Consistent time-step optimization in the lattice Boltzmann method

Tobias Horstmann, Hatem Touil, Lucien Vienne, Denis Ricot, Emmanuel Lévêque

Consistent time-step optimization in the lattice Boltzmann method

Tobias Horstmann[1,2,*], Hatem Touil[2], Lucien Vienne[3], Denis Ricot[4], and Emmanuel Lévêque[3]

[1]German Aerospace Center, Institute of Propulsion Technology, Department of Engine Acoustics, Bismarckstraße 101, D-10625 Berlin, Germany
[2]CS GROUP, 22, av. Galilée, F-92350, Le Plessis Robinson, France
[3]Univ Lyon, CNRS, Ecole Centrale de Lyon, Univ Claude Bernard Lyon 1, INSA Lyon, LMFA, UMR5509, F-69130, Ecully, France
[4]Renault, Technocentre, 1 av. du Golf, F-78280 Guyancourt, France
[*]Corresponding author. Email address: tobias.horstmann@dlr.de

August 24, 2022

Abstract

Owing to its efficiency and aptitude for a massive parallelization, the lattice Boltzmann method generally outperforms conventional solvers in terms of execution time in weakly-compressible flows. However, the authorized time-step (being inversely proportional to the speed of sound) becomes prohibitively small in the incompressible limit, so that the performance advantage over continuum-based solvers vanishes. A remedy to increase the time-step is provided by artificially tailoring the speed of sound throughout the simulation, so as to reach a fixed target Mach number much larger than the actual one. While achieving considerable speed-ups in certain flow configurations, such *adaptive time-stepping* comes with the flaw that the continuities of mass density and pressure cannot be fulfilled conjointly when the speed of sound is varied. Therefore, a trade-off is needed. By leaving the mass density unchanged, the conservation of mass is preserved but the pressure presents a discontinuity in the momentum equation. In contrast, a power-law rescaling of the mass density allows us to ensure the continuity of the pressure term in the momentum equation (per unit mass) but leaves the mass density locally discontinuous. This algorithm, which requires a rescaling operation of the mass density, will be called "*adaptive time-stepping* with correction" in the article. Interestingly, we found that this second trade-off is generally preferable.

In the case of a thermal plume, whose movement is governed by the balance of buoyancy and drag forces, the correction of the mass density (to ensure the continuity of the pressure force) has a beneficial impact on the resolved velocity field. In a pulsatile channel flow (Womersley's flow) driven by an external body force, no difference was observed between the two versions of *adaptive time-stepping*.

On the other hand, if the pulsatile flow is established by inlet and outlet pressure conditions, the results obtained with a continuous pressure force agree much better with the analytical solution. Finally, by using *adaptive time-stepping* in a channel entrance flow, it was shown that the correction is compulsory for the Poiseuille flow to develop. The expected compressibility error due to the discontinuity in the mass density remains small to negligible, and the convergence rate is not notably affected compared to a simulation with a constant time step.

Keywords— Lattice Boltzmann method, adaptive time-stepping, time-step optimization

1 Introduction

The lattice Boltzmann (LB) method has gained prominence as a statistical approach used to simulate continuum fluid dynamics [Suc15; Krü+17]. Nonetheless, to make it a competitive alternative to conventional solvers, a series of simplifications are necessary that eventually narrow its scope of application. Concretely, the macroscopic equivalent to the standard stream-and-collide LB algorithm reduces to an isothermal, weakly compressible Navier-Stokes model [Del01].

Due to the low symmetry of standard lattices, the LB method can only be used with confidence in the range of Mach number

$$\mathrm{Ma} = \frac{|\boldsymbol{u}|_{\max}}{c_s} \leq 0.3 \tag{1}$$

where compressible effects may be considered weak to negligible; $|\boldsymbol{u}|_{\max}$ denotes the maximum flow velocity and c_s is the speed of sound in the fluid. The restriction to isothermal fluids is another consequence of the low symmetry of the lattice, which fails to properly express the conservation of internal energy with a varying temperature field [Del01]. Moreover, the LB method is by nature a compressible method.

