour in the vicinity of boundaries with arbitrary Dirichlet and Neumann boundary conditions.

The modified bounce-back rule

$$\bar{f}_i(\mathbf{x}, t + \Delta t) = \bar{f}_i^{\text{col}} - \frac{2}{c_s^2} w_i c_{i\alpha} j_\alpha^*, \tag{21}$$

first reverses the direction of motion for the post-collision distribution functions $\bar{f}_i^{\text{col}}(\mathbf{x}, t)$ which would be streamed out of the bulk: the missing distribution functions $\bar{f}_i(\mathbf{x}, t + \Delta t)$ for directions \bar{i} are set to the values of the post-collision distribution functions $\bar{f}_i^{\text{col}}(\mathbf{x}, t)$ associated with the opposite direction i . Then, a term depending on the linear momentum density boundary values \mathbf{j}^* is subtracted, and the distribution functions are sent back into the interior of the lattice, in direction \bar{i} . The distribution function $\bar{f}_5(\mathbf{x}, t + \Delta t) = \bar{f}_7(\mathbf{x}, t + \Delta t)$ in Fig. 2, for example, is set to the value of the post-collision distribution function $\bar{f}_5^{\text{col}}(\mathbf{x}, t)$, minus the boundary value term $\frac{2}{c_s^2} w_5 c_{5\alpha} j_\alpha^*$.

The required boundary values in the linear momentum density \mathbf{j}^* can be obtained from Dirichlet boundary values in the displacement \mathbf{u}^* via (10). As these are typically known a priori, the necessary differentiation can be performed analytically.

In appendices B and C, the simple bounce-back rule suggested by Escande et al. [2] and extended here is shown to replicate the target behaviour at the boundary (10) with first-order accuracy.

The modified anti-bounce-back boundary rule - used here to model Neumann boundaries - operates similarly. The sign of the post-collision distribution function $\bar{f}_i^{\text{col}}(\mathbf{x}, t)$ at $\mathbf{x} \in \partial L_t$ is first inverted and a contribution depending on the linearised density ρ^* in the current configuration and the Poisson stress \mathbf{P}^* at the boundary is added, before the distribution functions are sent back into the direction whence they came

$$\begin{aligned} \bar{f}_i(\mathbf{x}, t + \Delta t) = & -\bar{f}_i^{\text{col}} + 2w_i \left(\rho^* \right. \\ & \left. + \frac{1}{2c_s^4} (P_{\alpha\beta}^* - \rho^* c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} - c_s^2 \delta_{\alpha\beta}) \right). \end{aligned} \tag{22}$$

While Escande et al. [2] extrapolate ρ and \mathbf{P} normal to the boundary, we use (13) to determine boundary values in the Poisson stress tensor \mathbf{P}^* from Neumann boundary conditions on the Cauchy traction vector \mathbf{t}^* . The entry of the Cauchy stress at the boundary σ_{tt}^{*n} which is left undetermined by the boundary conditions in a normal coordinate system, is extrapolated along the current lattice link i to $\mathbf{x}^{\text{bd}} = \mathbf{x} + \frac{1}{2} \mathbf{c}_i \Delta t$,

$$\begin{aligned} \sigma_{tt}^{*n} = & \sigma_{tt}^n(\mathbf{x}^{\text{bd}}, t) \\ = & \frac{1}{2} \left(3\sigma_{tt}^n(\mathbf{x}, t) - \sigma_{tt}^n(\mathbf{x} - \mathbf{c}_i \Delta t, t) \right) + \mathcal{O}(\Delta t^2). \end{aligned} \tag{23}$$

Similarly, the linearised density in the current configuration at the boundary, ρ^* —for which there are no boundary values—is similarly extrapolated to the boundary along the current lattice link i

$$\begin{aligned} \rho^* = & \rho(\mathbf{x}^{\text{bd}}, t) \\ = & \frac{1}{2} \left(3\rho(\mathbf{x}, t) - \rho(\mathbf{x} - \mathbf{c}_i \Delta t, t) \right) + \mathcal{O}(\Delta t^2). \end{aligned}$$

Figure 2 indicates, using $i = 1$ as an example, how the location of the boundary lattice site \mathbf{x} , neighbouring lattice site $\mathbf{x} - \mathbf{c}_i \Delta t$, and boundary site $\mathbf{x}^{\text{bd}} = \mathbf{x} + \frac{1}{2} \mathbf{c}_i \Delta t$ are defined in the lattice-conforming case.

The consistency study in appendices B and C indicates that the modified anti-bounce-back boundary rule (22) reproduces the desired boundary values in the Poisson stress \mathbf{P}^* , and thus, by (12) and (13), the desired Cauchy traction \mathbf{t}^* with first-order accuracy.

5 Empirical convergence study

As demonstrated in appendix A, the LBM by Murthy et al. [1] should act as a second-order solver for solid-mechanical problems written in the moment chain form discussed in Sect. 2. The analysis in appendix C, meanwhile, suggests that the simple boundary rules suggested in Sect. 4 should enforce Dirichlet and Neumann boundary values with first-order accuracy, respectively.

In this section, an empirical mesh refinement study is performed to investigate the convergence rate of the LBM algorithm outlined above on a simple example problem.

To this end, we consider the material domain and load case illustrated in Fig. 3: the rectangular region with side length l is subjected to a plane strain state, the upper and lower boundaries being loaded with a traction of \mathbf{t}^* in their respective normal directions. Note that the variables used here are non-dimensionalised, with $u_{\text{ref}} = 10^{-3}l$ (arbitrary units). The traction is increased linearly from $\mathbf{t}^* = 0.0\mu u_{\text{ref}}/l$ at time $t = 0.0l/c_s$ to $\mathbf{t}^* \approx 4.385\mu u_{\text{ref}}/l$ at $t = 1.15l/c_s$, and

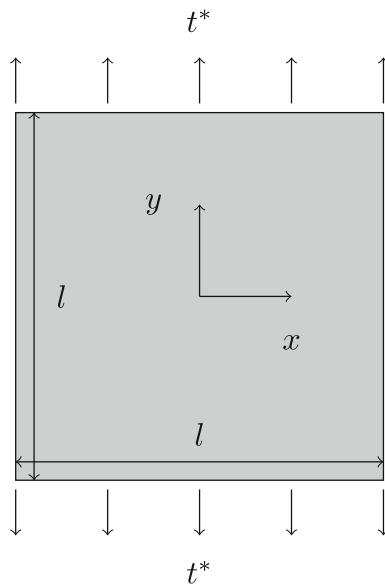


Fig. 3 Square domain subjected to an in-plane tension load

subsequently held at this level until the end of the simulation at $t = 2.281l/c_s$. The left and right sides of the domain are stress-free, i.e. subject to homogeneous Neumann boundary conditions. The material to be simulated is characterised by $c_d^2/c_s^2 \approx 2.769$ and $\nu \approx 0.2174$, which is typical for certain types of concretes and glasses [16].

Lattice Boltzmann simulations are run for a range of lattice spacings $\Delta x \in [0.003125, 0.05]l$, i.e. $\Delta x \in [1/320, 1/20]l$. In each case, the time step is determined to satisfy the lattice isotropy conditions via [12, p.64]

$$\frac{1}{\sqrt{3}} \frac{\Delta x}{\Delta t} = c_s.$$

A relaxation time of $\bar{\tau} = 0.55\Delta t$ is chosen, as in [2]. In the absence of analytical solutions, the convergence study is performed relative to a solution of a high-fidelity FEM simulation for the resulting displacement fields. The FE analyses were run using the FEAP FE program (see e.g. [17]). For the FEM simulations, we use bilinear shape functions, a regular mesh consisting of quadratric elements, an element edge length of $\Delta x^{\text{FEM}} \approx 0.001563l$, and an explicit central difference integrator with time step $\Delta t^{\text{FEM}} \approx 3.956 \times 10^{-4}l/c_s$. Convergence rates for elastodynamic FEM simulation can be quite low—examples with linear or superlinear convergence, even for geometrically simple problems, can be found, see e.g. [18]. However, fourth-order convergence has been observed for the FEM in similar well-behaved, regularly discretised elastodynamic problems [18]. Furthermore, the FE solution does not change appreciably, relative to the error between LBM and FEM solutions, upon further refinement. We therefore assume that the displacement field computed

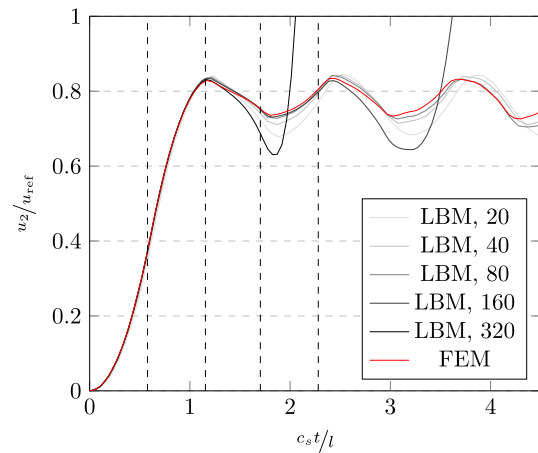


Fig. 4 x_2 -components of dimensionless displacements at the top right corner post-processed from LBM (shades of gray) and FE (red) simulations of the square plane strain domain used for the convergence study. Simulations with 320, 160, 80, 40, and 20 lattice sites along an edge were used for the LBM, while the higher-fidelity FEM reference simulation is shown for comparison. The time values for which convergence studies were performed are shown as dashed lines

by the FEM with this discretisation is sufficiently indistinct from the exact solution for use in the convergence study.

Figure 4 illustrates the tensile response of the plane strain domain as predicted by the FEM and the LBM: the x_2 -displacement values at the top right corner ($x_1 = x_2 = 0.5l$) are postprocessed from simulation runs with the LBM at different discretisations, as well as from the higher-fidelity reference FEM solution.

In both simulations, the displacement first increases gradually, then peaks (with the LBM yielding a slightly higher peak value), and then oscillates around the static solution. As dissipation is not considered in linear elastic theory, the oscillation would continue indefinitely if there were no artificial numerical damping. This behaviour is apparent in the first three dynamic overshoots in the displacement until $t \approx 4l/c_s$. Both simulations are initially in acceptable agreement, with a disparity that decreases as the lattice used for the LB simulation runs is refined. At some point, however, instabilities in the LB algorithm (which were already noted by [2]) cause the solution to diverge. For the higher-fidelity simulations, instabilities appear earlier in time, but after a similar number of time-steps. In appendix D, a series of contour plots of the displacement magnitude are shown to highlight how instabilities arise near the corners of the simulated domain. While this instability ultimately needs to be addressed in the further development of the method, this is beyond the scope of the current work.

As the oscillations exhibited by the plane strain domain continue indefinitely with little change to their amplitude or frequency, almost all of the mechanical behaviour of interest is captured by simulations of the first transient overshoot. In

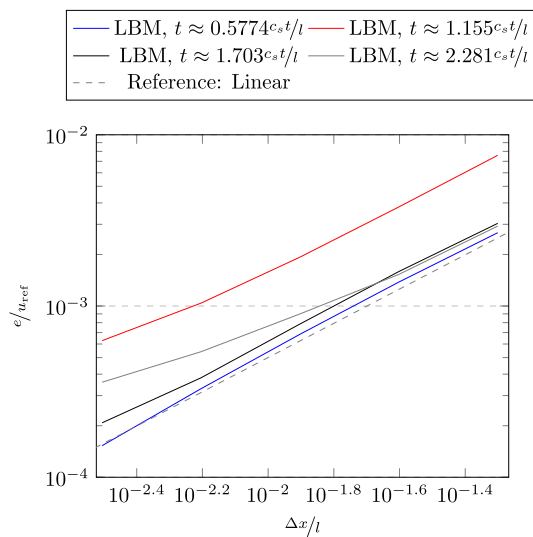


Fig. 5 Convergence study for several times t during the LBM simulation of a quadratic domain. The Euclidean norm of the error over the displacement field e/u_{ref} (non-dimensionalised and normalised with the number of evaluation points) is plotted over the non-dimensionalised lattice spacing $\Delta x/l$

the following, attention will thus be confined to this interval, $t \in [0, 2.5]l/c_s$.

A mean error with respect to the reference solution is computed over the displacement field obtained via the LBM for each discretisation, at times $t \approx \{0.5775, 1.155, 1.703, 2.281\}l/c_s$ (noted as dashed lines in Fig. 4). The resulting convergence plot in Fig. 5 shows that the LBM solution converges linearly toward the reference FE solution. Eventually, instabilities cause a reduction in the rate of convergence, as already noted by [2]. This indicates that the LBM by Murthy et al. [1] and the boundary rules presented here capture the transient behaviour of the Lamé-Navier equation and the chosen boundary values as suggested in the consistency studies in appendices A and C, though improvements must be made to the stability of the method. Furthermore, higher-order convergence would be desirable to achieve higher accuracy with a coarser discretisation. To achieve this with bounce-back-type boundary rules, the error terms in equations (C20) and (C21) in appendix C would have to be eliminated. However, the grid refinement study and the consistency studies in appendices A and C indicate that solid-mechanical boundary rules can be modeled consistently via bounce-back-type boundary rules. Furthermore, the additional linear momentum density source suggested by Murthy et al. - see (4) - seems to model the non-Poisson part of the material law successfully in this application. Despite the linear convergence rate and the eventual instabilities, the relative error of the LBM relative to the FEM solution can be reduced below 0.1% with moderate mesh refinement.

