PHILOSOPHICAL
TRANSACTIONS A

royalsocietypublishing.org/journal/rsta

Research

Cite this article: Silva G, Ginzburg I. 2020 Reviving the local second-order boundary approach within the two-relaxation-time lattice Boltzmann modelling. *Phil. Trans. R. Soc. A* **378**: 20190404. http://dx.doi.org/10.1098/rsta.2019.0404

Accepted: 27 March 2020

One contribution of 15 to a theme issue 'Fluid dynamics, soft matter and complex systems: recent results and new methods'.

Subject Areas:
computational physics, mechanical engineering, fluid mechanics

Keywords:
lattice Boltzmann method, two-relaxation-time, boundary conditions, LSOB, porous media

Authors for correspondence:
Goncalo Silva
e-mail: goncalo.nuno.silva@gmail.com
Irina Ginzburg
e-mail: irina.ginzburg@inrae.fr

THE ROYAL SOCIETY
PUBLISHING

Reviving the local second-order boundary approach within the two-relaxation-time lattice Boltzmann modelling

Goncalo Silva[1] and Irina Ginzburg[2]

[1]LAETA, IDMEC, Mechanical Engineering Department, IST, University of Lisbon, 1049-001 Lisbon, Portugal
[2]Université Paris-Saclay, INRAE, UR HYCAR, 92160, Antony, France

GS, 0000-0001-5719-799X; IG, 0000-0002-2660-350X

This work addresses the Dirichlet boundary condition for momentum in the lattice Boltzmann method (LBM), with focus on the steady-state Stokes flow modelling inside non-trivial shaped ducts. For this task, we revisit a local and highly accurate boundary scheme, called the local second-order boundary (LSOB) method. This work reformulates the LSOB within the two-relaxation-time (TRT) framework, which achieves a more standardized and easy to use algorithm due to the pivotal parametrization TRT properties. The LSOB explicitly reconstructs the unknown boundary populations in the form of a Chapman–Enskog expansion, where not only first- but also second-order momentum derivatives are locally extracted with the TRT symmetry argument, through a simple local linear algebra procedure, with no need to compute their non-local finite-difference approximations. Here, two LSOB strategies are considered to realize the wall boundary condition, the original one called Lwall and a novel one Lnode, which operate with the wall and node variables, roughly speaking. These two approaches are worked out for both plane and curved walls, including the corners. Their performance is assessed against well-established LBM boundary schemes such as the bounce-back, the local second-order accurate CLI scheme and two different parabolic multi-reflection (MR) schemes. They are all evaluated for 3D duct flows with rectangular, triangular, circular and annular cross-sections, mimicking the geometrical challenges of real porous structures. Numerical tests confirm that LSOB competes with the parabolic MR accuracy in this problem class, requiring only a single node to operate.

This article is part of the theme issue 'Fluid dynamics, soft matter and complex systems: recent results and new methods'.

1. Introduction

The computational fluid dynamics (CFD) modelling of flows in porous media is relevant in a wide range of applications [1]; from the groundwater movement in aquifers to smaller-scale systems associated with transport in soils. In all cases, the simulation accuracy gets strongly determined by the discretization of the bounding geometry [2]. While the lattice Boltzmann method (LBM) displays numerous advantages as a CFD tool, see [3,4], it formulates on an uniform Cartesian mesh, by default. Therefore, the task of prescribing boundary conditions over arbitrarily shaped surfaces requires particular care in order to avoid the inaccurate staircase discretization. This task is further complicated by the fact that LBM's working variables correspond to mesoscopic populations, rather than macroscopic fields [5]. Thus, the LBM realization of hydrodynamic boundary conditions tends to be made implicitly, through conditions set at the level of the mesoscopic populations. Such difficulties explain the relentless interest in this topic, which continues an active area of research [3,4,6]. The motivation of researchers is common: *to formulate LBM boundary schemes that are simultaneously accurate and easy to use*. These two assets gain even greater relevance in the modelling of porous media flows, typically governed by Stokes flow, and that is bounded by complex and irregular geometries [7–9]. For that reason, below we briefly revise the state of affairs on the LBM implementation of the *no-slip Dirichlet boundary condition*.[1] Broadly speaking, two families of LBM boundary schemes can be identified to realize the no-slip condition; they differ according to their operation principle.

The first strategy is the *link-wise approach* that expresses the unknown incoming populations through a linear combination of the known components, *all restricted to the same link*. This family of boundary schemes is very popular to model Dirichlet conditions in irregular geometries, possibly due to the good trade-off between accuracy and simplicity of implementation, e.g. [11–20]. The hallmark of link-wise boundary schemes is the bounce-back (BB) rule [14,15]. The BB is mostly advocated for its ease of implementation, locality, and exact mass conservation; however, its application is plagued by two noteworthy deficiencies: (i) grid shifted no-slip walls are located within first-order accuracy, and (ii) artificial 'Knudsen-layers' are developed that accommodate, but also may distort, the inner flow solution [13,21,22]. These accuracy limitations have motivated the development of many enhanced link-wise boundary strategies, such as Filipova–Hänel treatment [23], interpolated BB [24], unified interpolated BB [25], or equilibrium interpolation [18]. While sacrificing some of the BB assets, these strategies improve the geometrical fidelity of the, otherwise, staircase approximation of mesh-inclined boundaries. Despite this improvement, the accuracy of the accommodation of flow solutions on walls is still limited to first-order derivative terms, which justifies their labelling as *linear schemes* as opposed to *parabolic schemes*; the terminology proposed in [11] and also used here. Since the flow curvature dominates the low Reynolds flow regimes, the account of the second-order derivatives is therefore crucial in porous flow, and the classical Poiseuille flow clearly demonstrates this standpoint [14]. A related drawback of these linear schemes is that they tend to corrupt the parametrization properties of the bulk solution, e.g. leading to viscosity-dependent numerical errors [24,25]; though some of them were improved [11,18] and implemented locally [11]. The multi-reflection (MR) methods [16] were designed to methodologically improve these limitations for arbitrarily shaped walls. Compared to the accuracy of linear schemes, the 'parabolic' MR methods correctly describe the first- and second-order derivative terms in the accommodation of the flow solution, which is consistent with the order of derivative terms captured by LBM in bulk. The parabolic level of accuracy is

[1] Other types of boundary conditions in the LBM framework can be found in [10–12] for further details on this topic.

also supported by the MLI schemes [11] that capture the flow curvature through the directional velocity Laplacian explicit construction; these 'parabolic' schemes model exactly arbitrary grid rotated Poiseuille flow with their two-point boundary rule. While the two offer the highest 'parabolic accuracy' on various problem classes, e.g. [10,26–28], they deal with the shortcomings: (i) non-locality; (ii) modification of the scheme on edge/corner nodes (missing next directional neighbour); and (iii) difficulty to independently prescribe normal/tangential conditions [11,12].

The second strategy is the *node-based approach* where the boundary node populations are split into equilibrium and non-equilibrium parts, and this latter is approximated by the content predicted by the Chapman–Enskog expansion [29]. Although these two points are the same as with the MR rules, in the node-based approach the Chapman–Enskog approximation is applied differently and simultaneously for all populations *lying on the same node*, but pertaining to different links. Since its shear-stress components are only required for the formal derivation of the Navier–Stokes equation, the Chapman–Enskog series is commonly truncated after the first-order derivatives, following the finite-difference reconstruction of Skordos [30] or moments-based approach of Noble [31]. Contemporary to these works, and originally motivated by the intention to match the grid aligned (straight or diagonal) BB Poiseuille solution exactly, the second-order derivatives were included into the Chapman–Enskog reconstruction of the boundary populations, both analytically [14] and numerically [32]. The application of Skordos or Noble approaches is presently more popular on simple straight in-node walls that are coincident with the boundary nodes, e.g. [33–40]. Nonetheless, their extension to operate on more complex boundary shapes has also been devised, e.g. [41–46]. The first-order derivatives appearing there are usually computed through finite difference stencils [43,45], but exceptions exist, such as [36,40,42,47], which perform this task by locally extracting the first-order derivatives from other known mesoscopic populations. Overall, these approaches pertaining to the node-based class share the same level of accuracy as the linear schemes; a result that has been confirmed by several numerical studies, e.g. [42–47]. In practice, a formal equivalent accuracy is achievable with the local implementation of 'linear' MR schemes [11], while the finite-difference approximations involve a larger stencil. Contrary to these approaches, the local LSOB reconstruction [21,48] also includes the second-order derivatives into: (i) the population expansion, following [14]; and (ii) the Taylor expansion along the normal direction between the grid and the wall, following [32]. The distinctive point is that the LSOB derives the two sets locally, from the stress and the so-called 'ghost' moments, respectively. Thus, the LSOB differs from the other LBM boundary schemes, either link-wise or node-based, in that it supports the parabolic accuracy in a local manner, either at planar [21] and/or curved [48] surfaces. The LSOB idea was further adapted in [41] for a moving free-interface front based on the first-order Chapman–Enskog expansion expressed in interface-aligned, rotated frame; more recently, a downgraded version of LSOB, truncated at first order, but extended to Navier–Stokes equilibrium just like in [41], was devised by [47], which implements the no-slip condition on curved walls within the LBM-BGK framework. Although, in our experience, the linear (local) MR rules tend to be more stable than the parabolic (two-point) schemes, according to the reports in [7,36,43,45], the non-local finite-difference reconstruction of the first-order gradient, e.g. [43,45], appears to be more stable than the local one, e.g. [42,47]. Such a seeming contradiction appears to confirm the heuristic stability analysis of the MR rules; that is, rather than the stencil, it is the scheme coefficients range that determines the MR stability. Finally, we note that the operation principle of LSOB and MR methodologies retains the known populations untouched and only prescribes the unknown ones, as opposed to, e.g. 'regularized' in-node approaches [36]; the rationale behind the procedure followed here is that the unreconstructed, high-order kinetic component of the conserved populations shares the bulk truncation, parametrization and may better serve for invisible accommodation [13,49].

Therefore, the present work intends to revive the long time developed LSOB method [21,48], by taking a modern look at it. The following three main tasks are undertaken. First, we devise a more transparent, easy to use formulation of the original LSOB methodology, by re-expressing it in the TRT symmetrized structure for the $d3Q19$ lattice. Second, we resolve some issues not fully contemplated previously, such as the construction of the LSOB linear algebra problem and its effect on the method's support of the viscosity-independent numerical errors [22]. Third and finally, we compare the LSOB performance against several link-wise schemes, from the simplest BB and local linear CLI scheme [11] to the two most accurate MR families [11,16]. The common point for the LBM boundary schemes considered herein is that they support viscosity-independent numerical errors: this property is not automatic even with the parabolic boundary rules according to analysis [11,22] but, on top of the physical consistency, it is the only reasonable way for comparison between their steady-state accuracies. Here, as the first level of verification, we study the simulation of 3D duct flows featuring different cross-sections [50]. This problem class is interesting as analytical solutions are available for benchmarking in several non-trivial geometries [51]. Moreover, pipes are the elemental building blocks of generic porous media configurations, which is the application goal of this work. Finally, we note that the formulations and the tests presented here complement the exact application of LSOB to grid inclined 2D channels early studied [21]. The two works cover the different configuration scenarios expected in the discretization of no-slip boundaries in pipes.

2. The lattice Boltzmann method two-relaxation-time model: overview

The LBM [3–5] solves for the populations $f_q(\boldsymbol{x},t)$, defined on space $\boldsymbol{x}$ and time t, along a discrete velocity set, called the lattice, with one immobile $\boldsymbol{c}_0=\boldsymbol{0}$ and $Q_m=Q-1$ non-zero velocity vectors $\boldsymbol{c}_q$ per grid node. The LBM evolves through a succession of streaming and collision steps. Here, the collision step is the two-relaxation-time (TRT) model [11] formulated on the symmetry argument that any lattice quantity ψ_q can be decomposed onto symmetric and anti-symmetric components as $\psi_q^{\pm}=\frac{1}{2}(\psi_q \pm \psi_{\bar{q}})$, where $\boldsymbol{c}_{\bar{q}}=-\boldsymbol{c}_q$. Overall, the LBM-TRT evolution rule reads

$$f_q(\boldsymbol{x}+\boldsymbol{c}_q, t+1)=\widehat{f_q}(\boldsymbol{x},t), \quad q=0,1,\dots,Q-1, \tag{2.1a}$$

$$\widehat{f_q}(\boldsymbol{x},t)=[f_q+\hat{n}_q^+ +\hat{n}_q^- + S_q](\boldsymbol{x},t), \quad q=0,1,\dots,\frac{Q_m}{2}, \tag{2.1b}$$

$$\widehat{f_{\bar{q}}}(\boldsymbol{x},t)=[f_{\bar{q}}+\hat{n}_q^+ -\hat{n}_q^- - S_q](\boldsymbol{x},t), \quad q=1,\dots,\frac{Q_m}{2}. \tag{2.1c}$$

The *post-collision* state $\widehat{f_q}$ in (2.1) includes the external force source $S_q=t_q^{\star}\, c_{q\alpha}\, F_{\alpha}$ and the non-equilibrium post-collision components $\hat{n}_q^{\pm}$, which are given by $\hat{n}_q^{\pm}:=-n_q^{\pm}/\tau^{\pm}$ with: (i) non-equilibrium $n_q^{\pm}:=(f_q^{\pm}-e_q^{\pm})$, where $e_q^{\pm}$ denotes the (symmetric/anti-symmetric) equilibrium components, and (ii) relaxation times $\tau^{\pm}$ of the (symmetric/anti-symmetric) collision modes. With this parity decomposition introduced, the *post-stream* populations f_q expressed at generic space/time location $(\boldsymbol{x},t)$ are written as follows:

$$f_q(\boldsymbol{x},t)=\left[e_q^+ + e_q^- - \tau^+\,\hat{n}_q^+ - \tau^-\,\hat{n}_q^-\right](\boldsymbol{x},t). \tag{2.2}$$

In hydrodynamic models, τ^+ determines the fluid kinematic viscosity, as $\nu=\frac{1}{3}(\tau^+-\frac{1}{2})$, thanks to the adoption of the hydrodynamic weights $t_q^{\star}$ [52] (defined below), while τ^- is a free tunable parameter. The two define the key relaxation collision parameter $\Lambda=(\tau^+-\frac{1}{2})(\tau^- -\frac{1}{2})$, which controls the stationary field of non-dimensional TRT solutions [22] at exact discrete level, i.e. beyond the second-order hydrodynamic limit. The LBM-BGK collision model [52] is retrieved for $\tau^+=\tau^-=\tau$, which unavoidably recovers $\Lambda=9\nu^2$.