A variety of techniques exist to lift the restrictions to low Mach numbers and isothermal fluids in the LB approach [Qia93; CK06; FST15; Li+07]. On the contrary, improvements to address incompressible fluids are rather limited. Incompressible LB models have been postulated in the past but they only decrease the order of compressibility errors in steady flows [HL97; Zou+95; Del03].

In continuum fluid mechanics, the principal motivation behind an incompressible description stems from the numerical benefit of an increased time-step. In general, the maximum time-step is expressed as

$$\Delta t_{\max} = \frac{\mathrm{CFL}\ \Delta x}{v_{\max}} \tag{2}$$

where the Courant-Friedrichs-Lewy (CFL) number may be viewed as the normalized maximum velocity at which flow variations can be robustly propagated by the numerical scheme [CFL28]. For compressible Navier-Stokes solvers, $v_{\max}$ is the speed of sound relative to the maximum flow velocity, *i.e.*

$$v_{\max} = c_s + |\boldsymbol{u}|_{\max} \qquad \text{and} \qquad \Delta t_{\max} = \frac{\mathrm{CFL}\ \Delta x}{c_s(1+\mathrm{Ma})} \tag{3}$$

If one supposes a CFL number of unity and $\Delta x \approx 10^{-3}$ m then $\Delta t_{\max}$ for air at standard conditions ($c_s \simeq 343$ m/s) is approximately $3 \times 10^{-6}/(1+\mathrm{Ma})$ seconds yielding

very small time-steps. The assumption of a (truly) incompressible fluid removes the speed of sound from the reference velocity $v_{\max}$. Therefore, the maximum time-step becomes

$$\Delta t_{\max} = \frac{\text{CFL}\ \Delta x}{|\boldsymbol{u}|_{\max}} \tag{4}$$

so that the same CFL number yields considerably larger values at very small Mach number. Typically,

$$\Delta t_{\max}^{incomp.} \simeq \frac{\Delta t_{\max}^{comp.}}{\text{Ma}}. \tag{5}$$

Conceptually, the sound speed is viewed as infinite in the incompressible model implying that pressure and velocity fields adapt instantaneously. Mathematically, an additional Poisson equation arises (by taking the divergence of the momentum equation) to describe the evolution of pressure (and in turn to project velocity in divergence-free/solenoidal space). One possibility, which is known to belong to the class of SIMPLE algorithms, solves the Poisson equation while the pressure and velocity fields are updated in a (semi-implicit) iterative process [PS72]. Another possibility is the artificial compressibility method (ACM), where a pressure term is substituted into the continuity equation via an isothermal equation of state. This circumvents the iterative stepwise update of the pressure and velocity fields, but reintroduces an artificial speed of sound [Cho67]. In that situation, pseudo-waves propagating with a finite speed are introduced to "distribute the pressure".

In the LB method, the distribution functions move from one lattice node to another during exactly one time-step, *i.e.* with a characteristic speed $c = \Delta x/\Delta t$. On the other hand, the propagation of sound is related to the effective transport of mass-density variations via the distribution functions. Therefore, the speed of sound and the speed of microscopic propagation are physically related (but not strictly equal because the distribution functions also undergo collisions). For a standard isothermal lattice, this relation is

$$c = c_0\sqrt{3} \tag{6}$$

where $c_0 = \sqrt{p/\rho}$ represents here an isothermal speed of sound directly linking the pressure and the mass density. Let us note that in the following, we will distinguish the *physical* speed of sound, e.g. $c_s \simeq 343$ m/s for air at standard conditions, from the possibly variable speed of sound c_0 introduced in the LB framework. As evidenced by Eq. (6), the speed of sound cannot be removed from the statistical approach. Thus, a commonly used technique to accelerate a LB simulation is to artificially decrease c_0, or equivalently, to increase the compressibility of the fluid. This is the same technique used to maximize the time-step in ACM. In fact the two approaches share a striking similarity [HDC02; Asi+12]. Obviously, the Mach number should not exceed a critical value (Ma = 0.3) to remain in the domain of validity where compressibility effects are weak.