The LBM simulations for the convergence study outlined above were run using an in-house Python code. These simulations runs result in a slightly lower runtime than explicit FEAP simulations with an equivalent discretisation: an FEM simulation run with $\Delta x \approx 6.250 \times 10^3 l$ took 49 seconds on a Intel (R) Core(TM) i5-8500 CPU @ 3.00GHz, while the equivalent LBM simulation required only 24 seconds on the same hardware, with the same time step being used for both. A detailed performance benchmarking study would be necessary to investigate relative runtime advantages further, but the rough preliminary comparison is encouraging.

6 Numerical and analytical examples

This section considers two further benchmark examples to illustrate how the LBM can be used to approach elastodynamic problems. A dynamically tension-loaded rectangular domain with a circular hole is modeled first. In the latter example, results produced by the LBM are compared against an analytical solution for the sudden loading of a finite crack. In both cases, moderately coarse discretisations are used to investigate whether the LB algorithm and the boundary rules suggested above produce acceptable results at low resolution – and thus at low computational cost.

6.1 Tension loading of a rectangular domain with a circular hole

Firstly, we consider a rectangular domain (side length l) with a central circular hole (radius $r = 0.133l$), as illustrated in Fig. 6. This domain is subjected to a plane strain state, the upper and lower boundaries being loaded with a traction of t^* in their respective normal directions. Note that at the boundary of the circular hole, the condition for the determinacy of Neumann boundary conditions—discussed at the end of appendix C—is not met everywhere. The example serves to broadly illustrate the behaviour of the solid LBM in this scenario. Again, all variables used here are non-dimensionalised, with $u_{\text{ref}} = 10^{-3}l$ (arbitrary units).

The traction is increased linearly from $t^* = 0.0\mu u_{\text{ref}}/l$ at time $t = 0.0l/c_s$ to $t^* = 5.0\mu u_{\text{ref}}/l$ at $t = 1.0l/c_s$, and subsequently held at this level until the end of the simulation at $t = 2.0l/c_s$. The left and right sides of the rectangle as well as the circular hole are stress-free, i.e. subject to homogeneous Neumann boundary conditions.

To describe the simulated material, we use both parameters corresponding to a Poisson solid ($c_d^2/c_s^2 = 3.0$, $\nu = 0.25$) and a non-Poisson solid ($c_d^2/c_s^2 = 2.8$, $\nu = 0.2$). The former is often encountered in seismological wave modelling [6] and the latter is typical for certain concretes and glasses [16].

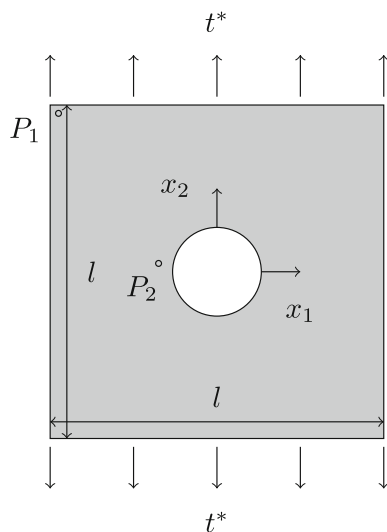


Fig. 6 Square domain with circular hole subjected to an in-plane tension load

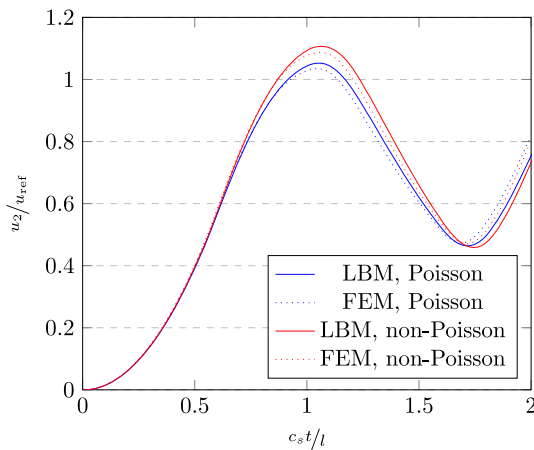


Fig. 7 x_2 -components of dimensionless displacements at P_1 post-processed from LBM (blue) and FE (red) simulations of a Poisson- and non-Poisson solid under tension loading

LBM and FEM simulations with lattice spacing/element size $\Delta x = 0.0125l$ and time step $\Delta t \approx 7.217 \times 10^{-3}l/c_s$ are compared qualitatively.

After the simulation run, the x_2 -component u_2 of the displacement at the point $P_1 = (-0.5l + \Delta x/2, 0.5l - \Delta x/2)$ in the top-left corner and the x_1 -component u_1 of the displacement at the point $P_2 = (-0.15l + \Delta x/2, \Delta x/2)$ near the hole are post-processed from the linear momentum density \mathbf{j} . These points are chosen as they yield relatively high displacements that vary comparatively strongly in time.

Figure 7 plots the vertical displacement (u_2) predicted at P_1 by the LBM and the FEM for the Poisson case in blue, and the non-Poisson case in red. The reference results produced by the FEM are displayed as dotted lines, while the continuous curves indicate the LBM's prognosis. As is apparent in the figure, the LBM and the FEM are in acceptable agreement

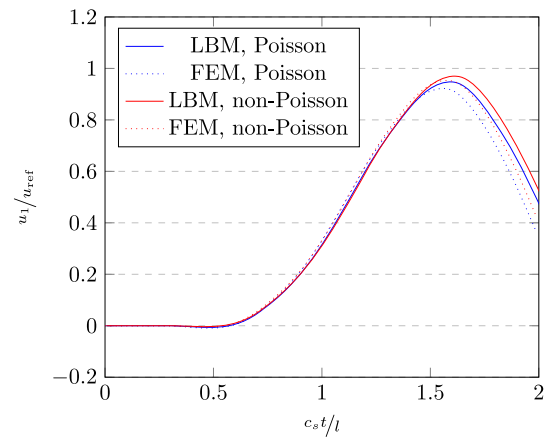


Fig. 8 x_1 -components of dimensionless displacements at P_2 post-processed from LBM (blue) and FE (red) simulations of a Poisson- and a non-Poisson solid under tension loading

throughout the simulation, in view of the moderate discretisation.

Similarly, Fig. 8 visualises the change of horizontal displacement (u_1) at P_2 computed via the LBM and the FEM. As above, the displacements are in decent agreement throughout the simulation. Interestingly, the LBM seems to predict a slightly lower and later peak in the displacement u_1 .

The displacements shown in Figs. 7 and 8 for Poisson and non-Poisson material parameters reveal a slight difference in the displacement amplitude due to the differing ratio of shear and volumetric stiffnesses. As is apparent, both the LBM and the FEM predict similar magnitudes for this effect.

Finally, the deformation predicted by the LBM and FEM simulations at $t = 1.5l/c_s$ is visualised in Fig. 9, for the non-Poisson solid with $c_d^2/c_s^2 = 2.8$. A contour plot for the vertical displacements u_2 is superimposed on the warped surface indicating the current configuration as determined by the FEM. The position of lattice sites in the current configuration is indicated by black dots. The agreement in the computed displacements throughout the simulated domain is apparent.

The agreement between the two simulation methods in this example is generally encouraging. The LBM and the FEM produce a very similar qualitative behaviour and the error in the displacement fields remains in the low single-digit percentage range, despite the underdeterminacy issues arising from the non-lattice-conforming interior boundary. Though instabilities are a common issue with the BGK-LBM [2], the transient LBM simulations here remain stable for a sufficiently long duration to capture the transient overshoots in the displacement of the dynamically loaded solid. However, further development in non-lattice conforming boundary conditions is necessary, as is apparent via the offset in the peaks of the postprocessed displacements, which appears to a considerably lesser extent in the example with lattice-conforming boundaries considered in Sect. 5.

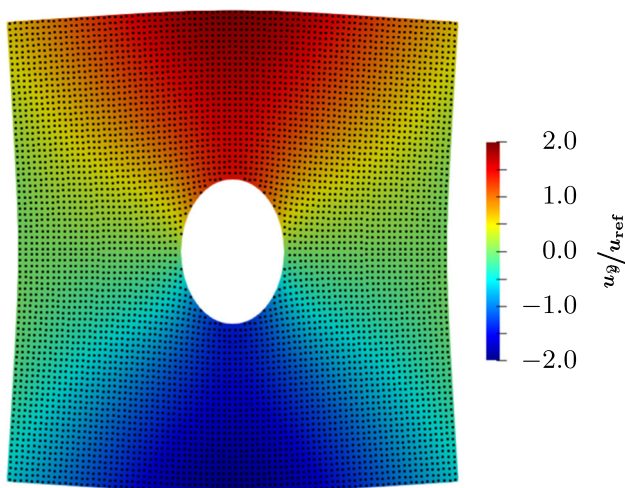


Fig. 9 Deformed configurations predicted at $t = 1.5l/c_s$ by FEM and LBM simulations. A contour plot for the vertical displacement u_2 is superimposed on the warped FEM domain, while the lattice sites of the LBM simulation are indicated by black dots. The deformation is scaled by a factor of 20

6.2 Sudden loading of a stationary crack

In the second numerical example, the LBM is used to treat a problem of dynamic fracture mechanics: the square cross-section in Fig. 10 with side length $l = 4.0l_c$ and free boundaries features a stationary crack of length l_c . The finite material domain modelled here is sufficiently large for our purposes: no P waves reflected by the exterior boundaries can return to and influence the crack within the simulated time. We non-dimensionalised this problem with $l_{ref} = l_c = 1.0$ (arbitrary units). Here, we use the dilatational wave speed c_d for non-dimensionalisation, as it determines the time of arrival of the P waves which are most characteristic for this problem.

The crack is oriented in the y-direction and centred at the origin $(0l/l_c, 0l/l_c)$. Both crack faces are suddenly loaded

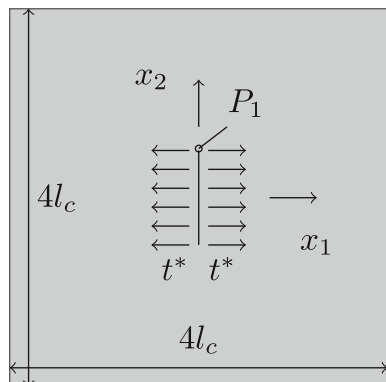


Fig. 10 Square domain with a crack (length l_c) subjected to tensile crack face traction. Side length not to scale

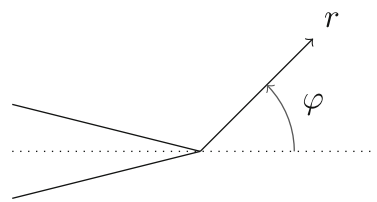


Fig. 11 Cylindrical coordinate system at the crack tip and definition of the displacement. Adapted from [10, p.71]

with a traction $t^* = 0.009615\mu u_{ref}/l$ in their negative normal directions at time $t = 0.0l/c_d$, and the traction remains constant until the end of the simulation at $t = 2.0l/c_d$. The LBM is used to simulate the consequences of this sudden so-called ‘mode I’ loading of the crack for a non-Poisson solid with $c_d^2/c_s^2 = 2.769$, $\nu = 0.2174$. A spatial discretisation with $\Delta x = 0.01l_c$ is chosen, leading to a time-step of $\Delta t \approx 2.669 \times 10^{-3}l/c_d$. We again use a relaxation time of $\bar{\tau} = 0.55\Delta t$.

Following the simulation, the x_1 -component of the Cauchy stress σ_{11} at a point $P_1 = (0.0l_c, 0.52l_c)$ just in front of one of the crack tips is evaluated via the second moment (20) and the formula relating the Poisson and Cauchy tensors (8). The result is used to calculate the mode I stress intensity factor K_I for each time step via [19, 74]

$$K_I = \lim_{r \rightarrow 0} \sigma_{11}(r, \varphi) \sqrt{2\pi r}. \tag{24}$$

Here, r is the distance from the crack tip to P_1 and $\varphi = 0$ the angle with which P_1 is offset from the crack plane (see Fig. 11).

The results are compared to the exact analytical solution from Freund [3, p.117], which yields

$$K_I^0 = 2t^* \frac{\sqrt{1-2\nu}}{1-\nu} \sqrt{\frac{c_d t}{\pi}}, \tag{25}$$

for the mode-I stress intensity factor from $t = 0.0l_c/c_d$ to $t = 1.0l_c/c_d$, i.e. until the first dilatational waves scattered by one crack tip arrive at the other [3, p.119]. The stress intensity factor from $t = l_c/c_d$ to $t = 2l_c/c_d$ (i.e., until the dilatational waves return to the other crack tip once more) is given by the nested integral [3, p.122]

$$K_I^1 = K_I^0 + \frac{2t^* F_+(0)}{\pi} \sqrt{\frac{2l_c}{\pi}} \dots \int_a^{t/c} \text{Im} \left(\sqrt{\frac{t}{l_c} - \eta} \sqrt{\frac{a - \eta c_r + \eta}{a + \eta c_r - \eta}} \frac{1}{\eta} \exp \left(\dots \frac{1}{\pi} \int_a^b \arctan \left(4\xi^2 \frac{\sqrt{(\xi^2 - a^2)(b^2 - \xi^2)}}{(b^2 - 2\xi^2)^2} \right) \dots \frac{2\eta}{\xi^2 - \eta^2} d\xi \right) d\eta \right), \tag{26}$$

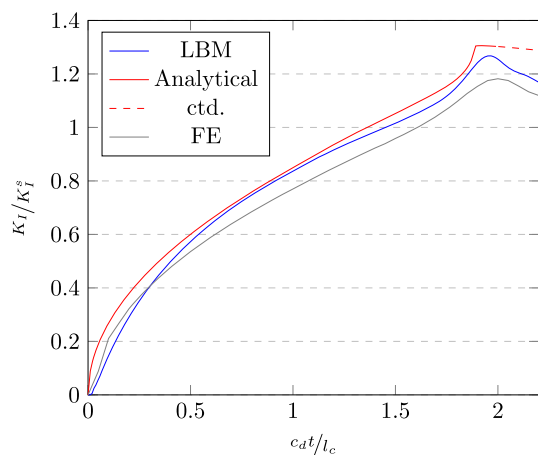


Fig. 12 Mode-I stress intensity factor postprocessed from the LBM simulation of a non-Poisson solid with a shock-loaded crack at $P_1 = (0.005, 0.52)$, compared with an analytical solution calculated using the Wiener-Hopf method. Stress intensity factor normalised with the quasistatic solution $K_I^s = t^* \sqrt{\pi l_c / 2}$

with $a = 1/c_d$ and $b = 1/c_s$. $c_r = 1/c$ is the inverse of the Rayleigh (surface) wave speed. In contrast to [3, p.83], where an approximation is used, we compute c via the roots of the equation [20]

$$0 = \left(2 - \left(\frac{c}{c_s}\right)^2\right)^2 - 4\sqrt{\left(1 - \left(\frac{c}{c_d}\right)^2\right)\left(1 - \left(\frac{c}{c_s}\right)^2\right)}.$$

The meaning of the function $F_+(\xi)$ is explained in [3, p.90]. Here, it suffices to know that it assumes the value of $F_+(0) \approx 0.8774$ in (26).