This work focuses on the simulation of slow flows, modelled by linear Stokes hydrodynamics

$$\partial_{\alpha}P-F_{\alpha}=\nu\,\partial_{\beta\beta}j_{\alpha}, \quad \partial_{\alpha}j_{\alpha}=0. \tag{2.3}$$

Hereinafter the summation convention applies for all d-dimensional vectors, denoted by Greek letters. The hydrodynamic parameters in (2.3) consider: P pressure (or ρ mass density[2]), j_{α} flow

[2] With the equation of state $P=c_s^2\,\rho$, where $c_s^2\in]0,1[$ is a free lattice-dependent parameter in a thermal models [19,20].

momentum[3] and F_α body force density. They are determined by the following discrete velocity moments $\rho = \sum_{q=0}^{Q-1} f_q$, $j_\alpha = \sum_{q=1}^{Q-1} f_q\, c_{q\alpha} + \frac{1}{2}\, F_\alpha$ and $F_\alpha = \sum_{q=1}^{Q-1} S_q\, c_{q\alpha}$, where $\alpha = 1, \dots, d$.

Stokes flow equations are reproduced in LBM-TRT with the prescription of the $e_q^{\pm}$ equilibrium

$$e_q^+ = t_q^\star\, P(\rho), \quad e_q^- = t_q^\star\, c_{q\alpha} \left(j_\alpha - \frac{1}{2}\, F_\alpha \right), \tag{2.4}$$

where $t_q^\star$ are isotropic lattice weights, see [4,16,52], which are $t_q^\star = \{t_c, t_d\} = \{1/6, 1/12\}$, for, respectively, the first (coordinate) and second (diagonal) links for the $d3Q19$ model. The whole equilibrium is recovered as $e_q = e_q^+ + e_q^-$. We emphasize that the quadratic velocity term [53,54] and its possible associated force correction are not required for the Stokes flow modelling.

The non-equilibrium part of populations $n_q = f_q - e_q$ can be accessed through the *second-order Chapman–Enskog expansion* [16], which unfolds n_q through the perturbation parameter ϵ, defined as the ratio of the lattice unit over a characteristic length scale L, i.e. $\epsilon = 1/L$. This expansion reads $n_q = \epsilon\, n_q^{(1)} + \epsilon^2\, n_q^{(2)} + \mathcal{O}(\epsilon^3)$, which transcribed to TRT framework leads to $n_q = n_q^+ + n_q^- = -\tau^+\, \hat{n}_q^+ - \tau^-\, \hat{n}_q^-$, implying that each term of the approximation obeys $\hat{n}_q^+ \sim n_q^{+(1)}$ and $\hat{n}_q^- \sim n_q^{-(2)}$ in flow problems [11]. This relation permits writing the content of $\hat{n}_q^{\pm}$, up to $\mathcal{O}(\epsilon^3)$ in the modelling of laminar incompressible flow, as follows [4,11,16,21]:

$$\hat{n}_q^+ = t_q^\star\, c_{q\alpha} c_{q\beta}\, \partial_\beta\, j_\alpha \tag{2.5a}$$

and

$$\hat{n}_q^- = -\nu\, t_q^\star \left(3 c_{q\alpha} c_{q\beta} c_{q\gamma} - c_{q\alpha}\, \delta_{\beta\gamma} - c_{q\beta}\, \delta_{\alpha\gamma} - c_{q\gamma}\, \delta_{\alpha\beta}\right) \partial_{\beta\gamma}\, j_\alpha. \tag{2.5b}$$

The substitution of (2.4) and (2.5) into (2.2) permits the explicit reconstruction of the TRT populations, up to $\mathcal{O}(\epsilon^3)$ within the Chapman–Enskog scaling, as follows:

$$\begin{aligned} f_q = t_q^\star\, P + t_q^\star\, c_{q\alpha} \left(j_\alpha - \frac{1}{2}\, F_\alpha \right) - \tau^+\, t_q^\star\, c_{q\alpha} c_{q\beta} \partial_\beta\, j_\alpha \\ + \tau^-\, \nu\, t_q^\star \left(3 c_{q\alpha} c_{q\beta} c_{q\gamma} - c_{q\alpha}\, \delta_{\beta\gamma} - c_{q\beta}\, \delta_{\alpha\gamma} - c_{q\gamma}\, \delta_{\alpha\beta}\right) \partial_{\beta\gamma}\, j_\alpha. \end{aligned} \tag{2.6}$$

Remark. The form of (2.6) in multiple-relaxation-time (MRT) hydrodynamic models [20,41] only differs with respect to the symmetric (viscous-stress) component, whenever bulk and kinematic viscosities are computed with different eigenvalues. Yet, for incompressible flow models, the projection on the bulk viscosity eigenvector vanishes due to the zero divergence value (see appendix A in [41]), then no change is required, with: (i) $\hat{n}_q^+$ determined as in TRT or through summation over the viscous-tensor post-collision projections, and (ii) $\hat{n}_q^-$ determined as in TRT or through summation over all post-collision projections on the odd-order polynomial eigenvectors (i.e. third- and fifth-order in $d3Q19$ lattice). The extension of the reconstruction (2.6) to weakly compressible Navier–Stokes MRT models may follow [41].

3. Local second-order boundary method

In standard LBM, the geometry is discretized on an uniform Cartesian mesh, with the computational nodes identified in three types. (i) *Solid nodes* are sites outside the fluid domain (ii) *Fluid nodes* are sites inside the fluid region, where LBM, (2.1), applies; (iii) *Boundary nodes* $\boldsymbol{x}_b$ are sites also belonging to the fluid region, but with at least one link connected to the outside solid domain, i.e. $\boldsymbol{x}_b + \boldsymbol{c}_q \in$ solid. At $\boldsymbol{x}_b$ we cannot apply (2.1), as the populations entering from outside the fluid region are unknown. Hence, it is useful to separate the populations $f(\boldsymbol{x}_b)$ into two sets:

$$\mathcal{K} = \{q \,|\, f_q \text{ is known}\} \quad \text{and} \quad \mathcal{U} = \{q \,|\, f_q \text{ is unknown}\}, \tag{3.1}$$

with $\dim(\mathcal{K}) + \dim(\mathcal{U}) = Q$. In reality, rather than Q, only Q_m populations are considered in this work, as focus is given to steady flows where the rest population f_0 shall play an irrelevant role when reaching its equilibrium.

[3] In this work, momentum and velocity are regarded as interchangeable quantities, based on $j_\alpha = \rho_0\, u_\alpha$ with $\rho_0 = 1$, e.g. [34].

The purpose of the boundary scheme is to assign the boundary populations within the set $\mathcal{U}$. This work revives the LSOB approach [21,48], employing a $d3Q19$ cubic lattice [52,55] in lieu of the $d4Q24$ hypercubic lattice, originally used in [21,48], which greatly modernizes and simplifies the formulation. For illustrative purposes, LSOB is explained through the modelling of fully-developed flows within *channel-like geometries of non-trivial cross-section* [51], as is common with porous media flows. Let us introduce the 3D Cartesian coordinate system (x, y, z) to describe the unidirectional flow $\boldsymbol{j} = j_x\,\boldsymbol{e}_x$, driven by constant body-force $\boldsymbol{F} = F_x\,\boldsymbol{e}_x$ and/or pressure-gradient $\nabla P = \partial_x P\,\boldsymbol{e}_x$, which undergoes negligible (or zero in periodic case) streamwise variations $\nabla j_x \cdot \boldsymbol{e}_x = \partial_x j_x = 0$, so that $j_x(y, z)$ is effectively the only momentum unknown. By applying these simplifications at boundary node $\boldsymbol{x}_b$, the $\hat{n}_q^{\pm}$ presented in (2.5) reduces to

$$\hat{n}_q^{+}(\boldsymbol{x}_b) = t_q^{\star}\,c_{qx}\,\left[c_{qy}\,\partial_y\,j_x(\boldsymbol{x}_b) + c_{qz}\,\partial_z\,j_x(\boldsymbol{x}_b)\right] \tag{3.2a}$$

and

$$\hat{n}_q^{-}(\boldsymbol{x}_b) = -\nu\,t_q^{\star}\,c_{qx}\left[\left(3c_{qy}^2 - 1\right)\partial_{yy}\,j_x(\boldsymbol{x}_b) + \left(3c_{qz}^2 - 1\right)\partial_{zz}\,j_x(\boldsymbol{x}_b)\right], \tag{3.2b}$$

where $q = 1, \ldots, Q_m/2$. Similarly, the populations f_q, in full form given by (2.6), reduce at $\boldsymbol{x}_b$ to:

$$\begin{aligned} f_q(\boldsymbol{x}_b) = t_q^{\star}\,P(\boldsymbol{x}_b) + t_q^{\star}\,c_{qx}\Bigg(j_x(\boldsymbol{x}_b) - \frac{1}{2}F_x - \tau^{+}\,\left[c_{qy}\,\partial_y j_x(\boldsymbol{x}_b) + c_{qz}\,\partial_z j_x(\boldsymbol{x}_b)\right] \\ + \tau^{-}\,\nu\left[\left(3c_{qy}^2 - 1\right)\partial_{yy} j_x(\boldsymbol{x}_b) + \left(3c_{qz}^2 - 1\right)\partial_{zz} j_x(\boldsymbol{x}_b)\right]\Bigg). \end{aligned} \tag{3.3}$$

The reconstruction of $f_q(\boldsymbol{x}_b)$, with $q \in \mathcal{U}$, requires specifying the hydrodynamic fields. The non-equilibrium fields in (3.3) require the prescription of the momentum derivatives. However, instead of resorting to finite differences, e.g. [30,32,35,43,45], the unknown derivatives in (3.3) are directly accessed through the $\hat{n}_q^{\pm}$ relations in (3.2), which can be made available[4] for any set $\mathcal{K}$ due to the TRT decomposition: $\hat{n}_q^{+} = \hat{n}_{\bar{q}}^{+}$ and $\hat{n}_q^{-} = -\hat{n}_{\bar{q}}^{-}$ for $\boldsymbol{c}_q = -\boldsymbol{c}_{\bar{q}}$; this avoids looking for particular subsets of populations, considerably simplifying the original methodology [21,48], where the momentum derivatives are accessed by selecting specific subsets within $\{f_q\}$ and/or $\{\widehat{f_q}\}$ for each $q \in \mathcal{K}$, which unavoidably produces a case by case formulation in irregular walls.

The boundary node equilibrium momentum $j_x(\boldsymbol{x}_b)$ in (3.3) is also unknown. We can use two approximate methods to relate $j_x(\boldsymbol{x}_b)$ to the available wall information. These two methods, called Lnode and Lwall, are explained below.

(i) The Lnode procedure, first proposed in the unpublished work [48], obtains the momentum in (3.3) from its known value at the wall location $\boldsymbol{x}_w$ by means of the *second-order* Taylor series expansion

$$j_x(\boldsymbol{x}_b) = j_x(\boldsymbol{x}_w) + \delta_n\,\partial_n j_x(\boldsymbol{x}_b) - \frac{\delta_n^2}{2}\,\partial_{nn} j_x(\boldsymbol{x}_b). \tag{3.4}$$

(ii) The Lwall procedure, originally developed in [21], approximates both the momentum in (3.3) and the momentum derivatives in (3.3) and (3.2) from the wall location $\boldsymbol{x}_w$ towards the boundary node $\boldsymbol{x}_b$ through the *second-order* Taylor series expansions

$$\left.\begin{aligned} & j_x(\boldsymbol{x}_b) = j_x(\boldsymbol{x}_w) + \delta_n\,\partial_n j_x(\boldsymbol{x}_w) + \frac{\delta_n^2}{2}\,\partial_{nn} j_x(\boldsymbol{x}_w) \\ & \partial_\beta j_x(\boldsymbol{x}_b) = \partial_\beta j_x(\boldsymbol{x}_w) + \delta_n \partial_{n\beta} j_x(\boldsymbol{x}_w) \\ & \partial_{n\beta} j_x(\boldsymbol{x}_b) = \partial_{n\beta} j_x(\boldsymbol{x}_w), \end{aligned}\right\} \tag{3.5}$$

where the bound indices β are taken in $\{y, z\}$, and $\delta_n = (\boldsymbol{x}_b - \boldsymbol{x}_w) \cdot \boldsymbol{n}$, where $\delta_n \geq 0$ represents the distance between the boundary node and the wall location, with $\boldsymbol{n}$ the wall normal unit vector directed into the fluid domain.

[4] Except in corner nodes, where the access to $\hat{n}_q^{\pm}$ is limited, as explained in §4.

When pressure is unknown, e.g. when it cannot be set to an arbitrary constant, one can locally determine its boundary value, $P(\boldsymbol{x}_b)$, by adapting the algorithms [21,34]. The idea is to reformulate the zeroth-order mass moment as follows $P/c_s^2 = \sum_{q\in\mathcal{K}} f_q + \sum_{q\in\mathcal{U}} f_q$ at $\boldsymbol{x}_b$ and then replace the f_q from the $\mathcal{U}$ set using (2.2); the result reads:

$$P(\boldsymbol{x}_b) = \frac{1}{c_s^{-2} - \sum_{q\in\mathcal{U}} t_q^\star}\left[\sum_{q\in\mathcal{K}} f_q + (j_x - \tfrac{1}{2}F_x)\sum_{q\in\mathcal{U}} t_q^\star c_{qx} - \tau^+ \sum_{q\in\mathcal{U}} \hat{n}_q^+ - \tau^- \sum_{q\in\mathcal{U}} \hat{n}_q^-\right](\boldsymbol{x}_b), \tag{3.6}$$

with $j_x(\boldsymbol{x}_b)$ in (3.6) approximated by equation (3.4) in Lnode or by (3.5) in Lwall procedures. Given that the Taylor series approximation of $j_x(\boldsymbol{x}_b)$ requires the knowledge of the momentum derivatives, the $P(\boldsymbol{x}_b)$ variable is computed after these derivative terms are determined.

With the momentum derivatives and pressure at $\boldsymbol{x}_b$ obtained, the boundary populations $f_q(\boldsymbol{x}_b)$, with $q\in\mathcal{U}$, cf. (3.3), can be finally reconstructed, closing the LBM boundary value problem. The LSOB algorithm in both Lnode and Lwall procedures is summarized in appendix A. Next, in §4 and §5, we will illustrate how LSOB (via Lnode or Lwall) imposes the Dirichlet momentum condition on plane and curved walls, respectively.

4. Application of local second-order boundary to plane walls

Plane surfaces are conveniently described with *Cartesian coordinate systems*. We consider fixed (y, z) and rotated (y', z') frames, with $y' = y\,\cos\theta + z\,\sin\theta$ and $z' = -y\,\sin\theta + z\,\cos\theta$ parallel to wall tangent and wall normal vectors, respectively. Next, Lnode and Lwall methods are described.