In practice, to determine an appropriate artificial speed of sound, the maximum expected flow velocity is usually overestimated to allow for a certain room of maneuver and to prevent the Mach number from exceeding its maximal allowed value during the simulation. Moreover, in unsteady simulations the maximum velocity may vary by orders of magnitude. As a consequence the time-step of these simulations is often unnecessarily small. A solution is given by using an adaptive time-step, which can be changed throughout a simulation as a function of the current maximum flow velocity. This technique allows for an optimization of the time-step,

however, it has a side effect on the distribution functions. Being a probability measure to find a particle with a microscopic velocity c, this probability changes when the speed of sound is modified according to Eq. (6). In literature, very few studies exist that describe how to handle a change of the speed of sound. Under the name of *Mach number annealing*, it is demonstrated in [AHS03] that the Mach number may be changed artificially during a simulation while preserving the dynamics of the flow characterized by a Reynolds number (Re) and a Strouhal number (St). Unfortunately, the reader is deprived of the actual algorithm. This is not the case in [Thü+06], where a comprehensive description of the *adaptive time-stepping* rescaling operations is provided for the simulation of a gradient-driven free-surface flow. In particular, the off-equilibrium part of the distribution functions is reconstructed from the rescaled Maxwellian distribution in conjunction with a rescaling operation that is used in mesh refinement algorithms [FH98]. In [Lat07], the author introduces *adaptive time-stepping* for the regularized LB method. Here the populations are reconstructed entirely from the rescaled macroscopic variables. The readjustment of distribution functions $f(\boldsymbol{x}, t, \boldsymbol{c})$ due to a change in $\boldsymbol{c}$ is common to the aforementioned studies, whether in dimensional or non-dimensional space. However, the continuity of the pressure is not preserved in these algorithms. In the present study we will comment on this impact and propose a correction to optimize the use of *adaptive time-stepping* for unsteady weakly-compressible flows.

The paper is organized as follows. Sec. 2 contains a description of the physical impact of a sudden change in the speed of sound on the fluid dynamics. Sec. 3 provides a brief recap of the LB method. Sec. 4 presents the different algorithms that have been tested in this study. Sec. 5 shows results of each algorithm for three different test cases, while concluding remarks are given in Sec. 6.

2 Physical aspects of the adaptive time-stepping

The following considerations apply to an isothermal fluid satisfying the equation of state $rT_0 = c_0^2$ with r being the specific gas constant and c_0 an isothermal speed of sound. In this case, the governing gas kinetic equation under the Bhatnagar–Gross–Krook (BGK) approximation [BGK54] reads for the distribution function $f(\boldsymbol{x}, \boldsymbol{c}, t)$ as

$$\frac{\partial f}{\partial t} + (\boldsymbol{c} \cdot \boldsymbol{\nabla}) f + \left(\frac{\boldsymbol{f}_{ext}}{\rho} \cdot \boldsymbol{\nabla}_{\boldsymbol{c}} \right) f = -\frac{1}{\tau} (f - f^{eq}) \tag{7}$$

where the equilibrium distribution f^{eq} is represented by the *Maxwellian distribution*

$$f^{eq}(\boldsymbol{x}, \boldsymbol{c}, t) = \frac{\rho(\boldsymbol{x}, t)}{(2\pi c_0^2)^{3/2}} exp\left(-\frac{(\boldsymbol{c} - \boldsymbol{u}(\boldsymbol{x}, t))^2}{2c_0^2} \right). \tag{8}$$