Crucially, the inner and outer integrands feature poles at $\xi = \eta$ and $\eta = c_r$, respectively. Since the integrands are sufficiently well-behaved and change sign as the variable of integration crosses the pole, Cauchy principal values exist [3, p.122]. Here, we evaluate (26) using a trapezoidal rule with an adaptive integration step near the poles, and we integrate around the inner singularity $\xi = \eta$ along a small semi-circular contour in the upper complex half-plane. A tolerance of 2×10^{-5} at the singularities seems sufficient for convergence. A lower tolerance does not visibly change the computed analytical solution. The MATLAB script for the evaluation of the Cauchy principal value integrals is provided with the supplementary material (Online Resource 1).

The numerically integrated analytical solution for K_I is plotted against the results postprocessed from the LBM simulation run at P_1 in Fig. 12. The analytical solution is not exact after $t = 2l_c/c_d$, and the plot is continued with a dashed line. The exact solution, however, is expected to deviate only slightly from this dashed line [3, p.123]. The LBM produces results that generally agree well with the analytical solution and a reference FE calculation: the stress intensity factor rises steeply initially, with the rate of change decreasing gradually.

After $t = \frac{l_c}{c_d}$ the solutions both behave almost linearly in time as stress waves from the other end of the crack arrive, with a notable uptick just before $t = 2\frac{l_c}{c_d}$ due to the arrival of Rayleigh surface waves [21]. K_I then levels off. The LBM (with the modified boundary rules) and the FEM produce a smoother behaviour than the exact solution.

However, the solution produced by the LBM captures key aspects of the behaviour in the vicinity of the crack tip roughly as well as the FEM, both qualitatively and quantitatively. The proposed computation of the Cauchy stress σ via Poisson stress \mathbf{P} and density in the current configuration ρ (see (8)) appears to work as desired in this example. Encouragingly, the simulation remains stable in the considered interval despite the discontinuous loading and the singular stress field in the vicinity of the crack tip.

7 Conclusions

In this work, local boundary rules for the LBM algorithm by Murthy et al. [1] and Escande et al. [2] were proposed. These approximate the behaviour of Dirichlet and Neumann boundary conditions with first-order accuracy.

In Sect. 2, the Lamé-Navier equation and the associated boundary conditions were first rephrased in moment chain form. In contrast to the presentation chosen by Murthy et al. [1], the reformulation was performed in the reference configuration. The relationship between the Poisson stress tensor \mathbf{P} appearing in this moment chain and the established Cauchy stress tensor σ was explored, and a convenient formula for the calculation of one from the other derived. Section 3 then very briefly outlined the Lattice Boltzmann Method for solids, and Sect. 4 presented the boundary rules used to model arbitrarily valued Dirichlet and Neumann boundary conditions. In Sect. 5, an empirical convergence study was performed for a simple benchmark example, showing linear convergence towards a high-fidelity FEM reference solution. In Sect. 6 two further examples—featuring a circular hole and a finite crack, respectively—were considered to illustrate the application of the LBM to elastodynamic problems. In appendix A, the LBM proposed by Murthy et al. [1] was shown to act as a second-order Crank-Nicolson scheme for the moment chain formulated in Sect. 2; in appendix C the simple bounce-back and anti-bounce-back rules were shown to replicate the desired boundary values with formal first-order accuracy.

While only first-order accuracy could be achieved here, the results indicate that solid-mechanical (Dirichlet and Neumann) boundary conditions can be modeled consistently via bounce-back-type boundary rules in the LB algorithm by Murthy et al. [1]. Despite the linear convergence rate and the eventual instabilities, the relative error of the LBM solution could be reduced below 0.1% with moderate mesh refinement, which is satisfactory for many engineering appli-

cations. Further improvements must be made to the stability of the method, as only finite-time simulations are possible to date. This issue is especially pronounced for material parameters which deviate significantly for the Poisson case ($\lambda = \mu$), as highlighted by [2]. Simulations of incompressible or nearly incompressible materials are therefore impossible currently. Furthermore, higher-order convergence would be desirable to achieve higher accuracy at lower resolutions.

Beyond this, the algorithm outlined here only facilitates the modelling of materially and geometrically linear solid behaviour. For an extension to the nonlinear regime, nonlinear terms in the material law must be accounted for. This may be challenging, as nonlinear terms can not easily be modeled via the third equation of the moment chain (4). As a first approach, source terms could be used to account for nonlinear terms in the material law, but depending on the finite difference stencils required to this end, this may interfere with the computational efficiency of the LBM.

On the other hand, the LBM seems promising as an elastodynamic solver: because the interaction between neighbouring lattice sites in the streaming step is linear, and the nonlinear behaviour enters the local collision step [12, p.55], the LBM can be implemented efficiently and parallelised easily. In particular, the LBM lends itself to modelling dynamic phenomena which call for fine spatial and temporal resolutions—an area of application in which computational efficiency becomes crucial. As discussed in Sect. 5, our in-house Python implementation required half the simulation time of an equivalently discretised explicit FE simulation using FEAP for a simple example problem. While a thorough performance study would be necessary to assess the relative performance of the LBM in detail, the preliminary results are encouraging. Furthermore, spatially discretising - ‘meshing’ - a material domain is comparatively easy. If the solid LBM can be turned into a practicable simulation method, much time might be saved in engineering design processes.

The numerical solutions to the dynamic example problems considered here are generally encouraging: the solid LBM by Murthy et al. [1] and the boundary rules presented here are able to consistently model elastodynamic problems described by the Lamé-Navier equation and Dirichlet and Neumann boundary conditions.

However, a need for further development is apparent in several key areas. Firstly, the stability of the method must be improved. Schemes with multiple relaxation times [12, p.407] and multi-reflection boundary rules [22] seem promising first places to start. Second-order accurate boundary rules could further improve the accuracy of the method. A possible approach to implement these may be to derive bounce-back-type boundary rules which eliminate the error terms arising in the consistency study in appendix C. Finally, extensions to material nonlinearity and large deformations are desirable.

If these shortcomings are addressed, however, fascinating areas of application beckon: the LBM could be used to study stresses caused by the propagation, reflection, and superposition of waves under highly dynamic loading. Crashproofing and mining are fields that might benefit from such studies. If an extension to the nonlinear regime can be achieved, simulations of polymer components (such as seals and tires) or tissue (such as muscle) under transient loads further become feasible. Here, also, the LBM might be useful on account of its efficiency and ease of parallelisation. The formulation in the reference configuration chosen here circumvents one key obstacle in the extension to the geometrically nonlinear regime: a formulation with respect to material rather than spatial coordinates means that no re-discretisation is necessary to deal with the large topological changes that may occur at finite strains.

Supplementary information

The MATLAB script for the evaluation of the analytical solution from [3, p.117] is provided with the supplementary material (Online Resource 1).

Acknowledgements The authors gratefully acknowledge the funding by the German Research Foundation (DFG) within the project 423809639. Open access funding enabled and organized by Projekt DEAL.

Funding Open Access funding enabled and organized by Projekt DEAL.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Appendix A: The LBM as a solver for moment chains

In line with [14], we perform a consistency study (based on Taylor expansions) to demonstrate that the BGK-LB scheme

outlined above recovers the desired moment chain in the limit as $\Delta t \rightarrow 0$.

To simplify the analysis, a notational shorthand will be introduced here: Moments are denoted as ${}^N\Pi$, their tensorial order as a left-hand superscript, and the term of which the moment is taken as a right-hand superscript. A set of indices $[\alpha] = \alpha_1, \dots, \alpha_N$ will concisely be written in square brackets. Additionally, the dyadic product of the velocity vector \mathbf{c}_i taken $N - 1$ times is denoted via a dyadic power operator \otimes^N in the superscript³. The N -th moment of \bar{f}_i can then for example be written as

$${}^N\Pi_{[\alpha]}^{\bar{f}} = \sum_i c_{i[\alpha]}^{\otimes N} \bar{f}_i.$$

In line with the second-order accurate discretisation of [15], the distribution functions \bar{f}_i are defined as

$$\bar{f}_i = f_i + \frac{\Delta t}{2\tau}(f_i - f_i^{\text{eq}}) - \frac{\Delta t}{2}\psi_i = f_i + \frac{\Delta t}{2\tau}f_i^{\text{neq}} - \frac{\Delta t}{2}\psi_i, \tag{A1}$$

in terms of the first-order distribution functions f_i , the moments of which are

$$\sum_i f_i = \rho, \quad \sum_i c_{i\alpha} f_i = j_\alpha, \quad \text{and} \quad \sum_i c_{i\alpha} c_{i\beta} f_i = P_{\alpha\beta}.$$

The post-collision distribution function is written in terms of the first-order accurate discretisation as

$$\begin{aligned} \bar{f}_i^{\text{col}} &= \bar{f}_i(\mathbf{x}, t) - \frac{2\Delta t}{2\tau + \Delta t}(\bar{f}_i(\mathbf{x}, t) - f_i^{\text{eq}}(\mathbf{x}, t)) \\ &+ \frac{2\tau}{2\tau + \Delta t}\psi_i(\mathbf{x}, t)\Delta t = f_i - \frac{\Delta t}{2\tau}f_i^{\text{neq}} + \frac{\Delta t}{2}\psi_i. \end{aligned}$$

As in [14], terms evaluated at $\mathbf{x} - \mathbf{c}_i\Delta t$ in the streaming step are related to spatial derivatives at \mathbf{x} via a Taylor expansion. In order to establish a relation to the field variables to be simulated, the moments of the resulting equations are then taken. Firstly, the Taylor expansion of \bar{f}_i^{col} at some neighbouring point $\mathbf{x} - \mathbf{c}_i\Delta t$ of lattice point \mathbf{x} , yields, with $\mathbf{x} - \mathbf{c}_i\Delta t = \mathbf{x} + \mathbf{c}_{\bar{i}}\Delta t$ and $\mathbf{c}_i = -\mathbf{c}_{\bar{i}}$ [14]:

$$\bar{f}_i^{\text{col}}(\mathbf{x} - \mathbf{c}_i\Delta t, t) = \bar{f}_i^{\text{col}}(\mathbf{x}, t) + \sum_{k=1}^{\infty} \frac{(-\Delta t)^k}{k!} (c_{i\alpha} d_\alpha)^k \bar{f}_i^{\text{col}}(\mathbf{x}, t).$$

The N -th moments of the above are then taken to move the discussion to moment space

$${}^N\Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x} - \mathbf{c}_i\Delta t, t) = \sum_i c_{i[\alpha]}^{\otimes N} \bar{f}_i^{\text{col}}(\mathbf{x}, t)$$

³ The dyadic power operator is defined such that $\mathbf{c}_i^{\otimes 1} = \mathbf{c}_i$, $\mathbf{c}_i^{\otimes 2} = \mathbf{c}_i \otimes \mathbf{c}_i$, $\mathbf{c}_i^{\otimes 3} = \mathbf{c}_i \otimes \mathbf{c}_i \otimes \mathbf{c}_i$, etc..

$$+ \sum_i c_{i[\alpha]}^{\otimes N} \sum_{k=1}^{\infty} \frac{(-\Delta t)^k}{k!} (c_{i\beta} d_\beta)^k \bar{f}_i^{\text{col}}(\mathbf{x}, t).$$

Notice that the first term on the right hand side is simply the N -th moment of \bar{f}_i^{col} , while in the second term, $(c_{i\beta} d_\beta)^k$ can be expanded to yield

$$\begin{aligned} {}^N\Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x} - \mathbf{c}_i\Delta t, t) &= {}^N\Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) \\ &+ \sum_{k=1}^{\infty} \frac{(-\Delta t)^k}{k!} \\ &\sum_i c_{i[\alpha]}^{\otimes N} \otimes c_{i[\beta]}^{\otimes k} d_{\beta_1} \dots d_{\beta_k} \bar{f}_i^{\text{col}}(\mathbf{x}, t). \end{aligned}$$

The k -th term of the (Taylor) series $\sum_{k=1}^{\infty}(\dots)$ can thus be written via k derivatives $d_{\beta_1} \dots d_{\beta_k}$ of the $N + k$ -th moment $\sum_i c_{i\alpha}^{\otimes N+k} f_i^{\text{col}}(\mathbf{x}, t)$

$$\begin{aligned} {}^N\Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x} - \mathbf{c}_i\Delta t, t) &= {}^N\Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) \\ &+ \sum_{k=1}^{\infty} \frac{(-\Delta t)^k}{k!} d_{\beta_1} \dots d_{\beta_k} {}^{N+k}\Pi_{[\alpha, \beta]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t). \end{aligned} \tag{A2}$$