(a) Lnode on plane walls

In Lnode, it is convenient to work on the fixed coordinate frame (y, z). By taking the inward unit normal vector to the plane wall, given by $\boldsymbol{n} = n_{z'}\,\boldsymbol{e}_{z'}$, and defining $\delta_{z'} = \delta_n\, n_{z'}$ with $\delta_n \in [0, 1]$, then the Taylor series approximation of the momentum $j_x(\boldsymbol{x}_b)$, given by (3.4), applies as follows:

$$\begin{aligned} j_x(\boldsymbol{x}_b) &= j_x(\boldsymbol{x}_w) + \delta_{z'}\,\partial_{z'} j_x(\boldsymbol{x}_b) - \frac{\delta_{z'}^2}{2}\,\partial_{z'z'} j_x(\boldsymbol{x}_b)\\ &= j_x(\boldsymbol{x}_w) + \delta_{z'}\,\big(-\sin\theta\,\partial_y j_x(\boldsymbol{x}_b) + \cos\theta\,\partial_z j_x(\boldsymbol{x}_b)\big)\\ &\quad - \frac{\delta_{z'}^2}{2}\left(\sin^2\theta\,\partial_{yy} j_x(\boldsymbol{x}_b) + \cos^2\theta\,\partial_{zz} j_x(\boldsymbol{x}_b) - \sin 2\theta\,\partial_{yz} j_x(\boldsymbol{x}_b)\right). \end{aligned} \tag{4.1}$$

By inserting (4.1) into (3.3), the unknown boundary populations, $f_q(\boldsymbol{x}_b)$ with $q\in\mathcal{U}$, read

$$\begin{aligned} f_q(\boldsymbol{x}_b) = t_q^\star\, P(\boldsymbol{x}_b) + t_q^\star\, c_{qx}\, &\Bigg(j_x(\boldsymbol{x}_w) - \frac{1}{2}\,F_x\\ &+ \left[-\delta_{z'}\sin\theta - \tau^+\, c_{qy}\right]\partial_y j_x(\boldsymbol{x}_b) + \left[\delta_{z'}\cos\theta - \tau^+\, c_{qz}\right]\partial_z j_x(\boldsymbol{x}_b)\\ &+ \left[-\frac{\delta_{z'}^2}{2}\sin^2\theta + \tau^-\,\nu(3c_{qy}^2 - 1)\right]\partial_{yy} j_x(\boldsymbol{x}_b)\\ &+ \left[-\frac{\delta_{z'}^2}{2}\cos^2\theta + \tau^-\,\nu(3c_{qz}^2 - 1)\right]\partial_{zz} j_x(\boldsymbol{x}_b) + \left[\frac{\delta_{z'}^2}{2}\sin 2\theta\right]\partial_{yz} j_x(\boldsymbol{x}_b)\Bigg). \end{aligned} \tag{4.2}$$

The unknown momentum derivatives in (4.2) are $\mathcal{X} = \{\partial_y j_x, \partial_z j_x, \partial_{yy} j_x, \partial_{zz} j_x, \partial_{yz} j_x\}^\mathsf{T}$. In Lnode to determine $\mathcal{X}$, attention must be paid to the types of boundary nodes arising from the discretization; they can be distinguished as *regular*, *singular* and *corner* nodes, see figure 1.

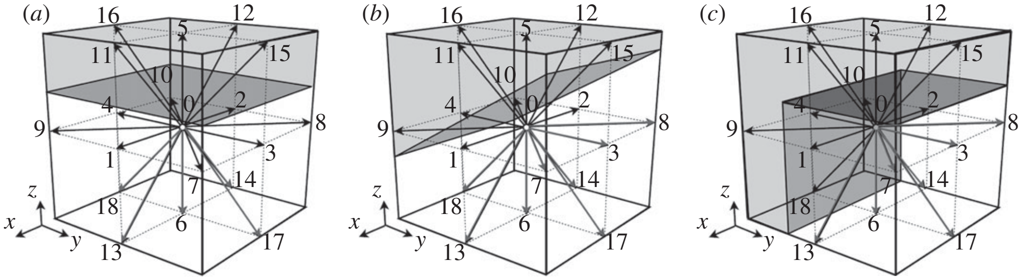

Figure 1. Examples of boundary node types for different wall discretizations on the *d*3*Q*19 computational cell with discrete velocities $\in \mathcal{U}$ marked in light grey color. Here, we illustrate for walls co-planar with the *x*-coordinate axis, but other rotated orientations are supported by the LSOB method (while not covered here examples are found in [21]). (*a*) Regular boundary node [dim($\mathcal{U}$) = 5]; (*b*) Singular boundary node [dim($\mathcal{U}$) = 8]; (*c*) Corner boundary node [dim($\mathcal{U}$) = 7].

Regular boundary nodes, characterized by rank $[\mathbf{M}] \geq \dim(\mathcal{X})$, so that (3.2) applies directly as

$$\underbrace{\begin{bmatrix}\hat{n}_q^+\\ \hat{n}_q^-\end{bmatrix}}_{\mathcal{N}} = \underbrace{\begin{bmatrix} t_q^\star\, c_{qx}\, c_{qy} & t_q^\star\, c_{qx}\, c_{qz} & 0 & 0 & 0\\ 0 & 0 & -\nu\, t_q^\star\, c_{qx}\,(3c_{qy}^2-1) & -\nu\, t_q^\star\, c_{qx}\,(3c_{qz}^2-1) & 0\end{bmatrix}}_{[\mathbf{M}]} \underbrace{\begin{bmatrix}\partial_y j_x\\ \partial_z j_x\\ \partial_{yy} j_x\\ \partial_{zz} j_x\\ \partial_{yz} j_x\end{bmatrix}}_{\mathcal{X}}.$$

We note that, in regular nodes, the $\mathcal{N}$ set can be formed by any two sub-sets of linearly independent elements in $\hat{n}_q^\pm$, as both are always available due to the TRT property: $\hat{n}_q^+ = \hat{n}_{\bar{q}}^+$ and $\hat{n}_q^- = -\hat{n}_{\bar{q}}^-$ for $\boldsymbol{c}_q = -\boldsymbol{c}_{\bar{q}}$. Let us illustrate its application for a *straight horizontal wall*, which aligns with the underlying mesh, as shown in figure 1*a*. In this case, $\theta = 0$, the unknown *d*3*Q*19 populations reduce to those where $c_{qx}c_{qz} \neq 0$, $c_{qy} = 0$ in (4.2), and we need only three derivatives $\{\partial_z j_x, \partial_{yy} j_x, \partial_{zz} j_x\}$. We can then reduce the $\mathcal{N}$ set to the minimal system composed of only two linearly independent populations, using the set $\mathcal{K} = \{{'c'}, {'d'}\}$ where '*c*' denotes any coordinate population $c_{qx} \neq 0$, $c_{qy} = 0$ with $t_c^\star$ and '*d*' denotes any known diagonal population $c_{qx}c_{qz} \neq 0$, $c_{qy} = 0$ with $t_d^\star$. Inserting this information into equation (3.2), the reduced system is given by:[5]

$$\underbrace{\begin{bmatrix}\hat{n}_c^+/t_c^\star\\ \hat{n}_d^+/t_d^\star\\ \hat{n}_c^-/t_c^\star\\ \hat{n}_d^-/t_d^\star\end{bmatrix}}_{\mathcal{N}} = \underbrace{\begin{bmatrix}0 & 0 & 0\\ c_{dx}c_{dz} & 0 & 0\\ 0 & \nu c_{cx} & \nu c_{cx}\\ 0 & \nu c_{dx} & -2\nu c_{dx}\end{bmatrix}}_{[\mathbf{M}]} \underbrace{\begin{bmatrix}\partial_z j_x(\boldsymbol{x}_b)\\ \partial_{yy} j_x(\boldsymbol{x}_b)\\ \partial_{zz} j_x(\boldsymbol{x}_b)\end{bmatrix}}_{\mathcal{X}}.$$

The computation of $\mathcal{X} = [\mathbf{M}]^{-1}\,\mathcal{N}$ yields

$$\mathcal{X} = \begin{cases} \partial_z j_x(\boldsymbol{x}_b) = \dfrac{\hat{n}_d^+}{t_d^\star c_{dx} c_{dz}}\\[2ex] \partial_{yy} j_x(\boldsymbol{x}_b) = \dfrac{2\hat{n}_c^-}{3\nu t_c^\star c_{cx}} + \dfrac{\hat{n}_d^-}{3\nu t_d^\star c_{dx}}\\[2ex] \partial_{zz} j_x(\boldsymbol{x}_b) = \dfrac{\hat{n}_c^-}{3\nu t_c^\star c_{cx}} - \dfrac{\hat{n}_d^-}{3\nu t_d^\star c_{dx}}. \end{cases} \tag{4.3}$$

Then, $\mathcal{X} = [\mathbf{M}]^{-1}\,\mathcal{N}$ permits closing $f_q(\boldsymbol{x}_b)$ with $q \in \mathcal{U}$ in (4.2). While in this example $[\mathbf{M}]^{-1}$ represents the inverse of a square matrix $[\mathbf{M}]$, in more general cases, e.g. when employing all

[5] The advantage of the Lwall approach is that we could equivalently employ all Q_m known values $\hat{n}_q^\pm$ and solve for the pseudo-inverse of the full system, e.g. through SVD, as explained in appendix A.

Q_m known values $\hat{n}_q^{\pm}$ into set $\mathcal{N}$, then $[\mathbf{M}]$ becomes a rectangular matrix so that $[\mathbf{M}]^{-1}$ represents the pseudo-inverse matrix, cf. appendix A.

Singular boundary nodes correspond to degenerated cases arising from particular discretizations, where unknown populations pertain to the three Cartesian planes of the $d3Q19$ lattice, see figure 1*b*. These cases are characterized by rank $[\mathbf{M}] < \dim(\mathcal{X})$. Here, the exclusive use of $\hat{n}_q^{\pm}$ is not sufficient to solve the system, requiring additional conditions to increase the number of known elements of $\mathcal{N}$. This is done [48] by including physically based constraints, readily available by differentiating the Dirichlet boundary condition along the wall. When limited to a uniform value on the flat wall, these conditions read

$$\begin{cases} \partial_{y'} j_x(\boldsymbol{x}_w) & = \partial_{y'} j_x(\boldsymbol{x}_b) - \delta_{z'}\, \partial_{y'z'} j_x(\boldsymbol{x}_b) \\ & = \left(\cos\theta\, \partial_y j_x(\boldsymbol{x}_b) + \sin\theta\, \partial_z j_x(\boldsymbol{x}_b)\right) \\ & \quad -\delta_{z'} \left(\tfrac{1}{2}\sin 2\theta\, (-\partial_{yy} j_x(\boldsymbol{x}_b) + \partial_{zz} j_x(\boldsymbol{x}_b)) + \cos 2\theta\, \partial_{yz} j_x(\boldsymbol{x}_b)\right) = 0 \\ \partial_{y'y'} j_x(\boldsymbol{x}_w) & = \cos^2\theta\, \partial_{yy} j_x(\boldsymbol{x}_b) + \sin^2\theta\, \partial_{zz} j_x(\boldsymbol{x}_b) + \sin 2\theta\, \partial_{yz} j_x(\boldsymbol{x}_b) = 0 \end{cases} \tag{4.4}$$

Combining (3.2) with the additional constraints (4.4), the singular node system reads as

$$\underbrace{\begin{bmatrix} \hat{n}_q^{+} \\ \hat{n}_q^{-} \\ 0 \\ 0 \end{bmatrix}}_{\mathcal{N}_h} = \underbrace{\begin{bmatrix} t_q^{\star}\, c_{qx}\, c_{qy} & t_q^{\star}\, c_{qx}\, c_{qz} & 0 & 0 & 0 \\ 0 & 0 & -\nu\, t_q^{\star}\, c_{qx}\, (3c_{qy}^2 - 1) & -\nu\, t_q^{\star}\, c_{qx}\, (3c_{qz}^2 - 1) & 0 \\ \cos\theta & \sin\theta & \tfrac{1}{2}\delta_{z'}\, \sin 2\theta & -\tfrac{1}{2}\delta_{z'}\, \sin 2\theta & -\delta_{z'} \cos 2\theta \\ 0 & 0 & \cos^2\theta & \sin^2\theta & \sin 2\theta \end{bmatrix}}_{[\mathbf{M_h}]} \underbrace{\begin{bmatrix} \partial_y j_x \\ \partial_z j_x \\ \partial_{yy} j_x \\ \partial_{zz} j_x \\ \partial_{yz} j_x \end{bmatrix}}_{\mathcal{X}},$$

where the vector $\mathcal{N}_h$ is now formed from Q_m known $\hat{n}_q^{\pm}$ post-collision components plus two constraints. The system inversion provides the elements of $\mathcal{X}$ uniquely, which permits closing (4.2).[6] Note, the inclusion of these constraints should be limited to singular nodes; applied to regular nodes they artificially over-constrain the system, compromising the solution's existence.

Corner boundary nodes, characterized by possessing outgoing boundary populations that are cut by two different walls. In the corner discretization, special type links arise where populations only travel across solid nodes, so-called *buried links* [4,33,37]. This limitation can be circumvented, and allow corner nodes to run as regular ones, through the use of the following two procedures. First, we develop the $j_x(\boldsymbol{x}_b)$ approximation in (4.1) individually for each wall of the corner, according to $\boldsymbol{n} = n_{z'}\, \boldsymbol{e}_{z'}$. Second, we form two systems, like (3.2), each of them composed of different $\hat{n}_q^{\pm}$ elements inside $\mathcal{N}$, and combine them so that, although the size of each $\mathcal{N}$ subset is *not* Q_m, as the two systems share the common unknowns $\mathcal{X}$ they can be solved together as done in regular nodes. Then, the reconstruction of $f_q(\boldsymbol{x}_b)$ with $q \in \mathcal{U}$ at corner nodes follows (4.2) again.

Remark. These three types of node configurations describe the main situations encountered in the discretization of duct flow. We may expect that more complicated cases, e.g. isolated nodes surrounded by walls bounding all directions except along one link, can be addressed in the same fashion by combining the 'singular' and 'corner' systems, namely by adding the surface-known information for all adjacent walls. Otherwise, the combination of different approaches, e.g. LSOB and MR, using equivalent or even degraded accuracy, is not forbidden in very particular cases, as was exemplified in [11,12].