Additionally, it will be convenient in the derivation to note the N -th moment of the streaming relation (18), which yields

$${}^N\Pi_{[\alpha]}^{\bar{f}}(\mathbf{x}, t + \Delta t) = {}^N\Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x} - \mathbf{c}_i\Delta t, t). \tag{A3}$$

Farag et al. [14] introduced an approximation for higher-order moments in terms of lower-order moments, which is outlined below. The $N + 2$ -th moment can be expressed via the $N + 1$ -th moment, enabling a second-order truncation using only one higher-order moment. For this, consider (A2) for the $N + 1$ -th moment

$$\begin{aligned} {}^{N+1}\Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x} - \mathbf{c}_i\Delta t, t) &= {}^{N+1}\Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) \\ &+ \sum_{k=1}^{\infty} \frac{(-\Delta t)^k}{k!} d_{\beta_1} \dots d_{\beta_k} {}^{N+1+k}\Pi_{[\alpha, \beta]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t), \end{aligned}$$

and truncate after the first derivative term

$$\begin{aligned} {}^{N+1}\Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x} - \mathbf{c}_i\Delta t, t) &= {}^{N+1}\Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) \\ &- \Delta t d_{\beta_1} {}^{N+2}\Pi_{[\alpha], \beta_1}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) + \mathcal{O}(\Delta t^2). \end{aligned}$$

Rearranging yields

$$\begin{aligned} \Delta t d_{\beta_1} {}^{N+2}\Pi_{[\alpha], \beta_1}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) &= {}^{N+1}\Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) \\ &- {}^{N+1}\Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x} - \mathbf{c}_i\Delta t, t) + \mathcal{O}(\Delta t^2). \end{aligned}$$

The second term on the right hand side can be simplified using the moments of the streaming relation (A3) to yield

$$\Delta t d_{\beta_1}^{N+2} \Pi_{[\alpha], \beta_1}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) = {}^{N+1} \Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) - {}^{N+1} \Pi_{[\alpha]}^{\bar{f}}(\mathbf{x}, t + \Delta t) + \mathcal{O}(\Delta t^2). \tag{A4}$$

Equation (A2) is truncated after the second term

$$\begin{aligned} {}^N \Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x} - \mathbf{c}_i \Delta t, t) &= {}^N \Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) \\ &+ \sum_{k=1}^2 \frac{(-\Delta t)^k}{k!} d_{\beta_1} \cdots d_{\beta_k} {}^{N+k} \Pi_{[\alpha], \beta_1}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) + \mathcal{O}(\Delta t^3) \\ &= {}^N \Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) - \Delta t d_{\beta_1} {}^{N+1} \Pi_{[\alpha], \beta_1}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) \\ &+ \frac{\Delta t^2}{2} d_{\beta_1} d_{\beta_2} {}^{N+2} \Pi_{[\alpha], \beta_1, \beta_2}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) + \mathcal{O}(\Delta t^3). \end{aligned}$$

and (A4) can be substituted into $\frac{\Delta t^2}{2} d_{\beta_1} d_{\beta_2} {}^{N+2} \Pi_{[\alpha], \beta_1, \beta_2}^{\bar{f}^{\text{col}}}(\mathbf{x}, t)$. Note that the β_1 in (A4) becomes β_2 here, and terms are rearranged for convenience. This yields,

$$\begin{aligned} {}^N \Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x} - \mathbf{c}_i \Delta t, t) - {}^N \Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) &= -\Delta t d_{\beta_1} {}^{N+1} \Pi_{[\alpha], \beta_1}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) + \frac{\Delta t}{2} d_{\beta_1} \left({}^{N+1} \Pi_{[\alpha], \beta_1}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) \right. \\ &\quad \left. - {}^{N+1} \Pi_{[\alpha], \beta_1}^{\bar{f}}(\mathbf{x}, t + \Delta t) \right) + \mathcal{O}(\Delta t^3) \\ &= -\frac{\Delta t}{2} d_{\beta_1} \left({}^{N+1} \Pi_{[\alpha], \beta_1}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) + {}^{N+1} \Pi_{[\alpha], \beta_1}^{\bar{f}}(\mathbf{x}, t + \Delta t) \right) \\ &\quad + \mathcal{O}(\Delta t^3). \end{aligned} \tag{A5}$$

In (A5), the N -th moment of the streaming relation (A3) can again be used for simplification. Additionally, the results may be expressed in terms of the moments of the first-order distribution function ${}^N \Pi^f = {}^N \Pi$, as these are the tensors to be modeled. The moments of the second-order distribution functions ${}^N \Pi^{\bar{f}}$ can be stated in terms of these as

$${}^N \Pi_{[\alpha]}^{\bar{f}} = {}^N \Pi_{[\alpha]}^f + \frac{\Delta t}{2} \left(\frac{1}{\tau} {}^N \Pi_{[\alpha]}^{f^{\text{neq}}} - {}^N S_{[\alpha]}^{\text{FD}} \right),$$

meaning that (A3) becomes

$$\begin{aligned} {}^N \Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x} - \mathbf{c}_i \Delta t, t) &= {}^N \Pi_{[\alpha]}^f(\mathbf{x}, t + \Delta t) \\ &+ \frac{\Delta t}{2} \left(\frac{1}{\tau} {}^N \Pi_{[\alpha]}^{f^{\text{neq}}}(\mathbf{x}, t + \Delta t) - {}^N S_{[\alpha]}^{\text{FD}}(\mathbf{x}, t + \Delta t) \right). \end{aligned} \tag{A6}$$

Meanwhile, the moments of the modified post-collision distribution functions \bar{f}_i^{col} at (\mathbf{x}, t) and the modified distribution functions \bar{f}_i at $(\mathbf{x}, t + \Delta t)$ - both of which appear on

the right in (A5) - are given in terms of the original distribution functions as

$$\begin{aligned} {}^N \Pi_{[\alpha]}^{\bar{f}}(\mathbf{x}, t + \Delta t) &= {}^N \Pi_{[\alpha]}^f(\mathbf{x}, t + \Delta t) \\ &+ \frac{\Delta t}{2} \left(\frac{1}{\tau} {}^N \Pi_{[\alpha]}^{f^{\text{neq}}}(\mathbf{x}, t + \Delta t) - {}^N S_{[\alpha]}^{\text{FD}}(\mathbf{x}, t + \Delta t) \right), \end{aligned}$$

and

$$\begin{aligned} {}^N \Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) &= {}^N \Pi_{[\alpha]}^f(\mathbf{x}, t) - \frac{\Delta t}{2} \left(\frac{1}{\tau} {}^N \Pi_{[\alpha]}^{f^{\text{neq}}}(\mathbf{x}, t) \right. \\ &\quad \left. - {}^N S_{[\alpha]}^{\text{FD}}(\mathbf{x}, t) \right). \end{aligned} \tag{A7}$$

Their sum - which appears in the bracket in (A5) - is given by

$$\begin{aligned} {}^N \Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) + {}^N \Pi_{[\alpha]}^{\bar{f}}(\mathbf{x}, t + \Delta t) &= {}^N \Pi_{[\alpha]}^f(\mathbf{x}, t + \Delta t) + {}^N \Pi_{[\alpha]}^f(\mathbf{x}, t) \\ &+ \frac{\Delta t}{2} \left(\frac{1}{\tau} {}^N \Pi_{[\alpha]}^{f^{\text{neq}}}(\mathbf{x}, t + \Delta t) - {}^N S_{[\alpha]}^{\text{FD}}(\mathbf{x}, t + \Delta t) \right) \\ &- \frac{\Delta t}{2} \left(\frac{1}{\tau} {}^N \Pi_{[\alpha]}^{f^{\text{neq}}}(\mathbf{x}, t) - {}^N S_{[\alpha]}^{\text{FD}}(\mathbf{x}, t) \right), \end{aligned}$$

which—with first-order accuracy—is equivalent to

$$\begin{aligned} {}^N \Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) + {}^N \Pi_{[\alpha]}^{\bar{f}}(\mathbf{x}, t + \Delta t) &= {}^N \Pi_{[\alpha]}^f(\mathbf{x}, t + \Delta t) \\ &+ {}^N \Pi_{[\alpha]}^f(\mathbf{x}, t) + \mathcal{O}(\Delta t^2). \end{aligned}$$

For the $N + 1$ -th moment, this can be substituted into (A5)

$$\begin{aligned} {}^N \Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x} - \mathbf{c}_i \Delta t, t) - {}^N \Pi_{[\alpha]}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) &= -\frac{\Delta t}{2} d_{\beta_1} \left({}^{N+1} \Pi_{[\alpha], \beta_1}^{\bar{f}^{\text{col}}}(\mathbf{x}, t) + {}^{N+1} \Pi_{[\alpha], \beta_1}^{\bar{f}}(\mathbf{x}, t + \Delta t) \right) \\ &+ \mathcal{O}(\Delta t^3) \\ &= -\frac{\Delta t}{2} d_{\beta_1} \left({}^{N+1} \Pi_{[\alpha], \beta_1}^f(\mathbf{x}, t + \Delta t) + {}^{N+1} \Pi_{[\alpha], \beta_1}^f(\mathbf{x}, t) \right) \\ &+ \mathcal{O}(\Delta t^3). \end{aligned}$$

With the moments of the modified streaming relation (A6) and the modified post-collision distribution functions (A7), this becomes

$$\begin{aligned} {}^N \Pi_{[\alpha]}^f(\mathbf{x}, t + \Delta t) - {}^N \Pi_{[\alpha]}^f(\mathbf{x}, t) &+ \frac{\Delta t}{2} \left(\frac{1}{\tau} {}^N \Pi_{[\alpha]}^{f^{\text{neq}}}(\mathbf{x}, t + \Delta t) - {}^N S_{[\alpha]}^{\text{FD}}(\mathbf{x}, t + \Delta t) \right) \\ &+ \frac{\Delta t}{2} \left(\frac{1}{\tau} {}^N \Pi_{[\alpha]}^{f^{\text{neq}}}(\mathbf{x}, t) - {}^N S_{[\alpha]}^{\text{FD}}(\mathbf{x}, t) \right) \\ &= -\frac{\Delta t}{2} d_{\beta_1} \left({}^{N+1} \Pi_{[\alpha], \beta_1}^f(\mathbf{x}, t + \Delta t) \right. \\ &\quad \left. + {}^{N+1} \Pi_{[\alpha], \beta_1}^f(\mathbf{x}, t) \right) + \mathcal{O}(\Delta t^3). \end{aligned}$$

Rearranging yields

$$\begin{aligned} & N \Pi_{[\alpha]}^f(\mathbf{x}, t + \Delta t) - N \Pi_{[\alpha]}^f(\mathbf{x}, t) \\ &= -\frac{\Delta t}{2} \left(\frac{1}{\tau} N \Pi_{[\alpha]}^{f, \text{neq}}(\mathbf{x}, t + \Delta t) - N S_{[\alpha]}^{\text{FD}}(\mathbf{x}, t + \Delta t) \right) \\ &\quad - \frac{\Delta t}{2} \left(\frac{1}{\tau} N \Pi_{[\alpha]}^{f, \text{neq}}(\mathbf{x}, t) - N S_{[\alpha]}^{\text{FD}}(\mathbf{x}, t) \right) \\ &\quad - \frac{\Delta t}{2} d_{\beta_1} \left(N^{+1} \Pi_{[\alpha], \beta_1}^f(\mathbf{x}, t + \Delta t) \right. \\ &\quad \left. + N^{+1} \Pi_{[\alpha], \beta_1}^f(\mathbf{x}, t) \right) + \mathcal{O}(\Delta t^3) \\ &= \frac{\Delta t}{2} \left(-d_{\beta_1} N^{+1} \Pi_{[\alpha], \beta_1}^f(\mathbf{x}, t + \Delta t) \right. \\ &\quad \left. - \frac{1}{\tau} N \Pi_{[\alpha]}^{f, \text{neq}}(\mathbf{x}, t + \Delta t) + N S_{[\alpha]}^{\text{FD}}(\mathbf{x}, t + \Delta t) \right) \\ &\quad + \frac{\Delta t}{2} \left(-d_{\beta_1} N^{+1} \Pi_{[\alpha], \beta_1}^f(\mathbf{x}, t) - \frac{1}{\tau} N \Pi_{[\alpha]}^{f, \text{neq}}(\mathbf{x}, t) \right. \\ &\quad \left. + N S_{[\alpha]}^{\text{FD}}(\mathbf{x}, t) \right) + \mathcal{O}(\Delta t^3), \end{aligned}$$

which is equivalent to

$$\begin{aligned} & \frac{1}{\Delta t} \left(N \Pi_{[\alpha]}^f(\mathbf{x}, t + \Delta t) - N \Pi_{[\alpha]}^f(\mathbf{x}, t) \right) \\ &= \frac{1}{2} \left(-d_{\beta_1} N^{+1} \Pi_{[\alpha], \beta_1}^f(\mathbf{x}, t + \Delta t) - \frac{1}{\tau} N \Pi_{[\alpha]}^{f, \text{neq}}(\mathbf{x}, t + \Delta t) \right. \\ &\quad \left. + N S_{[\alpha]}^{\text{FD}}(\mathbf{x}, t + \Delta t) \right) \\ &\quad + \frac{1}{2} \left(-d_{\beta_1} N^{+1} \Pi_{[\alpha], \beta_1}^f(\mathbf{x}, t) - \frac{1}{\tau} N \Pi_{[\alpha]}^{f, \text{neq}}(\mathbf{x}, t) \right. \\ &\quad \left. + N S_{[\alpha]}^{\text{FD}}(\mathbf{x}, t) \right) + \mathcal{O}(\Delta t^2). \end{aligned}$$

As outlined in appendix B, the moments of f_i^{neq} vanish in the non-dissipative limit. We are left with a second-order Crank-Nicolson scheme for a moment chain such as the one given in (4)—assuming the forcing terms computed via finite differences replicate their continuous equivalents with at least second-order accuracy.

Substituting the macroscopic moments (20) and source terms (17) yields

$$\begin{aligned} \frac{1}{\Delta t} \left(\rho(\mathbf{x}, t + \Delta t) - \rho(\mathbf{x}, t) \right) &= \frac{1}{2} \left(-d_\alpha j_\alpha(\mathbf{x}, t + \Delta t) \right. \\ &\quad \left. - d_\alpha j_\alpha(\mathbf{x}, t) \right) + \mathcal{O}(\Delta t^2), \end{aligned} \tag{A8}$$

for the zeroth,

$$\begin{aligned} \frac{1}{\Delta t} \left(j_\alpha(\mathbf{x}, t + \Delta t) - j_\alpha(\mathbf{x}, t) \right) &= \frac{1}{2} \left(-d_\beta P_{\alpha\beta}(\mathbf{x}, t + \Delta t) \right. \\ &\quad \left. + \rho_0 b_\alpha(\mathbf{x}, t + \Delta t) + N S_\alpha^{\text{FD}}(\mathbf{x}, t + \Delta t) \right) \\ &\quad + \frac{1}{2} \left(-d_\beta P_{\alpha\beta}(\mathbf{x}, t) + \rho_0 b_\alpha(\mathbf{x}, t) + N S_\alpha^{\text{FD}}(\mathbf{x}, t) \right) \\ &\quad + \mathcal{O}(\Delta t^2), \end{aligned} \tag{A9}$$

for the first, and

$$\begin{aligned} \frac{1}{\Delta t} \left(P_{\alpha\beta}(\mathbf{x}, t + \Delta t) - P_{\alpha\beta}(\mathbf{x}, t) \right) &= \frac{1}{2} \left(-d_\gamma Q_{\alpha\beta\gamma}(\mathbf{x}, t + \Delta t) \right) \\ &\quad + \frac{1}{2} \left(-d_\gamma Q_{\alpha\beta\gamma}(\mathbf{x}, t) \right) + \mathcal{O}(\Delta t^2), \end{aligned} \tag{A10}$$

for the second moment. For the sake of conciseness, we introduce \mathbf{Q} to denote the third moment

$$Q_{\alpha\beta\gamma} = \frac{\mu}{\rho_0} (j_\alpha \delta_{\beta\gamma} + j_\beta \delta_{\alpha\gamma} + j_\gamma \delta_{\beta\alpha}),$$

in (A10).

The non-Poisson component of the material law is implemented via a finite difference term in the second of the three equations in the chain. We use a second-order accurate central finite difference scheme to approximate the derivative appearing here, i.e.

$${}^1 S_\alpha^{\text{FD}} = \frac{\mu - \lambda}{\rho_0} d_\alpha^{\text{FD}} \rho = \frac{\mu - \lambda}{\rho_0} d_\alpha \rho + \mathcal{O}(\Delta x^2).$$

With acoustic scaling, i.e. $\Delta x \sim \Delta t$, we have

$${}^1 S_\alpha^{\text{FD}} = \frac{\mu - \lambda}{\rho_0} d_\alpha \rho + \mathcal{O}(\Delta t^2).$$

We therefore have, for the evolution equation for the first moment,

$$\begin{aligned} \frac{1}{\Delta t} \left(j_\alpha(\mathbf{x}, t + \Delta t) - j_\alpha(\mathbf{x}, t) \right) &= \frac{1}{2} \left(-d_\beta P_{\alpha\beta}(\mathbf{x}, t + \Delta t) \right. \\ &\quad \left. + \rho_0 b_\alpha(\mathbf{x}, t + \Delta t) + N S_\alpha(\mathbf{x}, t + \Delta t) \right) \\ &\quad + \frac{1}{2} \left(-d_\beta P_{\alpha\beta}(\mathbf{x}, t) + \rho_0 b_\alpha(\mathbf{x}, t) + N S_\alpha(\mathbf{x}, t) \right) \\ &\quad + \mathcal{O}(\Delta t^2), \end{aligned} \tag{A11}$$

i.e. a second-order leading error.

Notice that the discrete evolution equations for the density in the current configuration (A8), the momentum density configuration (A11), and the Poisson stress tensor (A10) amount to second-order Crank-Nicolson schemes for the target equations. In the limit as $\Delta t \rightarrow 0$ the BGK-LBM by Murthy et al. [1] therefore reproduces the continuous equivalent relations

$$\begin{aligned} d_t \rho + d_\alpha j_\alpha &= 0 + \mathcal{O}(\Delta t^2), \\ d_t j_\alpha + d_\beta P_{\alpha\beta} &= \rho_0 b_\alpha + \frac{\mu - \lambda}{\rho_0} d_\alpha \rho + \mathcal{O}(\Delta t^2), \\ d_t P_{\alpha\beta} + d_\gamma Q_{\alpha\beta\gamma} &= 0 + \mathcal{O}(\Delta t^2), \end{aligned}$$

which, as discussed above, is equivalent to the Lamé-Navier equations.

Appendix B: Distribution function expansion

Below, a consistency study based on Taylor expansions is performed to investigate the behaviour of boundary rules. Here, the terms evaluated at $\mathbf{x} + \mathbf{c}_i \Delta t$ or $t + \Delta t$ in the Lattice Boltzmann equation are related to terms at \mathbf{x} and t via Taylor expansions. The resulting partial differential equation is solved to obtain a second-order expansion of the non-equilibrium distribution function f_i^{neq} in terms of the equilibrium distribution function f_i^{eq} . The latter is known in terms of the moments appearing in the moment chain (4). This allows a relation to be established between the boundary rules, which are formulated in terms of distribution functions \bar{f}_i , and the target boundary conditions, which are formulated in terms of the moments (20).

To this end, the streaming relation is expanded

$$\bar{f}_i(\mathbf{x}, t + \Delta t) = \bar{f}_i^{\text{col}}(\mathbf{x}, t) + \sum_k \frac{(-\Delta t)^k}{k!} (c_{i\alpha} d_\alpha)^k \bar{f}_i^{\text{col}}(\mathbf{x}, t),$$

and the first-order distribution functions (A1) are substituted on both sides

$$\begin{aligned} f_i^{\text{eq}}(\mathbf{x}, t + \Delta t) + \left(1 + \frac{\Delta t}{2\tau}\right) f_i^{\text{neq}}(\mathbf{x}, t + \Delta t) - \frac{\Delta t}{2} \psi_i(\mathbf{x}, t + \Delta t) \\ = f_i^{\text{eq}} + \left(1 - \frac{\Delta t}{2\tau}\right) f_i^{\text{neq}} + \frac{\Delta t}{2} \psi_i + \sum_k \frac{(-\Delta t)^k}{k!} (c_{i\alpha} d_\alpha)^k \\ \left(f_i^{\text{eq}} + \left(1 - \frac{\Delta t}{2\tau}\right) f_i^{\text{neq}} + \frac{\Delta t}{2} \psi_i\right). \end{aligned}$$

This can be rearranged to yield

$$\begin{aligned} \left(f_i^{\text{eq}}(\mathbf{x}, t + \Delta t) - f_i^{\text{eq}}(\mathbf{x}, t)\right) + \left(f_i^{\text{neq}}(\mathbf{x}, t + \Delta t) - f_i^{\text{neq}}(\mathbf{x}, t)\right) \\ + \frac{\Delta t}{2\tau} \left(f_i^{\text{neq}}(\mathbf{x}, t + \Delta t) + f_i^{\text{neq}}(\mathbf{x}, t)\right) \\ - \frac{\Delta t}{2} \left(\psi_i(\mathbf{x}, t + \Delta t) + \psi_i(\mathbf{x}, t)\right) \\ = \sum_k \frac{(-\Delta t)^k}{k!} (c_{i\alpha} d_\alpha)^k f_i^{\text{eq}}(\mathbf{x}, t) + \left(1 - \frac{\Delta t}{2\tau}\right) \\ \sum_k \frac{(-\Delta t)^k}{k!} (c_{i\alpha} d_\alpha)^k f_i^{\text{neq}}(\mathbf{x}, t) \\ + \frac{\Delta t}{2} \sum_k \frac{(-\Delta t)^k}{k!} (c_{i\alpha} d_\alpha)^k \psi_i(\mathbf{x}, t) \end{aligned}$$

and related to the behaviour at (\mathbf{x}, t) via Taylor expansions. To arbitrary order, this yields

$$\sum_k \frac{\Delta t^k}{k!} d_t^{(k)} f_i^{\text{eq}} + \sum_k \frac{\Delta t^k}{k!} d_t^{(k)} f_i^{\text{neq}} + \frac{\Delta t}{\tau} f_i^{\text{neq}}$$

$$\begin{aligned} + \frac{\Delta t}{2\tau} \sum_k \frac{\Delta t^k}{k!} d_t^{(k)} f_i^{\text{neq}} - \psi_i \Delta t - \frac{\Delta t}{2} \sum_k \frac{\Delta t^k}{k!} d_t^{(k)} \psi_i \\ = \sum_k \frac{(-\Delta t)^k}{k!} (c_{i\alpha} d_\alpha)^k f_i^{\text{eq}} + \left(1 - \frac{\Delta t}{2\tau}\right) \\ \sum_k \frac{(-\Delta t)^k}{k!} (c_{i\alpha} d_\alpha)^k f_i^{\text{neq}} + \frac{\Delta t}{2} \sum_k \frac{(-\Delta t)^k}{k!} (c_{i\alpha} d_\alpha)^k \psi_i, \end{aligned}$$

or, when rearranged,

$$\begin{aligned} \frac{\Delta t}{\tau} f_i^{\text{neq}} + \left(1 + \frac{\Delta t}{2\tau}\right) \sum_k \frac{\Delta t^k}{k!} d_t^k f_i^{\text{neq}} - \left(1 - \frac{\Delta t}{2\tau}\right) \\ \sum_k \frac{(-\Delta t)^k}{k!} (c_{i\alpha} d_\alpha)^k f_i^{\text{neq}} \\ = - \sum_k \frac{\Delta t^k}{k!} d_t^k f_i^{\text{eq}} + \sum_k \frac{(-\Delta t)^k}{k!} (c_{i\alpha} d_\alpha)^k f_i^{\text{eq}} + \psi_i \Delta t \\ + \frac{\Delta t}{2} \sum_k \frac{\Delta t^k}{k!} d_t^k \psi_i + \frac{\Delta t}{2} \sum_k \frac{(-\Delta t)^k}{k!} (c_{i\alpha} d_\alpha)^k \psi_i. \end{aligned}$$

Up terms of second order in Δt , this partial differential equation for f_i^{neq} is solved in terms of f_i^{eq} and ψ_i by

$$f_i^{\text{neq}} = -\tau d_t f_i^{\text{eq}} - \tau c_{i\alpha} d_\alpha f_i^{\text{eq}} + \tau \psi_i + \mathcal{O}(\Delta t^2). \tag{B12}$$

The second-order distribution function \bar{f}_i can therefore be written as

$$\begin{aligned} \bar{f}_i = f_i^{\text{eq}} + \left(1 + \frac{\Delta t}{2\tau}\right) \left(-\tau d_t f_i^{\text{eq}} - \tau c_{i\alpha} d_\alpha f_i^{\text{eq}} + \tau \psi_i\right) \\ - \frac{\Delta t}{2} \psi_i + \mathcal{O}(\Delta t^2), \end{aligned}$$

and the post-collision distribution functions become

$$\begin{aligned} \bar{f}_i^{\text{col}} = f_i^{\text{eq}} + \left(1 - \frac{\Delta t}{2\tau}\right) \left(-\tau d_t f_i^{\text{eq}} - \tau c_{i\alpha} d_\alpha f_i^{\text{eq}} + \tau \psi_i\right) \\ + \frac{\Delta t}{2} \psi_i + \mathcal{O}(\Delta t^2). \end{aligned}$$

With [14]

$$\bar{\tau} = \tau \left(1 + \frac{\Delta t}{2\tau}\right), \quad \text{and} \quad \tau = \bar{\tau} \left(1 - \frac{\Delta t}{2\bar{\tau}}\right),$$

we have

$$\bar{f}_i = f_i^{\text{eq}} - \bar{\tau} d_t f_i^{\text{eq}} - \bar{\tau} c_{i\alpha} d_\alpha f_i^{\text{eq}} + \bar{\tau} \left(1 - \frac{\Delta t}{2\bar{\tau}}\right) \psi_i + \mathcal{O}(\Delta t^2),$$

and

$$\bar{f}_i^{\text{col}} = f_i^{\text{eq}} - (\bar{\tau} - \Delta t) d_t f_i^{\text{eq}} - (\bar{\tau} - \Delta t) c_{i\alpha} d_\alpha f_i^{\text{eq}}$$

$$+\bar{\tau}\left(1-\frac{\Delta t}{2\bar{\tau}}\right)\psi_i+\mathcal{O}(\Delta t^2).$$

Note that, on account of (B12), the moments of f_i^{neq} become, with second order accuracy

$${}^N\Pi_{[\alpha]}^{f^{\text{neq}}}= -\tau(d_t^N\Pi_{[\alpha]}+d_\beta^N\Pi_{[\alpha]\beta}-{}^N S_{[\alpha]})+\mathcal{O}(\Delta t^2),$$

which vanishes in the absence of dissipative terms.