(b) Lwall on plane walls

The Lwall fully operates with the rotated coordinate frame (y', z'). By taking the inward unit normal vector to the plane wall, given by $\boldsymbol{n} = n_{z'}\, \boldsymbol{e}_{z'}$, and defining $\delta_{z'} = \delta_n\, n_{z'}$ with $\delta_n \in [0, 1]$, then

[6] The hypothesis that additional constraints introduced in special situations are sufficient for the uniqueness of the solution of the global LBM equation is not yet confirmed or disproved in a rigorous manner, though it is verified in all considered cases.

the Taylor series approximations in (3.5) along $\boldsymbol{e}_{z'}$ from $\boldsymbol{x}_w$ towards $\boldsymbol{x}_b$, i.e. $\boldsymbol{x}_b = \boldsymbol{x}_w + \delta_{z'}\,\boldsymbol{e}_{z'}$, are represented as follows:

$$\begin{cases} j_x(\boldsymbol{x}_b) = j_x(\boldsymbol{x}_w) + \delta_{z'}\,\partial_{z'} j_x(\boldsymbol{x}_w) + \frac{\delta_{z'}^2}{2}\,\partial_{z'z'} j_x(\boldsymbol{x}_w) \\ \partial_{y'} j_x(\boldsymbol{x}_b) = \cancel{\partial_{y'} j_x(\boldsymbol{x}_w)} + \delta_{z'}\,\partial_{y'z'} j_x(\boldsymbol{x}_w) \\ \partial_{z'} j_x(\boldsymbol{x}_b) = \partial_{z'} j_x(\boldsymbol{x}_w) + \delta_{z'}\,\partial_{z'z'} j_x(\boldsymbol{x}_w) \\ \partial_{y'y'} j_x(\boldsymbol{x}_b) = \cancel{\partial_{y'y'} j_x(\boldsymbol{x}_w)} \\ \partial_{z'z'} j_x(\boldsymbol{x}_b) = \partial_{z'z'} j_x(\boldsymbol{x}_w) \end{cases} \tag{4.5}$$

Above, $\partial_{y'} j_x(\boldsymbol{x}_w) = 0$ and $\partial_{y'y'} j_x(\boldsymbol{x}_w) = 0$ assume the uniformity of the flow momentum over the wall tangent plane; these types of hydrodynamic-based conditions are easily generalized for any surface tangential velocity. They are intrinsic to the Lwall formulation [21] as it operates with derivatives expressed at $\boldsymbol{x}_w$, as shown in (4.5). This contrasts with the Lnode, where the information regarding the momentum derivatives on the wall is only brought up at singular nodes, via constraints (4.4).

By re-writing (3.3) in the rotated coordinate system (y', z') and then by substituting the expansions given by (4.5), $f_q(\boldsymbol{x}_b)$ with $q \in \mathcal{U}$, are constructed as follows:

$$\begin{aligned} f_q(\boldsymbol{x}_b) = t_q^\star\, P(\boldsymbol{x}_b) + t_q^\star\, c_{qx} \Bigg(& j_x(\boldsymbol{x}_w) - \frac{1}{2} F_x + \left[\delta_{z'} - \tau^+\, c_{qz'}\right] \partial_{z'} j_x(\boldsymbol{x}_w) \\ & + \left[\frac{\delta_{z'}^2}{2} - \tau^+\, c_{qz'}\delta_{z'} + \tau^-\, \nu \left((3c_{qy}^2 - 1)\sin^2\theta + (3c_{qz}^2 - 1)\cos^2\theta \right) \right] \partial_{z'z'} j_x(\boldsymbol{x}_w) \\ & + \left[-\tau^+\, c_{qy'}\delta_{z'} + \tau^-\, 3\,\nu \left(c_{qz}^2 - c_{qy}^2 \right) \sin 2\theta \right] \partial_{y'z'} j_x(\boldsymbol{x}_w) \Bigg), \end{aligned} \tag{4.6}$$

where $c_{qy'} = c_{qy}\cos\theta + c_{qz}\sin\theta$ and $c_{qz'} = -c_{qy}\sin\theta + c_{qz}\cos\theta$.

The unknown momentum derivatives in (4.6) form the set $\mathcal{X} = \{\partial_{z'} j_x, \partial_{z'z'} j_x, \partial_{y'z'} j_x\}$. Compared to Lnode, the size of $\mathcal{X}$ in Lwall is two elements smaller. As in Lnode, also in Lwall the elements of $\mathcal{X}$ are accessed from $\hat{n}_q^\pm \subseteq \mathcal{N}$. The difference is that the momentum derivatives in $\mathcal{N}$ are also Taylor expanded from $\boldsymbol{x}_b$ to $\boldsymbol{x}_w$, in accordance with (4.5). Hence, in Lwall the $\mathcal{N}$ set reads by inserting equation (3.5) into equation (3.2) and accounting for zero surface derivatives:

$$\hat{n}_q^+(\boldsymbol{x}_b) = t_q^\star\, c_{qx} \left[c_{qy'}\, \delta_{z'}\, \partial_{y'z'} j_x(\boldsymbol{x}_w) + c_{qz'} \left(\partial_{z'} j_x(\boldsymbol{x}_w) + \delta_{z'}\, \partial_{z'z'} j_x(\boldsymbol{x}_w) \right) \right] \tag{4.7a}$$

and

$$\begin{aligned} \hat{n}_q^-(\boldsymbol{x}_b) = -\nu\, t_q^\star\, c_{qx} \Big[& \left((3c_{qy}^2 - 1)\sin^2\theta + (3c_{qz}^2 - 1)\cos^2\theta \right) \partial_{z'z'} j_x(\boldsymbol{x}_w) \\ & + 3\left(c_{qz}^2 - c_{qy}^2 \right) \sin 2\theta\, \partial_{y'z'} j_x(\boldsymbol{x}_w) \Big]. \end{aligned} \tag{4.7b}$$

In practice, the crucial step of the Lwall approach is that, to determine $\mathcal{X}$ based on the linear algebra problem $\mathcal{X} = [\mathbf{M}]^{-1}\,\mathcal{N}$, the number and type of post-collision components $\hat{n}_q^\pm$ going into $\mathcal{N}$ should be chosen judiciously; though the process is much simpler than [35], since 'forbidden combinations' are obvious here. Two plane wall variants can be distinguished:

(i) *Mesh aligned plane walls:* consider again the *straight horizontal wall* where $\mathcal{X} = \{\partial_z j_x, \partial_{zz} j_x\}$; here, to determine $\mathcal{X}$ we just need one diagonal population with $c_{dx} c_{dz} \neq 0$ to form $\mathcal{N}$. Inserting $\theta = 0$ and $c_{dy} = 0$ into (4.7) we get:

$$\underbrace{\begin{bmatrix} \hat{n}_d^+ / t_d^\star \\ \hat{n}_d^- / t_d^\star \end{bmatrix}}_{\mathcal{N}} = \underbrace{\begin{bmatrix} c_{dx} c_{dz} & \delta_z\, c_{dx} c_{dz} \\ 0 & -2\nu\, c_{dx} \end{bmatrix}}_{[\mathbf{M}]} \underbrace{\begin{bmatrix} \partial_z j_x(\boldsymbol{x}_w) \\ \partial_{zz} j_x(\boldsymbol{x}_w) \end{bmatrix}}_{\mathcal{X}}$$

The computation of $\mathcal{X} = [\mathbf{M}]^{-1}\mathcal{N}$ provides the momentum derivatives at $\boldsymbol{x}_w$

$$\mathcal{X} = \begin{cases} \partial_z j_x(\boldsymbol{x}_w) = \dfrac{\hat{n}_d^+}{t_d^\star c_{dx} c_{dz}} + \delta_z \dfrac{\hat{n}_d^-}{2\nu t_d^\star c_{dx}}, \\ \\ \partial_{zz} j_x(\boldsymbol{x}_w) = -\dfrac{\hat{n}_d^-}{2\nu t_d^\star c_{dx}}. \end{cases} \tag{4.8}$$

(ii) *Arbitrarily inclined plane walls:* $\mathcal{X} = \{\partial_{z'} j_x, \partial_{z'z'} j_x, \partial_{y'z'} j_x\}$. The three elements of $\mathcal{X}$ can be uniquely determined via $\mathcal{X} = [\mathbf{M}]^{-1}\mathcal{N}$ with recourse to the two linearly independent $\hat{n}_q^\pm$ components inside $\mathcal{N}$, each pertaining to the links $c_{qx}c_{qy'} \neq 0$ and $c_{qx}c_{qz'} \neq 0$.

The prescription of *corner boundary nodes* in Lwall is similar to Lnode. But in Lwall, two independent systems are constructed that determine two $\mathcal{X}$ solutions, rather than a common $\mathcal{X}$ solution as in Lnode. This separation helps distinguish the components of $\mathcal{X}$ at different walls, which is necessary to avoid possible ambiguities from the wall-oriented rotated system (y', z'). Then, the reconstruction of $f_q(\boldsymbol{x}_b)$ with $q \in \mathcal{U}$ applies (4.6), but individually for each of the two $\mathcal{X}$.

5. Application of local second-order boundary to curved walls

In the description of curved surfaces, it is convenient to adopt a curvilinear coordinate system. Here, we exemplify the formulation of Lnode and Lwall for a Cylindrical system (r, θ), with $y = r\cos\theta$ and $z = r\sin\theta$, centred at middle pipe (y_0, z_0); generalizations to other cases are simple.

(a) Lnode on curved walls

Similarly to the plane wall case, it is convenient to work on the fixed coordinate frame (y, z). By taking the inward unit normal vector to the curved wall, given by $\boldsymbol{n} = n_r\,\boldsymbol{e}_r$, and defining $\delta_r = n_r\,\delta_n$ with $\delta_n \in [0, 1]$, the Taylor series approximation of the momentum $j_x(\boldsymbol{x}_b)$, given by (3.4), applies as follows:

$$\begin{aligned} j_x(\boldsymbol{x}_b) &= j_x(\boldsymbol{x}_w) + \delta_r\,\partial_r j_x(\boldsymbol{x}_b) - \frac{\delta_r^2}{2}\,\partial_{rr} j_x(\boldsymbol{x}_b) \\ &= j_x(\boldsymbol{x}_w) + \delta_r \left(1 + \frac{\delta_r}{2r}\right)\left(\cos\theta\,\partial_y j_x(\boldsymbol{x}_b) + \sin\theta\,\partial_z j_x(\boldsymbol{x}_b)\right) \\ &\quad - \frac{\delta_r^2}{2}\left(\partial_{yy} j_x(\boldsymbol{x}_b) + \partial_{zz} j_x(\boldsymbol{x}_b)\right) + \cancel{\frac{\delta_r^2}{2r^2}\,\partial_{\theta\theta} j_x(\boldsymbol{x}_b)}, \end{aligned} \tag{5.1}$$

where $r = \sqrt{(y_b - y_0)^2 + (z_b - z_0)^2}$ is the radial location of boundary nodes, and $\partial_{\theta\theta} j_x(\boldsymbol{x}_b) = 0$ follows from the uniform surface momentum condition, (5.3). Then, by inserting (5.1) into (3.3), the unknown boundary populations, $f_q(\boldsymbol{x}_b)$ with $q \in \mathcal{U}$, are reconstructed as follows:

$$\begin{aligned} f_q(\boldsymbol{x}_b) = t_q^\star\,P(\boldsymbol{x}_b) + t_q^\star\,c_{qx}\Bigg(& j_x(\boldsymbol{x}_w) - \frac{1}{2}F_x \\ &+ \left[\delta_r\left(1 + \frac{\delta_r}{2r}\right)\cos\theta - \tau^+ c_{qy}\right]\partial_y j_x(\boldsymbol{x}_b) + \left[\delta_r\left(1 + \frac{\delta_r}{2r}\right)\sin\theta - \tau^+ c_{qz}\right]\partial_z j_x(\boldsymbol{x}_b) \\ &+ \left[-\frac{\delta_r^2}{2} + \tau^-\,\nu(3c_{qy}^2 - 1)\right]\partial_{yy} j_x(\boldsymbol{x}_b) + \left[-\frac{\delta_r^2}{2} + \tau^-\,\nu(3c_{qz}^2 - 1)\right]\partial_{zz} j_x(\boldsymbol{x}_b)\Bigg). \end{aligned} \tag{5.2}$$

The unknown momentum derivatives in (5.2) are $\mathcal{X} = \{\partial_y j_x, \partial_z j_x, \partial_{yy} j_x, \partial_{zz} j_x\}^\intercal$. As it happened in the plane wall case, cf. §4a, also in the curved walls the discretization gives rise to different types of nodes, namely: regular and singular nodes, which need to be treated differently.

Regular boundary nodes are treated the same way for plane or curved walls. The same applies to *singular boundary nodes*. The only difference is that the extra constraints, instead of given by (4.4), apply the uniformity of the fluid momentum over the curved tangent surface [48]

$$\left\{\begin{aligned} \partial_\theta j_x(\boldsymbol{x}_w) &= \partial_\theta j_x(\boldsymbol{x}_b) - \delta_r\, \partial_{\theta r} j_x(\boldsymbol{x}_b)\\ &= (r-\delta_r)(-\sin\theta\, \partial_y j_x(\boldsymbol{x}_b) + \cos\theta\, \partial_z j_x(\boldsymbol{x}_b))\\ &\quad - r\,\delta_r \left(\tfrac{1}{2}\sin 2\theta(-\partial_{yy} j_x(\boldsymbol{x}_b) + \partial_{zz} j_x(\boldsymbol{x}_b)) + \cos 2\theta\, \partial_{yz} j_x(\boldsymbol{x}_b)\right) = 0\\ \\ \partial_{\theta\theta} j_x(\boldsymbol{x}_w) &= -r\, \left(\cos\theta\, \partial_y j_x(\boldsymbol{x}_b) + \sin\theta\, \partial_z j_x(\boldsymbol{x}_b)\right)\\ &\quad + r^2 \left(\sin^2\theta\, \partial_{yy} j_x(\boldsymbol{x}_b) + \cos^2\theta\, \partial_{zz} j_x(\boldsymbol{x}_b) - \sin 2\theta\, \partial_{yz} j_x(\boldsymbol{x}_b)\right) = 0 \end{aligned}\right. \tag{5.3}$$

Note, for *singular boundary nodes*, our unknowns are $\mathcal{X} = \{\partial_y j_x, \partial_z j_x, \partial_{yy} j_x, \partial_{zz} j_x, \partial_{yz} j_x\}^\intercal$. We conclude with two notes on alternatives to (5.3) on curved walls.

Remark 5.1. It is possible to replace the two constraints in (5.3) by the single constraint: $\partial_\theta j_x(\boldsymbol{x}_b) = -\sin\theta\, \partial_y j_x(\boldsymbol{x}_b) + \cos\theta\, \partial_z j_x(\boldsymbol{x}_b) = 0$, which explores the circumferential invariance of momentum. Performance-wise, Lnode with this constraint or with (5.3) yields roughly matching accuracies. Yet, although this single constraint leads to a slightly simpler formulation, it offers a much limited range of applicability, as the condition $\partial_\theta j_x(\vec{x}_b) = 0$ is only valid in axisymmetric flows.

Remark 5.2. Alternatively, we may explore the flow field axisymmetry and re-construct $f_q(\boldsymbol{x}_b)$ directly on the cylindrical coordinate system (r, θ). While this is expected to greatly reduce the number of unknowns to $\mathcal{X} = \{\partial_r j_x, \partial_{rr} j_x\}^\intercal$, this idea (also explored in other LSOB inspired works, e.g. [47]) is not recommended since it corrupts the TRT parametrization structure, leading to the inception of viscosity-dependent errors. The reason is due to the structure of the f_q populations in LBM, which by default rest on a Cartesian coordinate system. Hence, their mapping from a Cartesian to a Cylindrical coordinate system, as only covering terms up to second-order within the Chapman–Enskog scaling, inevitably conducts to a mismatch with respect to the other (unaccounted) higher-order terms.