Appendix C: Analysis of boundary conditions

The first-order bounce-back

$$\bar{f}_i(\mathbf{x},t)=\bar{f}_i^{\text{col}}(\mathbf{x},t-\Delta t)-\frac{2}{c_s^2}w_i c_{i\alpha} j_\alpha^*, \tag{C13}$$

and anti-bounce-back

$$\begin{aligned} \bar{f}_i(\mathbf{x},t)=&-\bar{f}_i^{\text{col}}(\mathbf{x},t-\Delta t)+2w_i\left(\rho^{\text{bd}}\right. \\ &\left.+\frac{1}{2c_s^4}(P_{\alpha\beta}^*-\rho^{\text{bd}}c_s^2\delta_{\alpha\beta})(c_{i\alpha}c_{i\beta}-c_s^2\delta_{\alpha\beta})\right). \end{aligned} \tag{C14}$$

boundary rules can generally be written as

$$\bar{f}_i(\mathbf{x},t)=\pm\bar{f}_i^{\text{col}}(\mathbf{x},t-\Delta t)\mp 2w_i B_i,$$

or

$$\bar{f}_i(\mathbf{x},t)\mp\bar{f}_i^{\text{col}}(\mathbf{x},t-\Delta t)=\mp 2w_i B_i, \tag{C15}$$

where B_i denotes the term enforcing the relevant boundary values. The sums and differences of the distribution functions appearing in (C15) can be thought of as isolating some among the coupled terms in the distribution functions which contain information about the moments with regard to which the boundary conditions are formulated. In the following analysis, these sums and differences are discussed, and the resulting behaviour of the boundary rules in moment space is analysed.

Appendix C.1: Information filtering

With $w_{\bar{i}}=w_i$, and $c_{\bar{i}\alpha}=-c_{i\alpha}$, we have

$$f_{\bar{i}}^{\text{eq}}=w_i\left(\rho-\frac{1}{c_s^2}c_{i\alpha}j_\alpha+\frac{1}{2c_s^4}(P_{\alpha\beta}-\rho c_s^2\delta_{\alpha\beta})(c_{i\alpha}c_{i\beta}-c_s^2\delta_{\alpha\beta})\right),$$

and

$$\psi_{\bar{i}}=-\frac{1}{c_s^2}w_i c_{i\alpha} S_\alpha=-\psi_i,$$

and thus

$$\bar{f}_{\bar{i}}=f_{\bar{i}}^{\text{eq}}-\bar{\tau}d_t f_{\bar{i}}^{\text{eq}}+\bar{\tau}c_{i\alpha}d_\alpha f_{\bar{i}}^{\text{eq}}-\bar{\tau}\left(1-\frac{\Delta t}{2\bar{\tau}}\right)\psi_i+\mathcal{O}(\Delta t^2),$$

and

$$\begin{aligned} \bar{f}_i^{\text{col}}=&f_i^{\text{eq}}-(\bar{\tau}-\Delta t)d_t f_i^{\text{eq}}-(\bar{\tau}-\Delta t)c_{i\alpha}d_\alpha f_i^{\text{eq}} \\ &+\bar{\tau}\left(1-\frac{\Delta t}{2\bar{\tau}}\right)\psi_i+\mathcal{O}(\Delta t^2). \end{aligned}$$

Therefore, the distribution function difference appearing in the bounce-back rule (C13) yields

$$\begin{aligned} \bar{f}_i(\mathbf{x},t)-\bar{f}_i^{\text{col}}(\mathbf{x},t-\Delta t) &= \left(f_i^{\text{eq}}(\mathbf{x},t)-f_i^{\text{eq}}(\mathbf{x},t-\Delta t)\right)-\bar{\tau}\left(d_t f_i^{\text{eq}}(\mathbf{x},t)\right. \\ &\quad \left.-d_t f_i^{\text{eq}}(\mathbf{x},t-\Delta t)\right) \\ &\quad +\bar{\tau}\left(c_{i\alpha}d_\alpha f_i^{\text{eq}}(\mathbf{x},t)+c_{i\alpha}d_\alpha f_i^{\text{eq}}(\mathbf{x},t-\Delta t)\right) \\ &\quad -\bar{\tau}\left(1-\frac{\Delta t}{2\bar{\tau}}\right)\left(\psi_i(\mathbf{x},t)+\psi_i(\mathbf{x},t-\Delta t)\right) \\ &\quad -\Delta t\left(d_t f_i^{\text{eq}}(\mathbf{x},t-\Delta t)+c_{i\alpha}d_\alpha f_i^{\text{eq}}(\mathbf{x},t-\Delta t)\right) \\ &\quad +\mathcal{O}(\Delta t^2), \end{aligned} \tag{C16}$$

and the sum appearing in the anti-bounce-back rule (C14) becomes

$$\begin{aligned} \bar{f}_i(\mathbf{x},t)+\bar{f}_i^{\text{col}}(\mathbf{x},t-\Delta t) &= \left(f_i^{\text{eq}}(\mathbf{x},t)+f_i^{\text{eq}}(\mathbf{x},t-\Delta t)\right) \\ &\quad -\bar{\tau}\left(d_t f_i^{\text{eq}}(\mathbf{x},t)+d_t f_i^{\text{eq}}(\mathbf{x},t-\Delta t)\right) \\ &\quad +\bar{\tau}\left(c_{i\alpha}d_\alpha f_i^{\text{eq}}(\mathbf{x},t)-c_{i\alpha}d_\alpha f_i^{\text{eq}}(\mathbf{x},t-\Delta t)\right) \\ &\quad -\bar{\tau}\left(1-\frac{\Delta t}{2\bar{\tau}}\right)\left(\psi_i(\mathbf{x},t)-\psi_i(\mathbf{x},t-\Delta t)\right) \\ &\quad +\Delta t\left(d_t f_i^{\text{eq}}(\mathbf{x},t-\Delta t)+c_{i\alpha}d_\alpha f_i^{\text{eq}}(\mathbf{x},t-\Delta t)\right) \\ &\quad +\mathcal{O}(\Delta t^2). \end{aligned} \tag{C17}$$

With $f(t)-f(t-\Delta t)=\Delta t d_t f(t)+\mathcal{O}(\Delta t^2)$ and $f(t)+f(t-\Delta t)=2f(t)-\Delta t d_t f(t)+\mathcal{O}(\Delta t^2)$, the difference of the equilibrium distribution functions can be written in terms of the moments as

$$\begin{aligned} f_{\bar{i}}^{\text{eq}}(\mathbf{x},t)-f_i^{\text{eq}}(\mathbf{x},t-\Delta t) &= w_i\left(\rho(\mathbf{x},t)-\frac{1}{c_s^2}c_{i\alpha}j_\alpha(\mathbf{x},t)+\frac{1}{2c_s^4}(P_{\alpha\beta}(\mathbf{x},t)\right. \\ &\quad \left.-\rho(\mathbf{x},t)c_s^2\delta_{\alpha\beta})(c_{i\alpha}c_{i\beta}-c_s^2\delta_{\alpha\beta})\right) \\ &\quad -w_i\left(\rho(\mathbf{x},t-\Delta t)+\frac{1}{c_s^2}c_{i\alpha}j_\alpha(\mathbf{x},t-\Delta t)\right) \end{aligned}$$

$$\begin{aligned}
 & + \frac{1}{2c_s^4} (P_{\alpha\beta}(\mathbf{x}, t - \Delta t) \\
 & - \rho(\mathbf{x}, t - \Delta t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} - c_s^2 \delta_{\alpha\beta}) \\
 = & w_i \left(d_t \rho(\mathbf{x}, t) \Delta t - 2 \frac{1}{c_s^2} c_{i\alpha} j_\alpha(\mathbf{x}, t) \right. \\
 & + \frac{1}{c_s^2} c_{i\alpha} d_t j_\alpha(\mathbf{x}, t) \Delta t \\
 & + \left. \frac{1}{2c_s^4} (d_t P_{\alpha\beta}(\mathbf{x}, t) - d_t \rho(\mathbf{x}, t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} \right. \\
 & \left. - c_s^2 \delta_{\alpha\beta}) \Delta t \right) + \mathcal{O}(\Delta t^2).
 \end{aligned}$$

Similarly, their sum becomes

$$\begin{aligned}
 & f_i^{\text{eq}}(\mathbf{x}, t) + f_i^{\text{eq}}(\mathbf{x}, t - \Delta t) \\
 = & w_i \left(\rho(\mathbf{x}, t) - \frac{1}{c_s^2} c_{i\alpha} j_\alpha(\mathbf{x}, t) + \frac{1}{2c_s^4} (P_{\alpha\beta}(\mathbf{x}, t) \right. \\
 & \left. - \rho(\mathbf{x}, t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} - c_s^2 \delta_{\alpha\beta}) \right) \\
 & + w_i \left(\rho(\mathbf{x}, t - \Delta t) + \frac{1}{c_s^2} c_{i\alpha} j_\alpha(\mathbf{x}, t - \Delta t) \right. \\
 & + \left. \frac{1}{2c_s^4} (P_{\alpha\beta}(\mathbf{x}, t - \Delta t) \right. \\
 & \left. - \rho(\mathbf{x}, t - \Delta t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} - c_s^2 \delta_{\alpha\beta}) \right) \\
 = & w_i \left(2\rho(\mathbf{x}, t) - d_t \rho(\mathbf{x}, t) \Delta t - \frac{1}{c_s^2} c_{i\alpha} d_t j_\alpha(\mathbf{x}, t) \Delta t \right. \\
 & + 2 \frac{1}{2c_s^4} (P_{\alpha\beta}(\mathbf{x}, t) - \rho(\mathbf{x}, t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} - c_s^2 \delta_{\alpha\beta}) \\
 & - \left. \frac{1}{2c_s^4} (d_t P_{\alpha\beta}(\mathbf{x}, t) - d_t \rho(\mathbf{x}, t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} \right. \\
 & \left. - c_s^2 \delta_{\alpha\beta}) \Delta t \right) + \mathcal{O}(\Delta t^2).
 \end{aligned}$$

Similar relations hold for the time-derivatives

$$\begin{aligned}
 & d_t f_i^{\text{eq}}(\mathbf{x}, t) - d_t f_i^{\text{eq}}(\mathbf{x}, t - \Delta t) \\
 = & w_i \left(d_t^2 \rho(\mathbf{x}, t) \Delta t - 2 \frac{1}{c_s^2} c_{i\alpha} d_t j_\alpha(\mathbf{x}, t) \right. \\
 & + \frac{1}{c_s^2} c_{i\alpha} d_t^2 j_\alpha(\mathbf{x}, t) \Delta t \\
 & + \left. \frac{1}{2c_s^4} (d_t^2 P_{\alpha\beta}(\mathbf{x}, t) - d_t^2 \rho(\mathbf{x}, t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} \right. \\
 & \left. - c_s^2 \delta_{\alpha\beta}) \Delta t \right) + \mathcal{O}(\Delta t^2), \\
 & d_t f_i^{\text{eq}}(\mathbf{x}, t) + d_t f_i^{\text{eq}}(\mathbf{x}, t - \Delta t) \\
 = & w_i \left(2d_t \rho(\mathbf{x}, t) - d_t^2 \rho(\mathbf{x}, t) \Delta t - \frac{1}{c_s^2} c_{i\alpha} d_t^2 j_\alpha(\mathbf{x}, t) \Delta t \right. \\
 & + \left. 2 \frac{1}{2c_s^4} (d_t P_{\alpha\beta}(\mathbf{x}, t) - d_t \rho(\mathbf{x}, t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} - c_s^2 \delta_{\alpha\beta}) \right)
 \end{aligned}$$

$$\begin{aligned}
 & - \frac{1}{2c_s^4} (d_t^2 P_{\alpha\beta}(\mathbf{x}, t) - d_t^2 \rho(\mathbf{x}, t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} \\
 & - c_s^2 \delta_{\alpha\beta}) \Delta t \Big) + \mathcal{O}(\Delta t^2),
 \end{aligned}$$

and the spatial derivatives

$$\begin{aligned}
 & c_{i\alpha} d_\alpha f_i^{\text{eq}}(\mathbf{x}, t) - c_{i\alpha} d_\alpha f_i^{\text{eq}}(\mathbf{x}, t - \Delta t) \\
 = & w_i \left(c_{i\alpha} d_\alpha d_t \rho(\mathbf{x}, t) \Delta t - 2 \frac{1}{c_s^2} c_{i\alpha} c_{i\beta} d_\alpha j_\beta(\mathbf{x}, t) \right. \\
 & + \frac{1}{c_s^2} c_{i\alpha} c_{i\beta} d_\alpha d_t j_\beta(\mathbf{x}, t) \Delta t \\
 & + \left. \frac{1}{2c_s^4} c_{i\alpha} d_\alpha (d_t P_{\beta\gamma}(\mathbf{x}, t) - d_t \rho(\mathbf{x}, t) c_s^2 \delta_{\beta\gamma}) (c_{i\beta} c_{i\gamma} \right. \\
 & \left. - c_s^2 \delta_{\beta\gamma}) \Delta t \right) + \mathcal{O}(\Delta t^2), \\
 & c_{i\alpha} d_\alpha f_i^{\text{eq}}(\mathbf{x}, t) + c_{i\alpha} d_\alpha f_i^{\text{eq}}(\mathbf{x}, t - \Delta t) \\
 = & w_i \left(2c_{i\alpha} d_\alpha \rho(\mathbf{x}, t) - c_{i\alpha} d_\alpha d_t \rho(\mathbf{x}, t) \Delta t \right. \\
 & - \frac{1}{c_s^2} c_{i\alpha} c_{i\beta} d_t d_\alpha j_\beta(\mathbf{x}, t) \Delta t \\
 & + 2 \frac{1}{2c_s^4} c_{i\alpha} d_\alpha (P_{\beta\gamma}(\mathbf{x}, t) - \rho(\mathbf{x}, t) c_s^2 \delta_{\beta\gamma}) (c_{i\beta} c_{i\gamma} - c_s^2 \delta_{\beta\gamma}) \\
 & - \left. \frac{1}{2c_s^4} c_{i\alpha} d_\alpha (d_t P_{\beta\gamma}(\mathbf{x}, t) - d_t \rho(\mathbf{x}, t) c_s^2 \delta_{\beta\gamma}) (c_{i\beta} c_{i\gamma} \right. \\
 & \left. - c_s^2 \delta_{\beta\gamma}) \Delta t \right) + \mathcal{O}(\Delta t^2).
 \end{aligned}$$

Appendix C.2: Dirchlet boundary conditions

The relations outlined above can be substituted into (C16) to obtain

$$\begin{aligned}
 & \bar{f}_i(\mathbf{x}, t) - \bar{f}_i^{\text{col}}(\mathbf{x}, t - \Delta t) \\
 = & w_i \left(d_t \rho(\mathbf{x}, t) \Delta t - 2 \frac{1}{c_s^2} c_{i\alpha} j_\alpha(\mathbf{x}, t) + \frac{1}{c_s^2} c_{i\alpha} d_t j_\alpha(\mathbf{x}, t) \Delta t \right. \\
 & + \frac{1}{2c_s^4} (d_t P_{\alpha\beta}(\mathbf{x}, t) - d_t \rho(\mathbf{x}, t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} - c_s^2 \delta_{\alpha\beta}) \Delta t \\
 & - \bar{\tau} w_i \left(- 2 \frac{1}{c_s^2} c_{i\alpha} d_t j_\alpha(\mathbf{x}, t) \right) \\
 & + \bar{\tau} w_i \left(2c_{i\alpha} d_\alpha \rho(\mathbf{x}, t) + 2 \frac{1}{2c_s^4} c_{i\alpha} d_\alpha (P_{\beta\gamma}(\mathbf{x}, t) \right. \\
 & \left. - \rho(\mathbf{x}, t) c_s^2 \delta_{\beta\gamma}) (c_{i\beta} c_{i\gamma} - c_s^2 \delta_{\beta\gamma}) \right) \\
 & - \bar{\tau} \left(1 - \frac{\Delta t}{2\bar{\tau}} \right) \left(2\psi_i(\mathbf{x}, t) \right) \\
 & - \Delta t w_i \left(d_t \rho(\mathbf{x}, t) + \frac{1}{c_s^2} c_{i\alpha} d_t j_\alpha(\mathbf{x}, t) \right. \\
 & + \left. \frac{1}{2c_s^4} (d_t P_{\alpha\beta}(\mathbf{x}, t) - d_t \rho(\mathbf{x}, t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} - c_s^2 \delta_{\alpha\beta}) \right)
 \end{aligned}$$

$$\begin{aligned}
 & -\Delta t w_i \left(c_{i\alpha} d_\alpha \rho(\mathbf{x}, t) + \frac{1}{c_s^2} c_{i\alpha} c_{i\beta} d_\alpha j_\beta(\mathbf{x}, t) \right. \\
 & + \frac{1}{2c_s^4} (c_{i\alpha} d_\alpha P_{\beta\gamma}(\mathbf{x}, t) \\
 & - c_{i\alpha} d_\alpha \rho(\mathbf{x}, t) c_s^2 \delta_{\beta\gamma}) \\
 & \left. (c_{i\beta} c_{i\gamma} - c_s^2 \delta_{\beta\gamma}) \right) + \mathcal{O}(\Delta t^2),
 \end{aligned}$$

where the order of appearance of the terms is retained for convenience. Further simplification yields

$$\begin{aligned}
 & \bar{f}_i(\mathbf{x}, t) - \bar{f}_i^{\text{col}}(\mathbf{x}, t - \Delta t) \\
 & = -2w_i \frac{1}{c_s^2} c_{i\alpha} j_\alpha(\mathbf{x}, t) - \Delta t w_i \frac{1}{c_s^2} c_{i\alpha} c_{i\beta} d_\alpha j_\beta(\mathbf{x}, t) \\
 & + 2\bar{\tau} w_i \frac{1}{c_s^2} c_{i\alpha} d_t j_\alpha(\mathbf{x}, t) \\
 & + (2\bar{\tau} - \Delta t) w_i \left(c_{i\alpha} d_\alpha \rho(\mathbf{x}, t) + \frac{1}{2c_s^4} c_{i\alpha} d_\alpha (P_{\beta\gamma}(\mathbf{x}, t) \right. \\
 & - \rho(\mathbf{x}, t) c_s^2 \delta_{\beta\gamma}) (c_{i\beta} c_{i\gamma} - c_s^2 \delta_{\beta\gamma}) \left. \right) \\
 & - (2\bar{\tau} - \Delta t) \psi_i(\mathbf{x}, t) + \mathcal{O}(\Delta t^2). \tag{C18}
 \end{aligned}$$

Substitution into the first-order bounce-back rule

$$\bar{f}_i(\mathbf{x}, t) - \bar{f}_i^{\text{col}}(\mathbf{x}, t - \Delta t) = -\frac{2}{c_s^2} w_i c_{i\alpha} j_\alpha^*,$$

yields

$$\begin{aligned}
 & -\frac{2}{c_s^2} w_i c_{i\alpha} j_\alpha^* = -2w_i \frac{1}{c_s^2} c_{i\alpha} j_\alpha(\mathbf{x}, t) \tag{C19} \\
 & - \Delta t w_i \frac{1}{c_s^2} c_{i\alpha} c_{i\beta} d_\alpha j_\beta(\mathbf{x}, t) \\
 & + 2\bar{\tau} w_i \frac{1}{c_s^2} c_{i\alpha} d_t j_\alpha(\mathbf{x}, t) \\
 & + (2\bar{\tau} - \Delta t) w_i \left(c_{i\alpha} d_\alpha \rho(\mathbf{x}, t) + \frac{1}{2c_s^4} c_{i\alpha} d_\alpha (P_{\beta\gamma}(\mathbf{x}, t) \right. \\
 & - \rho(\mathbf{x}, t) c_s^2 \delta_{\beta\gamma}) (c_{i\beta} c_{i\gamma} - c_s^2 \delta_{\beta\gamma}) \left. \right) \\
 & - (2\bar{\tau} - \Delta t) \psi_i(\mathbf{x}, t) + \mathcal{O}(\Delta t^2). \tag{C20}
 \end{aligned}$$

We therefore obtain, with first-order accuracy

$$-c_{i\alpha} j_\alpha^* = -c_{i\alpha} j_\alpha(\mathbf{x}, t) - \frac{1}{2} \Delta t c_{i\alpha} c_{i\beta} d_\alpha j_\beta(\mathbf{x}, t) + \mathcal{O}(\Delta t).$$

The terms on the right hand-side amount to a second-order spatial Taylor expansion of the term $c_{i\alpha} j_\alpha$ to the boundary, at $\mathbf{x} + \frac{1}{2} \mathbf{c}_i \Delta t$. On the left hand side, $c_{i\alpha} j_\alpha^*$ is an equivalent term, but with the prescribed boundary value $\mathbf{j}^* = \rho_0 d_t \mathbf{u}^*$, i.e.

$$c_{i\alpha} j_\alpha^* = c_{i\alpha} j_\alpha(\mathbf{x} + \frac{1}{2} \mathbf{c}_i \Delta t, t) + \mathcal{O}(\Delta t).$$

For a boundary lattice site with two or more lattice links crossing the boundary in linearly independent directions i , two independent, consistent conditions are imposed on the linear momentum density \mathbf{j} modeled by the LBM at the boundary in the vicinity of the boundary lattice site. This suffices to (consistently) determine the linear momentum density at the boundary, with first-order accuracy, for D2Q9 lattices. The consistency study thus indicates that the unmodified bounce-back rule is consistent with the desired boundary values at $\mathbf{x} + \frac{1}{2} \mathbf{c}_i \Delta t$, with first-order accuracy. Note that, as long as only first-order accuracy is required, this boundary rule can also be used for non-lattice conforming boundaries.

Second-order accuracy with a bounce-back boundary rule could be achieved if the rule were modified so as to cancel the linear terms in Δt which were neglected in (C20).

Appendix C.3: Neumann boundary conditions

The analysis is analogous for the case of Neumann boundaries. The relations derived in appendix C.1 can be substituted into (C17) to yield

$$\begin{aligned}
 & \bar{f}_i(\mathbf{x}, t) + \bar{f}_i^{\text{col}}(\mathbf{x}, t - \Delta t) \\
 & = w_i \left(2\rho(\mathbf{x}, t) - d_t \rho(\mathbf{x}, t) \Delta t - \frac{1}{c_s^2} c_{i\alpha} d_t j_\alpha(\mathbf{x}, t) \Delta t \right. \\
 & + 2\frac{1}{2c_s^4} (P_{\alpha\beta}(\mathbf{x}, t) - \rho(\mathbf{x}, t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} - c_s^2 \delta_{\alpha\beta}) \\
 & - \frac{1}{2c_s^4} (d_t P_{\alpha\beta}(\mathbf{x}, t) - d_t \rho(\mathbf{x}, t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} \\
 & - c_s^2 \delta_{\alpha\beta}) \Delta t \left. \right) \\
 & - \bar{\tau} w_i \left(2d_t \rho(\mathbf{x}, t) + 2\frac{1}{2c_s^4} (d_t P_{\alpha\beta}(\mathbf{x}, t) \right. \\
 & - d_t \rho(\mathbf{x}, t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} - c_s^2 \delta_{\alpha\beta}) \left. \right) \\
 & + \bar{\tau} w_i \left(-2\frac{1}{c_s^2} c_{i\alpha} c_{i\beta} d_\alpha j_\beta(\mathbf{x}, t) \right) \\
 & + \Delta t w_i \left(d_t \rho(\mathbf{x}, t) + \frac{1}{c_s^2} c_{i\alpha} d_t j_\alpha(\mathbf{x}, t) + \frac{1}{2c_s^4} (d_t P_{\alpha\beta}(\mathbf{x}, t) \right. \\
 & - d_t \rho(\mathbf{x}, t) c_s^2 \delta_{\alpha\beta}) (c_{i\alpha} c_{i\beta} - c_s^2 \delta_{\alpha\beta}) \left. \right) \\
 & + \Delta t w_i \left(c_{i\alpha} d_\alpha \rho(\mathbf{x}, t) + \frac{1}{c_s^2} c_{i\alpha} c_{i\beta} d_\alpha j_\beta(\mathbf{x}, t) \right. \\
 & + \frac{1}{2c_s^4} (c_{i\alpha} d_\alpha P_{\beta\gamma}(\mathbf{x}, t) - c_{i\alpha} d_\alpha \rho(\mathbf{x}, t) c_s^2 \delta_{\beta\gamma}) (c_{i\beta} c_{i\gamma} \\
 & - c_s^2 \delta_{\beta\gamma}) \left. \right) + \mathcal{O}(\Delta t^2).
 \end{aligned}$$

Again, the order of the terms is left unchanged for convenience. Upon simplification, this becomes

$$\bar{f}_i(\mathbf{x}, t) + \bar{f}_i^{\text{col}}(\mathbf{x}, t - \Delta t)$$

$$\begin{aligned}
 &= w_i \left(2\rho(\mathbf{x}, t) + 2\frac{1}{2c_s^4} (P_{\alpha\beta}(\mathbf{x}, t) - \rho(\mathbf{x}, t)c_s^2\delta_{\alpha\beta}) \right. \\
 &\quad \left. (c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}) \right) \\
 &\quad + \Delta t w_i \left(c_{i\alpha}d_\alpha\rho(\mathbf{x}, t) + \frac{1}{2c_s^4} (c_{i\alpha}d_\alpha P_{\beta\gamma}(\mathbf{x}, t) \right. \\
 &\quad \left. - c_{i\alpha}d_\alpha\rho(\mathbf{x}, t)c_s^2\delta_{\beta\gamma})(c_{i\beta}c_{i\gamma} - c_s^2\delta_{\beta\gamma}) \right) \\
 &\quad - 2\bar{\tau} w_i \left(d_t\rho(\mathbf{x}, t) + \frac{1}{2c_s^4} (d_t P_{\alpha\beta}(\mathbf{x}, t) - d_t\rho(\mathbf{x}, t)c_s^2\delta_{\alpha\beta}) \right. \\
 &\quad \left. (c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}) \right) \\
 &\quad + (\Delta t - 2\bar{\tau}) w_i \frac{1}{c_s^2} c_{i\alpha}c_{i\beta} d_\alpha j_\beta(\mathbf{x}, t) + \mathcal{O}(\Delta t^2).
 \end{aligned}$$