(b) Lwall on curved walls

The Lwall strategy on curved walls does not benefit from the adoption of a rotated frame (y', z') (or any specific curvilinear system as discussed previously in Remark 2 of §5a). Therefore, we keep the fixed Cartesian frame (y, z) on curved walls. By taking the inward unit normal vector to the curved wall, given by $\boldsymbol{n} = n_r\, \boldsymbol{e}_r$, and defining $\delta_r = n_r\, \delta_n$ with $\delta_n \in [0, 1]$, the Taylor series approximations in (3.5) along $\boldsymbol{e}_r$ from $\boldsymbol{x}_w$ towards $\boldsymbol{x}_b$, i.e. $\boldsymbol{x}_b = \boldsymbol{x}_w + \delta_n\, \boldsymbol{e}_r$, are given by

$$\left\{\begin{aligned} j_x(\boldsymbol{x}_b) &= j_x(\boldsymbol{x}_w) + \delta_r\, \partial_r j_x(\boldsymbol{x}_w) + \frac{\delta_r^2}{2}\, \partial_{rr} j_x(\boldsymbol{x}_w)\\ &= j_x(\boldsymbol{x}_w) + \delta_r \left(1 - \tfrac{\delta_r}{2R}\right) \left(\cos\theta\, \partial_y j_x(\boldsymbol{x}_w) + \sin\theta\, \partial_z j_x(\boldsymbol{x}_w)\right)\\ &\quad + \frac{\delta_r^2}{2}\, \left(\partial_{yy} j_x(\boldsymbol{x}_w) + \partial_{zz} j_x(\boldsymbol{x}_w)\right) - \cancel{\tfrac{\delta_r^2}{2R^2}\, \partial_{\theta\theta} j_x(\boldsymbol{x}_w)}\\ \partial_y j_x(\boldsymbol{x}_b) &= \partial_y j_x(\boldsymbol{x}_w) + \delta_r\, \partial_{ry} j_x(\boldsymbol{x}_w)\\ &= \partial_y j_x(\boldsymbol{x}_w) + \delta_r\, \left(\cos\theta\, \partial_{yy} j_x(\boldsymbol{x}_w) + \sin\theta\, \partial_{yz} j_x(\boldsymbol{x}_w)\right)\\ \partial_z j_x(\boldsymbol{x}_b) &= \partial_z j_x(\boldsymbol{x}_w) + \delta_r\, \partial_{rz} j_x(\boldsymbol{x}_w)\\ &= \partial_z j_x(\boldsymbol{x}_w) + \delta_r\, \left(\cos\theta\, \partial_{yz} j_x(\boldsymbol{x}_w) + \sin\theta\, \partial_{zz} j_x(\boldsymbol{x}_w)\right)\\ \partial_{yy} j_x(\boldsymbol{x}_b) &= \partial_{yy} j_x(\boldsymbol{x}_w)\\ \partial_{zz} j_x(\boldsymbol{x}_b) &= \partial_{zz} j_x(\boldsymbol{x}_w), \end{aligned}\right. \tag{5.4}$$

where $R = \sqrt{(y_w - y_0)^2 + (z_w - z_0)^2}$ is the radius of circular wall surface, and the assumption $\partial_{\theta\theta} j_x(\boldsymbol{x}_b) = 0$ is justified on the basis of (5.3). Note that, unlike the plane wall case, the formulation of Lwall on curved surfaces does not reduce the size of $\mathcal{X}$ as conditions $\partial_\theta j_x(\boldsymbol{x}_w) = 0$ and $\partial_{\theta\theta} j_x(\boldsymbol{x}_w) = 0$ do not fit naturally into the content of f_q populations, as noted at the end of §5a.

By substituting the expansions given by (5.4) into (3.3), $f_q(\boldsymbol{x}_b)$ with $q \in \mathcal{U}$ read as:

$$
\begin{aligned}
f_q(\boldsymbol{x}_b) = t_q^\star\, P(\boldsymbol{x}_b) + t_q^\star\, c_{qx} \Bigg(& j_x(\boldsymbol{x}_w) - \frac{1}{2} F_x \\
& + \left[\delta_r \left(1 - \frac{\delta_r}{2R}\right) \cos\theta - \tau^+\, c_{qy}\right] \partial_y j_x(\boldsymbol{x}_w) + \left[\delta_r \left(1 - \frac{\delta_r}{2R}\right) \sin\theta - \tau^+\, c_{qz}\right] \partial_z j_x(\boldsymbol{x}_w) \\
& + \left[\frac{\delta_r^2}{2} - \tau^+\, \delta_r\, c_{qy}\, \cos\theta + \tau^-\, \nu\, (3c_{qy}^2 - 1)\right] \partial_{yy} j_x(\boldsymbol{x}_w) \\
& + \left[\frac{\delta_r^2}{2} - \tau^+\, \delta_r\, c_{qz}\, \sin\theta + \tau^-\, \nu\, (3c_{qz}^2 - 1)\right] \partial_{zz} j_x(\boldsymbol{x}_w) \\
& + \left[-\tau^+\, \delta_r\, (c_{qy}\, \sin\theta + c_{qz}\, \cos\theta)\right] \partial_{yz} j_x(\boldsymbol{x}_w) \Bigg).
\end{aligned} \tag{5.5}
$$

The unknown momentum derivatives in (5.5) are $\mathcal{X} = \{\partial_y j_x,\ \partial_z j_x,\ \partial_{yy} j_x,\ \partial_{zz} j_x,\ \partial_{yz} j_x\}^\mathsf{T}$, which are accessed from $\hat{n}_q^\pm \subseteq \mathcal{N}$ in the Lwall frame as follows:

$$
\begin{aligned}
\hat{n}_q^+(\boldsymbol{x}_b) = t_q^\star\, c_{qx} \Bigg[& \left(c_{qy}\, \partial_y j_x(\boldsymbol{x}_w) + c_{qz}\, \partial_z j_x(\boldsymbol{x}_w)\right) + \delta_r \Big(c_{qy} \cos\theta\, \partial_{yy} j_x(\boldsymbol{x}_w) \\
& + c_{qz} \sin\theta\, \partial_{zz} j_x(\boldsymbol{x}_w) + \left(c_{qy} \sin\theta + c_{qz} \cos\theta\right)\, \partial_{yz} j_x(\boldsymbol{x}_w) \Big) \Bigg]
\end{aligned} \tag{5.6a}
$$

and

$$
\hat{n}_q^-(\boldsymbol{x}_b) = -\nu\, t_q^\star\, c_{qx} \left[\left(3c_{qy}^2 - 1\right) \partial_{yy} j_x(\boldsymbol{x}_w) + \left(3c_{qz}^2 - 1\right) \partial_{zz} j_x(\boldsymbol{x}_w) \right]. \tag{5.6b}
$$

Then, by inserting (5.4) into (5.5), we reconstruct $f_q(\boldsymbol{x}_b)$ with $q \in \mathcal{U}$; here, two notes are in order.

Remark 1. On curved walls, $\dim(\mathcal{X})$ is identical in Lwall and in Lnode. Thus, the application of Lwall on curved walls needs to distinguish between *regular* and *singular boundary nodes* as well. In singular nodes, the set $\mathcal{N}$, originally given by (5.6), is enlarged with two extra constraints. This way, we form the augmented system $\mathcal{N}_h = [\mathbf{M_h}]\, \mathcal{X}$. These extra constraints invoke the uniformity of the fluid momentum over the curved tangent surface, applying (5.3) to $\boldsymbol{x}_w$, as follows:

$$
\begin{cases}
\partial_\theta j_x(\boldsymbol{x}_w) & = R\,(-\sin\theta\, \partial_y j_x(\boldsymbol{x}_w) + \cos\theta\, \partial_z j_x(\boldsymbol{x}_w)) = 0 \\
\partial_{\theta\theta} j_x(\boldsymbol{x}_w) & = -R\, \left(\cos\theta\, \partial_y j_x(\boldsymbol{x}_w) + \sin\theta\, \partial_z j_x(\boldsymbol{x}_w)\right) \\
& \quad + R^2 \left(\sin^2\theta\, \partial_{yy} j_x(\boldsymbol{x}_w) + \cos^2\theta\, \partial_{zz} j_x(\boldsymbol{x}_w) - \sin 2\theta\, \partial_{yz} j_x(\boldsymbol{x}_w)\right) = 0.
\end{cases} \tag{5.7}
$$

Remark 2. In *regular boundary nodes* the set $\mathcal{N}$, originally given by (5.6), can be further enriched by the inclusion of the individual constraint $\partial_{\theta\theta} j_x(\boldsymbol{x}_w) = 0$, featuring in (5.7). This will replace the standard system $\mathcal{N} = [\mathbf{M}]\, \mathcal{X}$ with $\mathcal{N}_h = [\mathbf{M_h}]\, \mathcal{X}$, where only $\partial_{\theta\theta} j_x(\boldsymbol{x}_w) = 0$ in (5.7) is considered; we call this extension the 'half-augmented' system. While both standard and 'half-augmented' systems converge to an unique solution, our numerical tests suggest that the solution accuracy of the latter tends to be slightly enhanced.

6. Numerical results

The numerical performance of the LSOB, formulated as Lnode or Lwall, is compared against other, more well established, LBM link-wise boundary schemes, namely: (i) the BB, due to its popularity; (ii) the central linear interpolation (CLI) scheme [11], as an example of a linear accurate scheme, (iii) the multireflection MR1 scheme [16] and (iv) the MCLI scheme [11], as examples of parabolic accurate schemes. Their explicit algorithms are given in appendix B. These choices represent a trade-off between accuracy and implementation simplicity; in common, they all support viscosity-independent numerical errors. Concerning their implementation effort, both BB and CLI operate locally on one node, while MR1 and MCLI can be implemented with two nodes [11]. In terms of accuracy, taking grid shifted walls as an example, BB is typically $\mathcal{O}(\epsilon)$,

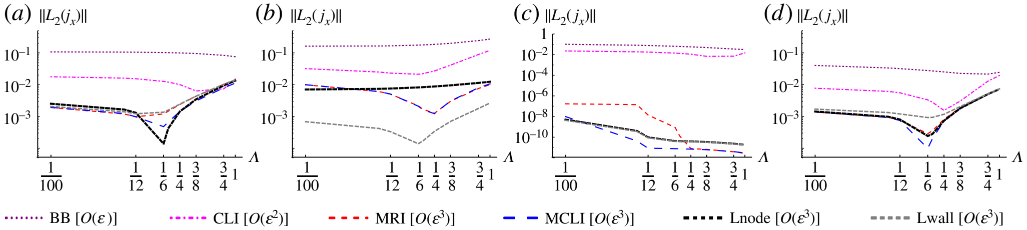

Figure 2. $\|L_2(j_x)\|$ versus Λ with fixed grid resolution $N=14$, for each pipe flow configuration: (*a*) rectangular duct flow $N=2H=2W/\zeta$; (*b*) triangular duct flow $N=h$; (*c*) circular pipe flow $N=R$; (*d*) annular pipe flow $N=R_2-R_1$. (Online version in colour.)

CLI is $\mathcal{O}(\epsilon^2)$, and MR1 and MCLI are formally $\mathcal{O}(\epsilon^3)$; the proof of these error-scales can be found in, e.g. [11]. The LSOB gathers the best of both attributes: it is local and formally $\mathcal{O}(\epsilon^3)$ accurate. In this work, numerical accuracy is measured as $\|\mathrm{L}_2(j_x)\| = \sqrt{\sum (j_x^{(\text{num})} - j_x^{(\text{ex})})^2 / \sum (j_x^{(\text{ex})})^2}$ where sums apply to all non-solid sites; that is, $\|\mathrm{L}_2(j_x)\|$ sums bulk and boundary errors. Since we are modelling Stokes duct flow, the momentum equation $(\partial_y^2 j_x + \partial_z^2 j_x) = -F_x/\nu$ is similar to a pure diffusion problem for j_x. Consequently, its LBM-TRT modelling leads to a bulk leading-order error of the form $\mathcal{E}_{\text{bulk}} \propto \epsilon^2\,(\Lambda - 1/6)(\partial_y^4 j_x + \partial_z^4 j_x) + \mathcal{O}(\epsilon^4)$ [11,20], whereas at boundaries the leading error is $\mathcal{E}_{\text{bc}} \propto \mathcal{O}(\epsilon^n)$ with $n = 1, 2, 3$ depending on the boundary scheme.[7] The performance of the six boundary schemes—BB, CLI, MR1, MCLI, Lnode, Lwall—will be examined in this section for the simulation of four pipe flows featuring different cross-sectional shapes. Figure 2 illustrates the $\|L_2(j_x)\|$ versus Λ behaviour of these six schemes for each pipe flow configuration, running on a relatively coarse mesh.

(1) *Rectangular duct flow* with $\bar{y} = y/2W$ and $\bar{z} = z/2H$, where $-\frac{1}{2} \le (\bar{y}, \bar{z}) \le \frac{1}{2}$, has solution [51]:

$$j_x^{(\text{ex})}(\bar{y},\bar{z}) = \frac{16}{\pi^3}\frac{F_x H^2}{\nu} \sum_{n=1,3,\ldots}^{\infty} \frac{1}{n^3}(-1)^{(n-1)/2}\left[1 - \frac{\cosh(n\pi\zeta\bar{y})}{\cosh(n\pi\zeta/2)}\right]\cos(n\pi\bar{z}), \tag{6.1}$$

with $2W$ and $2H$ the duct width and height, the cross-section aspect ratio is $\zeta = W/H$, here $\zeta = 2$. The walls are discretized on a grid-aligned setting placed $\delta_n = 3/4$ away from the boundary nodes (in all four walls). This differs from the traditional halfway $\delta_n = 1/2$, e.g. [33,34], or the on-grid $\delta_n = 0$, e.g. [36,37], discretizations. The other challenging feature refers to corners. At these sites, non-local boundary schemes, such as MR1 and MCLI, do not apply in the two-wall cut diagonal links and are replaced by the local approximation method proposed in Section 5.2.4 of [11], and summarized in appendix B.

Figure 2*a* shows the $\|L_2(j_x)\|$ estimate as a function of Λ, fixing the grid resolution at $N = 14$, with $N = 2H = 2W/\zeta$. Figure 3 presents the mesh convergence studies, $\|L_2(j_x)\|$ versus N, for $\Lambda = \{1/12,\ 1/6,\ 3/16,\ 1/4\}$, and table 1 quantifies the corresponding convergence rates.

Overall, parabolic accurate schemes $\mathcal{O}(\epsilon^3)$ offer the highest accuracy and consistently establish the minimum error at $\Lambda = 1/6$. This observation is theoretically expected as for this purely diffusive problem the optimal diffusion value $\Lambda = 1/6$ [11,56] improves the leading-order bulk correction to $\mathcal{E}_{\text{bulk}} \propto \epsilon^4$, turning the global error $\|L_2(j_x)\|$ limited by $\mathcal{E}_{\text{bc}}$; for parabolic schemes $\mathcal{E}_{\text{bc}} \propto \epsilon^3$. Outside this $\Lambda = 1/6$ neighbourhood, the leading-order bulk error becomes $\mathcal{E}_{\text{bulk}} \propto \epsilon^2$; hence, although parabolic schemes still gain in accuracy, particularly for $\Lambda \le 3/8$, as shown in figure 2*a*, the global error inevitably stays at $\mathcal{O}(\epsilon^2)$. This is the order of accuracy of the linearly accurate CLI scheme. Here, regardless any $\mathcal{E}_{\text{bulk}}$ improvement due to Λ, the measure $\|L_2(j_x)\|$ remains controlled by $\mathcal{E}_{\text{bc}} \propto \epsilon^2$. The same happens with the BB rule, but in a rather more punitive way as $\mathcal{E}_{\text{bc}} \propto \epsilon$. This error degrades the overall simulation accuracy to first order.

[7] The structure of these errors in the duct flow modelling results from the fact that in bulk the third-order derivatives vanish on the account of $\sum_q c_{qx}^2 c_{qy}^3 \partial_y^3 j_x = 0$, $\sum_q c_{qx}^2 c_{qy}^2 c_{qz} \partial_{yyz}^3 j_x = 0$, etc. because the implicated odd-order velocity moments amount to zero. However, this form of truncation error does not apply to boundary rules, which implies that third-order derivatives are typically non-zero. This constrains the boundary error of parabolic schemes to third order.

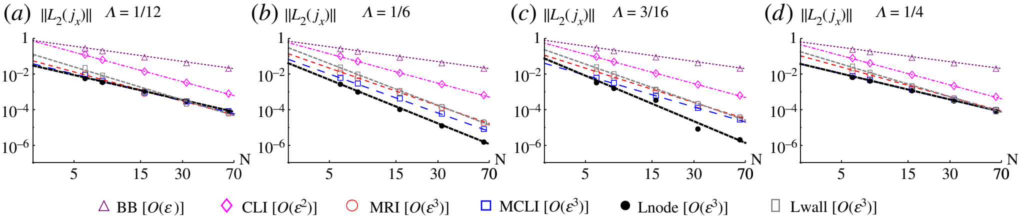

Figure 3. Rectangular duct flow. Mesh convergence analyses $\|L_2(j_x)\|$ versus N for $\Lambda = \{1/12,\ 1/6,\ 3/16,\ 1/4\}$. (Online version in colour.)

Table 1. Quantification of grid convergence rates in figure 3, obtained through a least-square linear fitting of the data points.

	$\Lambda = 1/12$	$\Lambda = 1/6$	$\Lambda = 3/16$	$\Lambda = 1/4$
BB	−1.11	−1.07	−1.10	−1.04
CLI	−2.13	−2.12	−2.12	−2.11
MR1	−2.10	−2.72	−2.48	−2.15
MCLI	−1.94	−2.80	−2.29	−1.86
Lnode	−1.80	−3.15	−3.29	−1.86
Lwall	−2.38	−3.01	−2.75	−2.35

Finally, two interesting numerical observations are worth mentioning, which hint at the form of the higher-order errors in the LSOB prescription of planar mesh-aligned walls. First, when $\Lambda = 1/4$ (which determines the non-equilibrium populations exclusively in the form of directional finite-differences of the equilibrium components [22,49]) the two parabolic schemes Lnode and MLI become equivalent and lead to identical solutions (up to the round-off error) $\forall\, \delta_n \in]0, 1]$. Second, when $\Lambda = 3/16$ and $\delta_n = 1/2$ are fixed, then Lwall and BB lead to identical solutions (up to the round-off error).[8]

(2) *Equilateral triangular duct flow* with $-\frac{1}{2} \leq \bar{y} = y/h \leq \frac{1}{2}$ and $0 \leq \bar{z} = 2/\sqrt{3}(z/h) \leq 1$ has the solution [51]:

$$j_x^{(\text{ex})}(\bar{y}, \bar{z}) = \frac{3}{16}\frac{F_x\, h^2}{\nu}\,(1 - \bar{z})\,\big(\bar{z} - 2\bar{y}\big)\,\big(\bar{z} + 2\bar{y}\big)\,, \tag{6.2}$$

with h the triangle width. Since this duct flow solution is given by a third-order polynomial, then its Stokes flow modelling is exact in bulk, $\mathcal{E}_{\text{bulk}} = 0$, on account of the explanation given at the beginning of this section. Consequently, this benchmark test evaluates the ability of the different boundary schemes to accommodate the flow solution on non mesh-aligned plane walls and the corners forming sharp acute angles. We note that specific LSOB extensions developed to third-order polynomials in [48] were shown to be capable of matching this benchmark exactly.

Figure 2*b* shows the $\|L_2(j_x)\|$ estimate as a function of Λ, fixing the grid resolution at $N = 14$, with $N = h$. The triangle discretization is grid-symmetric, with its centroid placed halfway between grid nodes. Figure 4 presents the mesh convergence studies, $\|L_2(j_x)\|$ versus N, for $\Lambda = \{1/12,\ 1/6,\ 3/16,\ 1/4\}$, and table 2 quantifies the corresponding convergence rates.

The absence of bulk errors explains the lack of a common Λ in the minimization of $\|L_2(j_x)\|$, as visible in figure 2*b*. Such a difference is attributed to the distinct truncations in boundary schemes and the associated formation of 'accommodation layers' between the bulk solution and the wall condition [13,21,22,49]. The primary decisive factor is played by the boundary scheme structure, in terms of link-wise versus node-based. For example, the $\mathcal{O}(\epsilon^3)$ link-wise schemes, MR1 and

[8] For halfway walls $\delta_n = 1/2$, the BB rule establishes the no-slip condition with $\mathcal{O}(\epsilon^3)$ accuracy when $\Lambda = 3/16$ [14,16]. What this observation further indicates is that BB with $\Lambda = 3/16$ is also $\mathcal{O}(\epsilon^3)$ accurate in the prescription of halfway *corner* conditions [48].

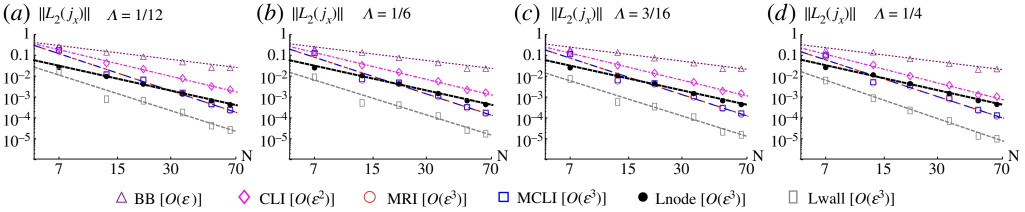

Figure 4. Triangular duct flow. Mesh convergence analyses $\|L_2(j_x)\|$ versus N for $\Lambda = \{1/12,\ 1/6,\ 3/16,\ 1/4\}$. (Online version in colour.)

Table 2. Quantification of grid convergence rates in figure 4, obtained through a least-square linear fitting of the data points.

	$\Lambda = 1/12$	$\Lambda = 1/6$	$\Lambda = 3/16$	$\Lambda = 1/4$
BB	−1.09	−1.06	−1.06	−1.05
CLI	−2.02	−2.03	−2.05	−2.17
MR1	−2.80	−2.77	−2.76	−2.77
MCLI	−2.81	−2.77	−2.77	−2.77
Lnode	−1.88	−1.87	−1.87	−1.89
Lwall	−2.70	−2.63	−2.66	−2.91

MCLI, display nearly overlapping curves, sharing the $\|L_2(j_x)\|$ minimum at $\Lambda = 1/4$ where Eq. (2.6) is exact in terms of the finite-difference operators. However, they significantly differ from the node-based LSOB strategies; what is more, within the LSOB formulation, the Lnode and Lwall display marked differences. The Lwall by making full use of the wall information to determine $\mathcal{X}$ is (i) more accurate, (ii) converges as $\mathcal{O}(\epsilon^3)$, and (iii) minimizes $\|L_2(j_x)\|$ at $\Lambda = 1/6$. The Lnode by operating without any information from the wall to determine $\mathcal{X}$, at least on regular nodes, is (i) seemingly less accurate on the discretization of irregular walls, (ii) degrades its convergence rate to $\mathcal{O}(\epsilon^2)$ and (iii) is roughly insensitive to Λ.[9] By switching to a mixed Lnode/Lwall procedure, where Lnode is shifted to Lwall only on regular nodes, and Lnode is maintained at singular and corner nodes, this hybrid LSOB scheme reaches the performance accuracy of Lwall; this option is worthwhile considering as, at least in corners, Lnode is simpler to implement than Lwall. Comparatively, the other local boundary schemes tested here, CLI and BB, display a lower accuracy, where asymptotically the linearly accurate CLI scheme maintains its $\mathcal{O}(\epsilon^2)$ convergence rate, while the BB rule is clearly $\mathcal{O}(\epsilon)$, as shown in figure 4 and quantified in table 2.

(3) *Circular pipe flow* with $\bar{r} = r/R$, where $0 \leq \bar{r} \leq 1$, has solution [51]:

$$j_x^{(\mathrm{ex})}(\bar{r}) = \frac{F_x R^2}{4\nu}\left(1 - \bar{r}^2\right), \tag{6.3}$$

with R the pipe radius. Here, $\mathcal{E}_{\mathrm{bulk}} = 0$ so this is the canonical example to benchmark the modelling of curved geometries, as the only error source comes from the boundary scheme.

Figure 2, panel (c), shows the $\|L_2(j_x)\|$ estimate as a function of Λ, fixing the grid resolution at $N = 14$, with $N = R$. The circular pipe discretization is grid-symmetric, with its centre placed halfway between grid nodes. Figure 5 presents the mesh convergence studies, $\|L_2(j_x)\|$ versus N, for $\Lambda = \{1/12,\ 1/6,\ 3/16,\ 1/4\}$, and table 3 quantifies the corresponding convergence rates.

Parabolic accurate boundary schemes $\mathcal{O}(\epsilon^3)$ lead to $\mathcal{E}_{\mathrm{bc}} = 0$ in this test. Thus, they exactly reproduce the parabolic solution (6.3) up to machine precision. In contrast, the less accurate schemes, BB and CLI, only describe (6.3) approximately. Surprisingly, despite the BB largest

[9] Technically, these three observations are explained by the Lnode operation principle: the Lnode reconstruction uniformly mixes the first- and third-order derivatives on regular nodes (alongside with all odd higher-order derivatives). As in this test, the third-order derivatives play a dominant role, this mixing tends to produce a lower accuracy.

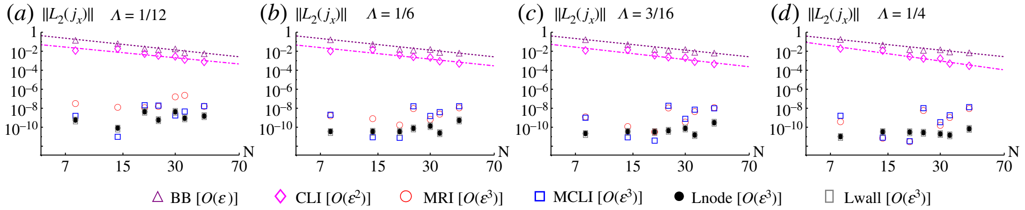

Figure 5. Circular pipe flow. Mesh convergence analyses $||L_2(j_x)||$ versus N for $\Lambda = \{1/12,\ 1/6,\ 3/16,\ 1/4\}$.

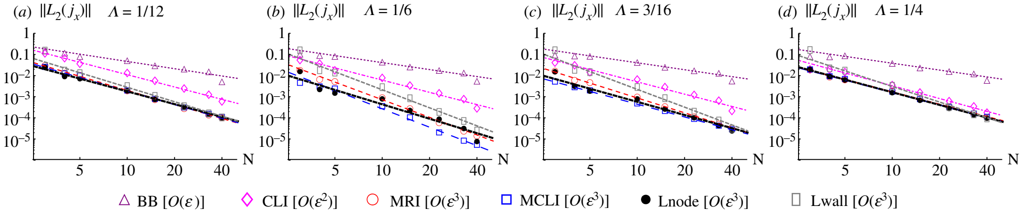

Figure 6. Concentric annular pipe flow. Mesh convergence analyses $||L_2(j_x)||$ versus N for $\Lambda = \{1/12,\ 1/6,\ 3/16,\ 1/4\}$. (Online version in colour.)

Table 3. Quantification of grid convergence rates in figure 5, obtained through a least-square linear fitting of the data points.

	$\Lambda = 1/12$	$\Lambda = 1/6$	$\Lambda = 3/16$	$\Lambda = 1/4$
BB	−1.95	−1.93	−1.92	−1.90
CLI	−1.78	−1.92	−2.07	−2.55

errors, both BB and CLI follow a $\mathcal{O}(\epsilon^2)$ asymptotic behaviour; indeed, that is because the accommodation layers benefit from this problem symmetry and do not distort momentum (cf. a related NSE Poiseuille flow example in [12]). Compared to other local boundary schemes presently available, e.g. [42,47], we highlight that LSOB is, to the best of our knowledge, the only single node-based strategy capable of exactly solving this parabolic profile within a circular pipe geometry (6.3), as originally shown in [48].

(4) *Concentric annular pipe flow* $\bar{r} = r/R_2$, where $\gamma \leq \bar{r} \leq 1$, has solution [51]:

$$j_x^{(\mathrm{ex})}(\bar{r}) = \frac{F_x\, R_2^2}{4\,\nu}\left(1 - \bar{r}^2 + \frac{(\gamma^2 - 1)}{\log\gamma}\log\bar{r}\right), \tag{6.4}$$

with R_1 and R_2 the inner and outer pipe radii, and their ratio $\gamma = R_1/R_2$, here $\gamma = 3/11$.

Figure 2*d* shows the $||L_2(j_x)||$ estimate as a function of Λ, fixing the grid resolution at $N = 14$, with $N = R_2 - R_1$. The annular pipe discretization is grid-symmetric, with its centre placed halfway between grid nodes. Figure 6 presents the mesh convergence studies, $||L_2(j_x)||$ versus N, for $\Lambda = \{1/12,\ 1/6,\ 3/16,\ 1/4\}$, and table 4 quantifies the corresponding convergence rates.

The simulation of this problem is affected by both bulk and boundary errors. The vanishing of the leading-order bulk error occurs with $\Lambda = 1/6$ (the optimal diffusion value), which leads to $\mathcal{E}_{\mathrm{bulk}} \propto \epsilon^4$, so that, $||L_2(u_x)||$ becomes essentially determined by the boundary scheme accuracy. For parabolic schemes we confirm the $\mathcal{O}(\epsilon^3)$ convergence rate at $\Lambda = 1/6$. The exception happens for Lnode, where the convergence rate is somewhat slower, i.e. $\mathcal{O}(\epsilon^{2.5})$, due to the problems already identified in the triangular duct flow benchmark. Also different from the other parabolic schemes is the optimal Λ interval in Lwall, which appears to be wider, cf. figure 2*d* and table 4. Comparing Lnode against Lwall, the Lnode approach seems better at representing the curved wall shapes at coarse meshes, while at finer meshes, as the radial location of boundary nodes tends to the

Table 4. Quantification of grid convergence rates in figure 6, obtained through a least-square linear fitting of the data points.