Substitution into the first-order anti-bounce-back rule

$$\begin{aligned}
 &\bar{f}_i(\mathbf{x}, t) + \bar{f}_i^{\text{col}}(\mathbf{x}, t - \Delta t) \\
 &= 2w_i \left(\rho^{\text{bd}} + \frac{1}{2c_s^4} (P_{\alpha\beta}^* \right. \\
 &\quad \left. - \rho^{\text{bd}}c_s^2\delta_{\alpha\beta})(c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}) \right).
 \end{aligned}$$

yields

$$\begin{aligned}
 &2w_i \left(\rho^{\text{bd}} + \frac{1}{2c_s^4} (P_{\alpha\beta}^* - \rho^{\text{bd}}c_s^2\delta_{\alpha\beta})(c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}) \right) \\
 &= w_i \left(2\rho(\mathbf{x}, t) + 2\frac{1}{2c_s^4} (P_{\alpha\beta}(\mathbf{x}, t) \right. \\
 &\quad \left. - \rho(\mathbf{x}, t)c_s^2\delta_{\alpha\beta})(c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}) \right) \\
 &\quad + \Delta t w_i \left(c_{i\alpha}d_\alpha\rho(\mathbf{x}, t) + \frac{1}{2c_s^4} (c_{i\alpha}d_\alpha P_{\beta\gamma}(\mathbf{x}, t) \right. \\
 &\quad \left. - c_{i\alpha}d_\alpha\rho(\mathbf{x}, t)c_s^2\delta_{\beta\gamma})(c_{i\beta}c_{i\gamma} - c_s^2\delta_{\beta\gamma}) \right) \\
 &\quad - 2\bar{\tau} w_i \left(d_t\rho(\mathbf{x}, t) + \frac{1}{2c_s^4} (d_t P_{\alpha\beta}(\mathbf{x}, t) \right. \\
 &\quad \left. - d_t\rho(\mathbf{x}, t)c_s^2\delta_{\alpha\beta})(c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}) \right) \\
 &\quad + (\Delta t - 2\bar{\tau}) w_i \frac{1}{c_s^2} c_{i\alpha}c_{i\beta} d_\alpha j_\beta(\mathbf{x}, t) + \mathcal{O}(\Delta t^2). \quad (C21)
 \end{aligned}$$

Thus, with a first-order accuracy

$$\begin{aligned}
 &2w_i \left(\rho^{\text{bd}} + \frac{1}{2c_s^4} (P_{\alpha\beta}^* - \rho^{\text{bd}}c_s^2\delta_{\alpha\beta})(c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}) \right) \\
 &= w_i \left(2\rho(\mathbf{x}, t) + 2\frac{1}{2c_s^4} (P_{\alpha\beta}(\mathbf{x}, t) - \rho(\mathbf{x}, t)c_s^2\delta_{\alpha\beta}) \right. \\
 &\quad \left. (c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}) \right) \\
 &\quad + \Delta t w_i \left(c_{i\alpha}d_\alpha\rho(\mathbf{x}, t) + \frac{1}{2c_s^4} (c_{i\alpha}d_\alpha P_{\beta\gamma}(\mathbf{x}, t) \right. \\
 &\quad \left. - c_{i\alpha}d_\alpha\rho(\mathbf{x}, t)c_s^2\delta_{\beta\gamma})(c_{i\beta}c_{i\gamma} - c_s^2\delta_{\beta\gamma}) \right) + \mathcal{O}(\Delta t),
 \end{aligned}$$

and, by an argument analogous to that made in the bounce-back case

$$\begin{aligned}
 &\left(\rho^{\text{bd}} + \frac{1}{2c_s^4} (P_{\alpha\beta}^* - \rho^{\text{bd}}c_s^2\delta_{\alpha\beta})(c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}) \right) \\
 &= \left(\rho \left(\mathbf{x} + \frac{1}{2}\mathbf{c}_i\Delta t, t \right) + \frac{1}{2c_s^4} (P_{\alpha\beta} \left(\mathbf{x} + \frac{1}{2}\mathbf{c}_i\Delta t, t \right) \right. \\
 &\quad \left. - \rho \left(\mathbf{x} + \frac{1}{2}\mathbf{c}_i\Delta t, t \right) c_s^2\delta_{\alpha\beta})(c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}) \right) + \mathcal{O}(\Delta t).
 \end{aligned}$$

The prescribed value for ρ^{bd} is determined by linear extrapolation

$$\begin{aligned}
 \rho^{\text{bd}} &= \frac{1}{2} (3\rho(\mathbf{x}, t) - \rho(\mathbf{x} - \mathbf{c}_i\Delta t, t)) \\
 &= \rho \left(\mathbf{x} + \frac{1}{2}\mathbf{c}_i\Delta t, t \right) + \mathcal{O}(\Delta t^2),
 \end{aligned}$$

which implies

$$\begin{aligned}
 \frac{1}{2c_s^4} P_{\alpha\beta}^* (c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}) &= \frac{1}{2c_s^4} P_{\alpha\beta} \left(\mathbf{x} + \frac{1}{2}\mathbf{c}_i\Delta t, t \right) \\
 (c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}) &+ \mathcal{O}(\Delta t), \quad (C22)
 \end{aligned}$$

as the density terms cancel. In two dimensions, \mathbf{P} features three independent components due to its symmetry. In a coordinate system normal to the boundary, one of these is determined by extrapolation similarly to the density boundary values

$$\begin{aligned}
 P_{22}^{\text{n,bd}} &= \frac{1}{2} (3P_{22}^{\text{n}}(\mathbf{x}, t) - P_{22}^{\text{n}}(\mathbf{x} - \mathbf{c}_i\Delta t, t)) \\
 &= P_{22}^{\text{n}} \left(\mathbf{x} + \frac{1}{2}\mathbf{c}_i\Delta t, t \right) + \mathcal{O}(\Delta t^2).
 \end{aligned}$$

For the anti-bounce-back rule (C14) to be consistent with the prescribed boundary values in the first order, (C22) must impose three independent, consistent conditions on the components of \mathbf{P} . At a boundary lattice site with lattice links i crossing the boundary, the dyadic products $\mathbf{c}_i \otimes \mathbf{c}_i$ must form a basis in the space of symmetric second-order tensors for this to be the case. Note that this condition is satisfied at a boundary lattice site in a D2Q9 lattice, whenever at least three adjacent lattice links cross the boundary.

Second-order accuracy with an anti-bounce-back boundary rule could be achieved if the rule were modified so as to cancel the linear terms in Δt which were neglected in (C21).

Appendix D: Evolution of instabilities

Figure 13 shows the evolution of the displacement magnitude $|\mathbf{u}|$ in the top left corner of the plane strain domain

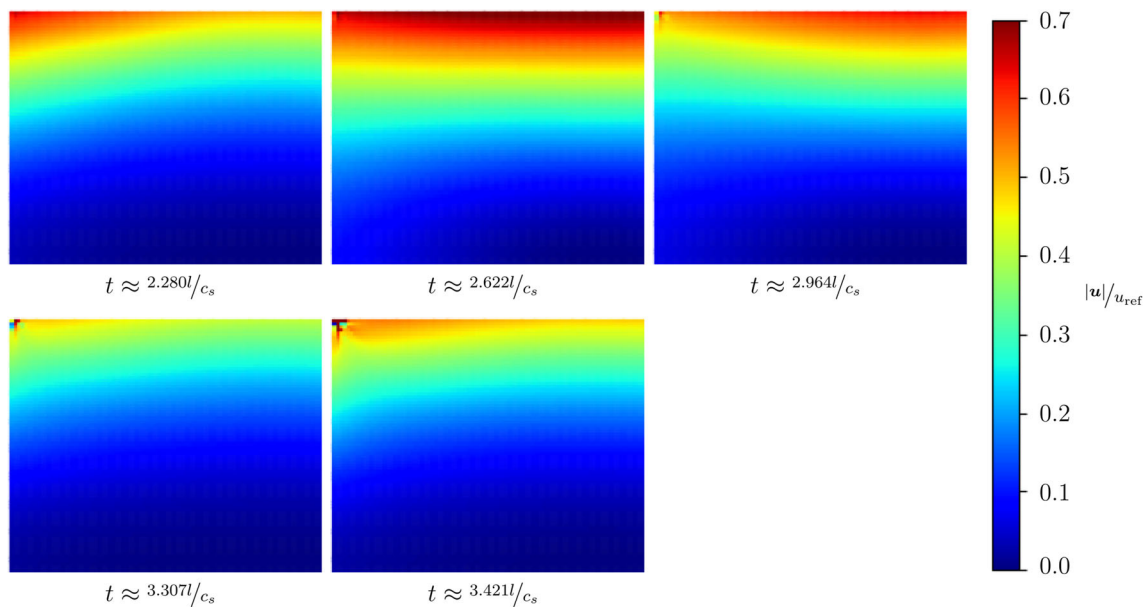


Fig. 13 Contour plot of the displacement magnitude $|u|$ in the top left corner for the example outlined in Sect. 5, at several points in time. It is apparent that instabilities originate near the corner of the rectangular plane strain domain. On grounds of symmetry, only one corner of the domain is shown

investigated in Sect. 5. Instabilities originate near the corner of the domain. As instabilities of this kind already appear in the LB algorithm in the absence of boundaries (see [2]), the source of the instabilities exhibited here is difficult to ascertain. The repeated reflection of incident waves at corners with Neumann boundaries produces waves with short wavelength, which were found in [2] to provoke instabilities in the LB algorithm proposed in [1]. Alternatively, the boundary conditions proposed in this work may introduce a separate source of instabilities. While instabilities in the LB algorithm must be investigated in detail and eventually eliminated for elastodynamic LBM simulations to be viable for application, this is beyond the scope of the present investigation.

References

- Murthy J, Kolluru P, Kumaran V, Ansumali S (2017) Lattice Boltzmann method for wave propagation in elastic solids. *Commun Comput Phys* 2:23. <https://doi.org/10.4208/cicp.OA-2016-0259>
- Escande M, Kolluru PK, Cléon LM, Sagaut P (2020) Lattice Boltzmann method for wave propagation in elastic solids with a regular lattice: theoretical analysis and validation. *arXiv*. doi:1048550/ARXIV.2009.06404 . arXiv:2009.06404
- Freund LB (1990) *Dynamic fracture mechanics*. Cambridge Monographs on Mechanics. Cambridge University Press, Cambridge. <https://doi.org/10.1017/CBO9780511546761>
- Hashemi AS, Katsabanis P (2020) The effect of stress wave interaction and delay timing on blast-induced rock damage and fragmentation. *Rock Mech Rock Eng* 53(5):2327–2346. <https://doi.org/10.1007/s00603-019-02043-9>
- Marconi S, Chopard B (2003) A lattice Boltzmann model for a solid body. *Int J Mod Phys B* 17(01n02):153–156. <https://doi.org/10.1142/S0217979203017254>
- O'Brien GS, Nissen-Meyer T, Bean CJ (2012) A lattice Boltzmann method for elastic wave propagation in a poisson solid. *Bull Seismol Soc Am* 102(3):1224–1234. <https://doi.org/10.1785/0120110191>
- Boolakee O, Geier M, De Lorenzis L (2023) A new lattice Boltzmann scheme for linear elastic solids: periodic problems. *Comput Methods Appl Mech Eng* 404:115756. <https://doi.org/10.1016/j.cma.2022.115756>
- Boolakee O, Geier M, De Lorenzis L. Dirichlet and neumann boundary conditions for a lattice boltzmann scheme for linear elastic solids on arbitrary domains. Available at SSRN 4446582. <https://doi.org/10.2139/ssrn.4446582>
- Landau LD, Lifshitz EM, Kosevich AM, Pitaevskii LP (1986) *Theory of elasticity*, 3rd edn. Butterworth-Heinemann, Oxford. <https://doi.org/10.1016/B978-0-08-057069-3.50003-6>
- Becker W, Gross D (2002) *Mechanik Elastischer Körper und Strukturen*, 1st edn. Springer, Heidelberg. <https://doi.org/10.1007/978-3-642-56124-5>
- Poruchikov VB (1993) *Methods of the classical theory of elastodynamics*, 1st edn. Springer, Heidelberg. <https://doi.org/10.1007/978-3-642-77099-9>
- Krüger T, Kusumaatmaja H, Kuzmin A, Shardt O, Silva G, Viggen EM (2017) *The lattice Boltzmann method: principles and practice*, 1st edn. Springer, Heidelberg. <https://doi.org/10.1007/978-3-319-44649-3>
- Holzappel AG (2000) *Nonlinear solid mechanics: a continuum approach for engineering*, 1st edn. Wiley, Chichester. <https://doi.org/10.1023/A:1020843529530>
- Farag G, Zhao S, Chiavassa G, Boivin P (2021) Consistency study of lattice-Boltzmann schemes macroscopic limit. *Phys Fluids* 33(3):037101. <https://doi.org/10.1063/5.0039490>
- He X, Chen S, Doolen GD (1998) A novel thermal model for the lattice Boltzmann method in incompressible limit. *J Comput Phys* 146(1):282–300. <https://doi.org/10.1006/jcph.1998.6057>

16. Mott PH, Roland CM (2009) Limits to poisson's ratio in isotropic materials. *Phys Rev B* 80:132104. <https://doi.org/10.1103/PhysRevB.80.132104>
17. Taylor RL (2014) FEAP-A finite element analysis program
18. Tschöke K, Gravenkamp H (2018) On the numerical convergence and performance of different spatial discretization techniques for transient elastodynamic wave propagation problems. *Wave Motion* 82:62–85. <https://doi.org/10.1016/j.wavemoti.2018.07.002>
19. Gross D, Seelig T (2016) *Bruchmechanik*, 6th edn. Springer, Heidelberg. <https://doi.org/10.1007/978-3-662-46737-4>
20. Rahman M, Michelitsch T (2006) A note on the formula for the rayleigh wave speed. *Wave Motion* 43(3):272–276. <https://doi.org/10.1016/j.wavemoti.2005.10.002>
21. Chirino F, Dominguez J (1989) Dynamic analysis of cracks using boundary element method. *Eng Fract Mech* 34(5):1051–1061. [https://doi.org/10.1016/0013-7944\(89\)90266-X](https://doi.org/10.1016/0013-7944(89)90266-X)
22. Ginzburg I, d'Humières D (2003) Multireflection boundary conditions for lattice Boltzmann models. *Phys Rev E* 68:066614. <https://doi.org/10.1103/PhysRevE.68.066614>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.