	$\Lambda = 1/12$	$\Lambda = 1/6$	$\Lambda = 3/16$	$\Lambda = 1/4$
BB	−1.12	−1.03	−1.02	−0.99
CLI	−1.93	−1.89	−1.88	−1.88
MR1	−2.22	−2.75	−2.37	−1.96
MCLI	−2.13	−2.76	−2.00	−1.97
Lnode	−2.06	−2.45	−2.24	−2.03
Lwall	−2.54	−3.02	−2.99	−2.67

circular wall surface, i.e. $\lim_{N\to\infty} r \approx R$, then the performance of Lwall becomes comparable or even superior. The CLI scheme displays a $\mathcal{O}(\epsilon^2)$ accuracy for all tested Λ values. Finally, the BB rule, which leads to a staircase discretization of the real geometry, makes the overall error $\mathcal{O}(\epsilon)$.

7. Conclusion

This work revived a long-developed boundary scheme for the Dirichlet momentum boundary condition in LBM, called the LSOB method. As a novelty, we reformulated the original LSOB idea to operate within the TRT framework for the d3Q19 lattice; a task that carried many operational advantages. Two LSOB strategies were considered in this work: Lwall [21] and Lnode [48], each of which operated on wall and node variables, respectively. Both strategies were exemplified for planar (mesh-aligned and/or inclined) and curved walls, including corners. Numerical tests performed here for several non-trivial duct flow geometries indicate the good performance of LSOB schemes, revealing their matching accuracy with other parabolic accurate schemes, such as MR1 and MCLI schemes, although the former operate on a single node.

Ideally, the use of parabolic accurate boundary schemes, like LSOB, should be advantageous for a number of reasons. Namely, (i) they tremendously save the computational time in the asymptotic limit [19,20] and (ii) they provide high levels of accuracy within coarse to moderate grid resolutions [11,16]. This last asset is particularly important for porous media applications, where the spacing between the walls bounding the preferential flow path is often limited to a few grid nodes. Although the ‘ideal’ characteristics of parabolic schemes are conceived to not always prevail for all flow problem classes in an universal manner, they were globally confirmed to be satisfied here in the simulation of slow flows within several pipe geometries. Equally important, the renewed LSOB formulations were verified to support the correct parametrization beyond the second order, ensuring that, in Stokes flow, the dimensionless flow and relative momentum errors hold independent of kinematic viscosity. The establishment of this property was one of our original motivations to revive the LSOB working principle in this work. Still, additional work requires ‘standardising’ the LSOB application over arbitrary flow classes and/or geometries. This task includes the method extension to multi-dimensional Stokes and Navier–Stokes flows, as initiated in [41,42,47]. Also, the generalization of the LSOB formulation to other lattices, including higher-order lattices [39], is planned for future work, along with the extension of the LSOB framework to other collision operators. Lastly, a dedicated stability analysis comparing the parabolic two-point MR and local LSOB schemes, which has not been devised yet, should be conducted in a future study.

Finally, we conclude with a summary of the main LSOB characteristics. The main drawback of the LSOB, compared to the parabolic MR schemes, is that the prescription of Dirichlet conditions is made lattice, dimension and problem dependent, moreover its implementation is presently restricted to laminar (low Reynolds) flows; however, on the positive side, the LSOB is local and it offers a larger potential for mixed-type boundary conditions than the directional boundary schemes in general. These advantageous characteristics of LSOB to prescribe strain-rate, or pressure and tangential velocity in flow schemes, but also Neumann- or Robin-type boundary conditions in flow or ADE problems [56], are planned to be explored in future works.

Data accessibility. This article has no additional data.

Authors' contributions. Both authors worked together on the methodology, construction of the numerical tests and writing of the paper. G.S. performed the coding and the numerical computations.

Competing interests. The authors declare that they have no competing interests.

Funding. IRSTEA through project 4100 financially supported the talk on this work presented at the 28th DSFD conference. G.S. acknowledges *Fundação para a Ciência e Tecnologia* (FCT) through grant no. SFRH/BPD/111228/2015, and FCT, through IDMEC, under LAETA, project UID/EMS/50022/2019.

Acknowledgements. I.G. is grateful to D. d'Humières for common work on the LSOB in 1995.

Appendix A. Summary of LSOB algorithms in Stokes duct flows

Algorithm 1 . Lnode approach.

Initialization step

1: Identify the **boundary nodes** $\boldsymbol{x}_b$ and the associated **links** $q \in \mathcal{U}$ and $q \in \mathcal{K}$.

2: Identify the **set of unknowns** $\mathcal{X} = \{\partial_y j_x, \ldots, \partial_{yy} j_x, \ldots\}^\intercal$, which are more relevant to reconstruct $f_q(\boldsymbol{x}_b)$ with $q \in \mathcal{U}$, as given by (3.3) with expansion (3.4).

3: Differentiate between **regular and singular nodes** based on $\dim(\mathcal{U})$ and $\dim(\mathcal{X})$ for each $\boldsymbol{x}_b$. Also identify **corner nodes** based on the geometry discretisation.

4: Construct **matrix of coefficients** $[\mathbf{M}]$, defined as part of the system:

$$\mathcal{N} = [\mathbf{M}]\,\mathcal{X} \tag{A 1}$$

which is the linear algebra transcription of (3.2); assuming the vector $\mathcal{N}$ contains Q_m *known* post-collision populations, $\mathcal{N} = \{\hat{n}_q^+, \hat{n}_q^-\}^\intercal$, which are readily available from the TRT symmetry argument.

5: Only for **singular nodes**, i.e. $\boldsymbol{x}_b$ sites where $\operatorname{rank}[\mathbf{M}] < \dim(\mathcal{X})$, add additional constraints (discussed in Secs. 4 and/or 5) to the linearly independent rows of $[\mathbf{M}]$ so that (A 1) gets replaced by the augmented system:

$$\mathcal{N}_h = [\mathbf{M_h}]\,\mathcal{X}, \tag{A 2}$$

where $\operatorname{rank}[\mathbf{M_h}] > \operatorname{rank}[\mathbf{M}]$. The content of $[\mathbf{M_h}]$ is specific to singular $\boldsymbol{x}_b$, cf. Section 4a.

6: Compute the **inverse of the matrix of coefficients** $[\mathbf{M}]^{-1}$ or $[\mathbf{M_h}]^{-1}$. In regular nodes, $\dim(\mathcal{X}) \leq \operatorname{rank}[\mathbf{M}]$, then $[\mathbf{M}]$ is typically a rectangular matrix and its pseudo-inverse $[\mathbf{M}]^{-1}$ can be determined through standard numerical programming techniques, such as the singular value decomposition (SVD);[10]note, SVD also applies to singular nodes, as given by (A 2).

Main LBM algorithm: Boundary Condition step (computed after TRT collision step)

7: Compute the **non-equilibrium hydrodynamic fields** $\mathcal{X}$ on $\boldsymbol{x}_b$ at each time step as follows:

$$\mathcal{X} = \begin{cases} [\mathbf{M_h}]^{-1}\,\mathcal{N}_h & \text{if } \boldsymbol{x}_b \text{ is singular node,} \\ [\mathbf{M}]^{-1}\,\mathcal{N} & \text{otherwise,} \end{cases} \tag{A 3}$$

where the post-collision populations, $\{\hat{n}_q^+, \hat{n}_q^-\}^\intercal$, to be used in $\mathcal{N}$ or $\mathcal{N}_h$, are furnished by the TRT collision step, and so the full set Q_m is readily available from TRT symmetry argument.

8: Reconstruct the *post-stream* **unknown boundary populations**, i.e. $f_q(\boldsymbol{x}_b, t+1)$ with $q \in \mathcal{U}$, by substituting the $\mathcal{X}$ solution into (3.3) with approximation (3.4), plus P computed from (3.6).

Algorithm 2 . Lwall approach.

Initialization step

1: Identify **boundary nodes** $\boldsymbol{x}_b$ and the associated **links** $q \in \mathcal{U}$ and $q \in \mathcal{K}$.

2: Identify the **set of unknowns** $\mathcal{X}$, which are more relevant to reconstruct $f_q(\boldsymbol{x}_b)$ with $q \in \mathcal{U}$, as given by (3.3) with expansion (3.5).

3: If applicable, e.g. over curved walls, differentiate between **regular and singular nodes** based on $\dim(\mathcal{U})$ and $\dim(\mathcal{X})$ for each $\boldsymbol{x}_b$. Also identify **corner nodes** based on the geometry discretization.

4: Construct **matrix of coefficients** $[\mathbf{M}]$, given in the form (A 1); but now vector $\mathcal{N}$ contains a subset of *known* post-collision populations and vector $\mathcal{X}$ the *unknown* hydrodynamic fields set at $\boldsymbol{x}_w$, following (3.5). Here, $\text{rank}\,[\mathbf{M}] = \dim(\mathcal{X})$ with $[\mathbf{M}]$ specific to each $\boldsymbol{x}_b$, cf. 4b and 5b.

5: Compute the **inverse of the matrix of coefficients**, $[\mathbf{M}]^{-1}$. Since $[\mathbf{M}]$ is a square matrix its inverse $[\mathbf{M}]^{-1}$ is direct (assuming $\det[\mathbf{M}] \neq 0$). No additional constraints are necessary.

6: If applicable, e.g. over curved walls, augment the Lwall system with additional constraints as $\mathcal{N}_h = [\mathbf{M_h}]\,\mathcal{X}$, cf. 5b, and proceed with solution $\mathcal{X} = [\mathbf{M}]^{-1}\,\mathcal{N}_h$, as explained in steps 5 and 6 of the Lnode algorithm.

Main LBM algorithm: Boundary Condition step (computed after TRT collision step)

7: Compute **non-equilibrium hydrodynamic fields** $\mathcal{X}$ on $\boldsymbol{x}_w$ at each time step as follows:

$$\mathcal{X} = [\mathbf{M}]^{-1}\,\mathcal{N}, \tag{A 4}$$

using a pre-selected subset of post-collision populations, $\{\hat{n}_q^+, \hat{n}_q^-\}^\intercal$ in $\mathcal{N}$ that is furnished by the TRT collision step. If applicable, compute $\mathcal{X} = [\mathbf{M_h}]^{-1}\,\mathcal{N}_h$ at singular nodes only.

8: Reconstruct the *post-stream* **unknown boundary populations**, i.e. $f_q(\boldsymbol{x}_b, t+1)$ with $q \in \mathcal{U}$, by substituting the $\mathcal{X}$ solution into (3.3) with approximation (3.5), plus P computed from (3.6).

Appendix B. Summary of link-wise boundary schemes for general flow

(a) Wall surfaces

Section 6 implements the Dirichlet boundary condition on the wall surface considering four types of link-wise boundary schemes. They can be expressed through the generic boundary update rule, applied *after* streaming step [11]:

$$\begin{aligned} f_{\bar{q}}(\boldsymbol{x}_b, t+1) = {} & \kappa_1 \widehat{f}_q(\boldsymbol{x}_{\mathrm{b}}, t) + \bar{\kappa}_{-1} \widehat{f}_{\bar{q}}(\boldsymbol{x}_b, t) + \kappa_0 f_q(\boldsymbol{x}_b, t+1) \\ & + \kappa_{-1} f_q(\boldsymbol{x}_b - \boldsymbol{c}_q, t+1) + \bar{\kappa}_{-2} \widehat{f}_{\bar{q}}(\boldsymbol{x}_b - \boldsymbol{c}_q, t) + F_q^{p.c.}(\boldsymbol{x}_b, t) - \alpha^{(u)} j^{\star}_{q\,w}(\boldsymbol{x}_w, t), \end{aligned} \tag{B 1}$$

where $\boldsymbol{x}_b$ is a boundary node, with $\boldsymbol{x}_b + \boldsymbol{c}_q$ a solid node and $\boldsymbol{x}_b - \boldsymbol{c}_q$ the nearest fluid node. For each link-wise boundary scheme the coefficients $\{\kappa_1, \kappa_0, \bar{\kappa}_{-1}, \kappa_{-1}, \bar{\kappa}_{-2}\}$ and the parametrization factor $\alpha^{(u)}$ are provided in table 5. The Dirichlet momentum correction in (B 1) is given by

$$j^{\star}_{q\,w} = t^{\star}_q\, j_{q\,w}, \quad j_{q\,w} = \boldsymbol{c}_q \cdot \boldsymbol{j}_w, \tag{B 2}$$

and the post-collision correction $F_q^{p.c.}$ in (B 1) is given by

$$F_q^{p.c.}(\boldsymbol{x}_b) = \begin{cases} 0 & \text{for} \quad \text{BB and CLI}, \\ \alpha^{(u)} \left(\tau^- - \frac{1}{2}\right) \left(\hat{n}_q^- - S_q\right)|_{(\boldsymbol{x}_b)} & \text{for} \quad \text{MR1}, \\ \alpha^{(u)} \left(\tau^- - \frac{1}{2}\right) \left(\hat{n}_q^- - S_q\right)|_{(\boldsymbol{x}_b)} + \alpha^{(u)} \frac{\delta_q^2}{2} \overline{\Delta}_q^2 j_q|^{F.D.}_{(\boldsymbol{x}_b)} & \text{for} \quad \text{MCLI}. \end{cases} \tag{B 3}$$

[10] The SVD is generally effortless, being automatic with command ‘pinv’ in MATLAB or in C++ Math Library, for example.

Table 5. Coefficients in (A 1), with $\kappa_1 = 1$ in all schemes, and the corresponding parametrization factor $\alpha^{(u)}$.

	BB	CLI	MR1	MCLI
κ_0	0	$\frac{1-2\delta_q}{1+2\delta_q}$	$\frac{1-2\delta_q-2\delta_q^2}{(1+\delta_q)^2}$	$\frac{1-2\delta_q}{1+2\delta_q}$
$\bar{\kappa}_{-1}$	0	$-\kappa_0$	$-\kappa_0$	$-\kappa_0$
κ_{-1}	0	0	$\frac{\delta_q^2}{(1+\delta_q)^2}$	0
$\bar{\kappa}_{-2}$	0	0	$-\kappa_{-1}$	0
$\alpha^{(u)}$	2	$\frac{4}{1+2\delta_q}$	$\frac{4}{(1+\delta_q)^2}$	$\frac{4}{1+2\delta_q}$

Note, for the MCLI scheme the $F_q^{p.c.}$ term in (B 3) includes the momentum Laplacian approximation $\overline{\Delta}_q^2 j_q|^{F.D.}$ computed through the link-wise second-order finite-difference

$$\overline{\Delta}_q^2 j_q|^{F.D.}(\boldsymbol{x}_b) = \frac{2}{\delta_q + \delta_{\bar{q}}} \left(\frac{j_q(\boldsymbol{x}_w) - j_q(\boldsymbol{x}_b)}{\delta_q} - \frac{j_q(\boldsymbol{x}_b) - j_q(\boldsymbol{x}_{\bar{w}})}{\delta_{\bar{q}}} \right), \tag{B 4}$$

with $\boldsymbol{x}_w = \boldsymbol{x}_b + \delta_q \boldsymbol{c}_q$, $\boldsymbol{x}_{\bar{w}} = \boldsymbol{x}_b + \delta_{\bar{q}} \boldsymbol{c}_{\bar{q}}$, $\delta_q \neq 0$ and $\delta_{\bar{q}} \neq 0$. In our notation, if $\boldsymbol{x}_w$ denotes a solid node, with wall momentum $\boldsymbol{j}(\boldsymbol{x}_w)$ (or simply $\mathbf{0}$ for a resting wall), then $\boldsymbol{x}_{\bar{w}}$ denotes a fluid node, meaning $\delta_{\bar{q}} = 1$ and $j_q(\boldsymbol{x}_{\bar{w}})$ is extracted from the LBM bulk solution, unless in corners as considered below.

(b) Wall corners

At corner nodes there is no next directional neighbour. Hence, the non-local MR schemes, MR1 and MCLI, cannot be applied at these sites. Only at corner nodes do the parabolic schemes use (B 1) with the parameters of the MCLI scheme, given in table 5 and equations (B 2) and (B 3). The only difference is that the non-local momentum Laplacian approximation $\overline{\Delta}_q^2 j_q|^{F.D.}$ is no longer computed with (B 4). Instead, $\overline{\Delta}_q^2 j_q|^{F.D.}$ in $F_q^{p.c.}$ is locally approximated as follows [11]:

$$\overline{\Delta}_q^2 j_q|^{F.D.}(\boldsymbol{x}_b) = -\left(\tau^- - 1/2\right)^{-1} \left(\hat{n}_q^- - \partial_q^{F.D.} e_q^+\right)|_{(\boldsymbol{x}_b)}, \tag{B 5}$$

with

$$\partial_q^{F.D.} e_q^+(\boldsymbol{x}_b) = -\sum_{\alpha=1}^{d} \left(e_q^+(\boldsymbol{x}_b + c_{\bar{q}\alpha}) - e_q^+(\boldsymbol{x}_b)\right), \tag{B 6}$$

where $c_{\bar{q}\alpha}$ denotes propagation links parallel to the principal coordinate axis.

References

1. Nield DA, Bejan A. 2006 *Convection in porous media*, 3rd edn. New York, NY: Springer.
2. Schiller UD, Wang F. 2018 Multiscale simulation of transport phenomena in porous media: from toy models to materials models. *MRS Commun.* **8**, 358–371. (doi:10.1557/mrc.2018.29)
3. Succi S. 2018 *The Lattice Boltzmann equation: for complex states of flowing matter*. Oxford, UK: Oxford University Press.
4. Krüger T, Kusumaatmaja H, Kuzmin A, Shardt O, Silva G, Viggen EM. 2016 *The Lattice Boltzmann method - principles and practice*, 1st edn. New York, NY: Springer.
5. Chen S, Doolen G. 1998 Lattice Boltzmann method for fluid flows. *Annu. Rev. Fluid Mech.* **30**, 329. (doi:10.1146/annurev.fluid.30.1.329)
6. Aidun CK, Clausen JR. 2010 Lattice-Boltzmann method for complex flows. *Annu. Rev. Fluid Mech.* **42**, 439. (doi:10.1146/annurev-fluid-121108-145519)

7. Maier RS, Bernard RS. 2010 Lattice-Boltzmann accuracy in pore-scale flow simulation. *J. Comput. Phys.* **229**, 233–255. (doi:10.1016/j.jcp.2009.09.013)
8. Boek E, Venturoli M. 2010 Lattice-Boltzmann studies of fluid flow in porous media with realistic rock geometries. *Comput. Math. Appl.* **59**, 2305–2314. (doi:10.1016/j.camwa.2009.08.063)
9. Rao P, Schaefer L. 2019 Lattice Boltzmann models for micro-tomographic pore-spaces. *Comput. Fluids* **193**, 104294. (doi:10.1016/j.compfluid.2019.104294)
10. Ginzburg I. 2005 Generic boundary conditions for Lattice Boltzmann models and their application to advection and anisotropic-dispersion equations. *Adv. Wat. Res.* **28**, 1196–1216. (doi:10.1016/j.advwatres.2005.03.009)
11. Ginzburg I, Verhaeghe F. 2008 Two-relaxation-time lattice Boltzman scheme: about parametrization, velocity, pressure and mixed conditions. *Commun. Comp. Phys.* **3**, 427.
12. Ginzburg I, Verhaeghe F. 2008 Study of simple hydrodynamic solutions with the two-relaxation-times Lattice Boltzmann scheme. *Commun. Comput. Phys.* **3**, 519.
13. Cornubert R, Levermore D. 1991 A Knudsen layer theory for lattice gases. *Phys. D* **47**, 241–259. (doi:10.1016/0167-2789(91)90295-K)
14. Ginzbourg I, Adler PM. 1994 Boundary flow condition analysis for the three-dimensional lattice Boltzmann model. *J. Phys. II Fr.* **4**, 191–214.
15. Ladd AJC, Verberg R. 2001 Lattice-Boltzmann simulations of particle–fluid suspensions. *J. Stat. Phys.* **104**, 1191–1251. (doi:10.1023/A:1010414013942)
16. Ginzburg I. 2003 Multi-reflection boundary conditions for lattice Botlzmann models. *Phys. Rev. E* **68**, 066614. (doi:10.1103/PhysRevE.68.066614)
17. Pan C, Luo L-S, Miller CT. 2006 An evaluation of lattice Boltzmann schemes for porous media simulation. *Comput. Fluids* **35**, 898–909. (doi:10.1016/j.compfluid.2005.03.008)
18. Chun B, Ladd AJC. 2007 Interpolated boundary conditions for lattice Boltzmann simulations of flows in narrow gaps. *Phys. Rev. E* **75**, 066705. (doi:10.1103/PhysRevE.75.066705)
19. Talon L, Bauer D, Gland N, Youssef S, Auradou H, Ginzburg I. 2012 Assessment of the two relaxation time lattice-Boltzmann scheme to simulate Stokes flow in porous media. *Water Resour. Res.* **48**, W04526. (doi:10.1029/2011WR011385)
20. Khirevich S, Ginzburg I, Tallarek U. 2015 Coarse-and fine-grid numerical behavior of MRT/TRT lattice-Boltzmann schemes in regular and random sphere packings. *J. Comput. Phys.* **281**, 708–742. (doi:10.1016/j.jcp.2014.10.038)
21. Ginzburg I. 1996 Local second-order boundary method for lattice Botlzmann models. *J. Stat. Phys.* **84**, 927–971. (doi:10.1007/BF02174124)
22. d'Humières D, Ginzburg I. 2009 Viscosity independent numerical errors for Lattice Boltzmann models: from recurrence equations to 'magic' collision numbers. *Comput. Math. Appl.* **58**, 823. (doi:10.1016/j.camwa.2009.02.008)
23. Filippova O, Hänel D. 1997 Lattice Boltzmann simulation of gas-particle flow in filters. *Comput. Fluids* **26**, 697. (doi:10.1016/S0045-7930(97)00009-1)
24. Bouzidi M, Firdaouss M, Lallemand P. 2001 Momentum transfer of a Boltzmann-lattice fluid with boundaries. *Phys. Fluids* **13**, 3452. (doi:10.1063/1.1399290)
25. Yu D, Mei R, Luo L-S, Shyy W. 2003 Viscous flow computations with the method of lattice Boltzmann equation. *Prog. Aerosp. Sci.* **39**, 329. (doi:10.1016/S0376-0421(03)00003-4)
26. Silva G, Talon T, Ginzburg I. 2017 Low-and high-order accurate boundary conditions: from Stokes to Darcy porous flow modeled with standard and improved Brinkman lattice Boltzmann schemes. *J. Comput. Phys.* **335**, 50–83. (doi:10.1016/j.jcp.2017.01.023)
27. Silva G, Semiao V. 2017 Consistent lattice Boltzmann modeling of low-speed isothermal flows at finite Knudsen numbers in slip-flow regime: application to plane boundaries. *Phys. Rev. E* **96**, 013311. (doi:10.1103/PhysRevE.96.013311)
28. Silva G. 2018 Consistent lattice Boltzmann modeling of low-speed isothermal flows at finite Knudsen numbers in slip-flow regime. II. Application to curved boundaries. *Phys. Rev. E* **98**, 023302. (doi:10.1103/PhysRevE.98.023302)
29. Chapman S, Cowling T. 1970 *The mathematical theory of non-uniform gases*. Cambridge, UK: Cambridge University Press.
30. Skordos PA. 1993 Initial and boundary conditions for the lattice Boltzmann method. *Phys. Rev. E* **48**, 4823. (doi:10.1103/PhysRevE.48.4823)

31. Noble NR, Chen S, Georgiadis JG, Buckius RO. 1995 A consistent hydrodynamic boundary condition for the lattice Boltzmann method. *Phys. Fluids,* **7**, 203–209. (doi:10.1063/1.868767)
32. Ginzburg I. 1994 *Boundary conditions problems in lattice gas methods for single and multiple phases.* Ph.D. Thesis, University Paris VI.
33. Maier RS, Bernard RS, Grunau DW. 1996 Boundary conditions for the Lattice Boltzmann method. *Phys. Fluids* **8**, 1788–1801. (doi:10.1063/1.868961)
34. Zou Q, He X. 1997 On pressure and velocity boundary conditions for the lattice Boltzmann BGK model. *Phys. Fluids.* **9**, 1592–1598.
35. Halliday I, Hammond LA, Care CM. 2002 Enhanced closure scheme for lattice Boltzmann equation hydrodynamics. *J. Phys. A: Math. Gen.* **35**, 157–166. (doi:10.1088/0305-4470/35/12/102)
36. Latt J, Chopard B, Malaspinas O, Deville M, Michler A. 2008 Straight velocity boundaries in the lattice Boltzmann method. *Phys. Rev. E* **77**, 056703. (doi:10.1103/PhysRevE.77.056703)
37. Hecht M, Harting J. 2010 Implementation of on-site velocity boundary conditions for D3Q19 lattice Boltzmann simulations. *J. Stat. Mech. Theory Exp.* **1**, P01018.
38. Verschaeve JCG. 2011 Involving the Navier–Stokes equations in the derivation of boundary conditions for the lattice Boltzmann method. *Phil. Trans. R. Soc. A* **369**, 2184–2192. (doi:10.1098/rsta.2011.0045)
39. Malaspinas O, Chopard B, Latt J. 2011 General regularized boundary condition for multi-speed lattice Boltzmann models. *Comput. Fluids* **49**, 29–35. (doi:10.1016/j.compfluid.2011.04.010)
40. Mohammed S, Graham G, Reis T. 2018 Assessing moment-based boundary conditions for the lattice Boltzmann equation: a study of dipole-wall collisions. *Comput. Fluids.* **176**, 79–96. (doi:10.1016/j.compfluid.2018.08.025)
41. Ginzburg I, Steiner C. 2003 Lattice Boltzmann model for free-surface flow and its application to filling process in casting. *J. Comput. Phys.* **185**, 61–99. (doi:10.1016/S0021-9991(02)00048-7)
42. Junk M, Yang Z. 2005 One-point boundary condition for the lattice Boltzmann method. *Phys. Rev. E* **72**, 066701. (doi:10.1103/PhysRevE.72.066701)
43. Verschaeve JCC, Müller B. 2010 A curved no-slip boundary condition for the lattice Boltzmann method. *J. Comput. Phys.* **229**, 6781–6803. (doi:10.1016/j.jcp.2010.05.022)
44. Mohammadipour OR, Niazmand H, Mirbozorgi SA. 2014 Alternative curved-boundary treatment for the lattice Boltzmann method and its application in simulation of flow and potential fields. *Phys. Rev. E* **89**, 013309. (doi:10.1103/PhysRevE.89.013309)
45. Dorschner B, Chikatamarla SS, Bösch F, Karlin IV. 2015 Grad's approximation for moving and stationary walls in entropic lattice Boltzmann simulations. *J. Comput. Phys.* **295**, 340–354. (doi:10.1016/j.jcp.2015.04.017)
46. Mohammadipour OR, Niazmand H, Succi S. 2017 General velocity, pressure, and initial condition for two-dimensional and three-dimensional lattice Boltzmann simulations. *Phys. Rev. E* **95**, 033301. (doi:10.1103/PhysRevE.95.033301)
47. Zhang L., Zeng Z, Xie H, Tau X, Zhang Y, Lu Y, Yoshikawa A, Kawazoe Y. 2015 An alternative second order scheme for curved boundary condition in lattice Boltzmann method. *Comput. Fluids.* **144**, 193–202. (doi:10.1016/j.compfluid.2015.03.006)
48. Ginzburg I. 1995 Local second-order boundary method for lattice Botlzmann models. Part II. Application to Poisson equation in complicated geometries. Unpublished.
49. Ginzburg I, Roux L, Silva G. 2015 Local boundary reflections in lattice Boltzmann schemes: spurious boundary layers and their impact on the velocity, diffusion and dispersion. *C. R. Mec.* **343**, 518–532. (doi:10.1016/j.crme.2015.03.004)
50. Sengupta A, Hammond PS, Frenkel D, Boek ES. 2012 Error analysis and correction for lattice Boltzmann simulated flow conductance in capillaries of different shapes and alignments. *J. Comput. Phys.* **231**, 2634–2640. (doi:10.1016/j.jcp.2011.12.004)
51. Shah RK, London AL 1978 *Laminar Flow Forced Convection in Ducts*. Supplement 1 to Advances in Heat Transfer. New York, NY: Academic Press.
52. Qian Y, Lallemand P. 1992 Lattice BGK models for Navier-Stokes equation. *Europhys. Lett.* **17**, 479–483. (doi:10.1209/0295-5075/17/6/001)
53. Guo Z, Zheng C, Shi B. 2002 Discrete lattice effects on the forcing term in the lattice Boltzmann method. *Phys. Rev. E.* **65**, 046308. (doi:10.1103/PhysRevE.65.046308)

54. Silva G, Semiao V. 2012 First- and second-order forcing expansions in a lattice Boltzmann method reproducing isothermal hydrodynamics in artificial compressibility form. *J. Fluid Mech.* **698**, 282–303. (doi:10.1017/jfm.2012.83)

55. Bauer M, Silva G, Rüde U. 2020 Truncation errors of the D3Q19 lattice model for the lattice Boltzmann method. *J. Comput. Phys.* **405**, 109111. (doi:10.1016/j.jcp.2019.109111)

56. Ginzburg I. 2005 Equilibrium-type and link-type lattice Boltzmann models for generic advection and anisotropic-dispersion equation. *Adv. Wat. Res.* **28**, 1171–1195. (doi:10.1016/j.advwatres.2005.03.004)