$\boldsymbol{f}_{ext}$ represents a possible external force (per unit volume) acting on the fluid. Furthermore, we make the assumption that a particular flow is uniquely defined by its Reynolds number

$$\mathrm{Re} = \frac{UD}{\nu}$$

and, possibly, its Strouhal number

$$\mathrm{St} = \frac{fD}{U}$$

where U and D are respectively a reference velocity and length scale, f is a frequency of oscillation of the flow and ν is the kinematic viscosity of the fluid. The invariance of the Reynolds and Strouhal numbers implies that U and hence ν must remain unaffected when the speed of sound is changed. In addition, the theory provides a direct relation between the kinematic viscosity and the speed of sound, namely

$$\nu = \tau c_0^2 \tag{9}$$

where τ is the relaxation time that directly intervenes in the BGK collision operator of Eq. (7). It follows that changing the speed of sound requires modifying the relaxation time inversely in order to keep the viscosity constant. The relaxation towards equilibrium thus occurs at a larger time-scale when c_0 is decreased. The physical explanation is straightforward. Under the equation of state $rT_0 = c_0^2$, the speed of sound is directly related to the temperature. As a consequence a sudden drop in c_0 means that molecular activity is abruptly lowered. With the particles acting in a more sedate fashion, the collisional time-scale is increased.

On the other hand, the isothermal speed of sound expresses as

$$c_0^2 = \frac{p}{\rho} \tag{10}$$

which implies that *adaptive time-stepping* will lead to discontinuities in either the pressure or the density field. In order to determine the physical consequences of this relation, it is informative to consider the macroscopic equivalent to Eq. (7), which is obtained through a Chapman-Enskog multiple time-scale analysis up to second-order in the Knudsen and Mach numbers [Suc15; Krü+17]. Namely,

$$\begin{aligned} &\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}) = 0 \\ &\frac{\partial \rho \boldsymbol{u}}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}\boldsymbol{u}) = -\nabla p + \mu \Delta \boldsymbol{u} + \boldsymbol{f}_{ext}. \end{aligned} \tag{11}$$

By applying the chain rule to the momentum equation and accounting for the mass conservation, the above equations may be reformulated as

$$\frac{\partial \rho'}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}) = 0 \tag{12}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\, \boldsymbol{u} = -\frac{1}{\rho} \nabla \left(\rho' c_0^2\right) + \nu \Delta \boldsymbol{u} + \boldsymbol{a} \tag{13}$$

where the mass density is decomposed into $\rho(\boldsymbol{x}, t) = \rho_{\text{ref}} + \rho'(\boldsymbol{x}, t)$, ν is the kinematic fluid viscosity and $\boldsymbol{a} \equiv \boldsymbol{f}_{ext}/\rho$. By default, the above Navier-Stokes model is continuous in $\rho(\boldsymbol{x}, t)$. As a consequence, the pressure force (per unit mass) $-\nabla\left(\rho' c_0^2\right)/\rho$ in the momentum equation will be affected by an abrupt change of c_0. It is possible to derive a pressure-continuous Navier Stokes model, but in this case the density field has to be adapted. More precisely, if the speed of sound is changed from c_0 to

$$c_0^* = \lambda c_0 \tag{14}$$

the continuity of the pressure force per unit mass requires that

$$-\frac{c_0^2}{\rho} \nabla \rho' = -\frac{(\lambda c_0)^2}{\rho^*} \nabla \rho'^* \tag{15}$$

where – here and in the following – the superscript * denotes a quantity rescaled in response to a modified speed of sound. After some calculus (detailed in the Appendix A) we obtain the rescaling of the density

$$\rho^* = \rho_{\text{ref}} \left(\frac{\rho}{\rho_{\text{ref}}} \right)^{\frac{1}{\lambda^2}} . \tag{16}$$

In summary, we note the following changes as a consequence of a modified speed of sound: i) Decreasing the speed of sound with the purpose to increase the time-step also increases τ, which has a decelerating effect on the relaxation process. ii) By default, the Navier-Stokes model with *adaptive time-stepping* is density continuous. Nevertheless, a modification of the density field (Eq. (16)) allows us to render it pressure force (per unit mass) continuous and, therefore, leave Eq. (13) unaffected by the change of c_0. Let us note that it is here essential to consider the pressure force *per unit mass*, and not the pressure force per unit volume, to preserve the spatio-temporal evolution of the velocity itself, and not $\rho \boldsymbol{u}$, in coherency with the incompressible limit. On the other hand, injecting Eq. (16) into Eq. (12) yields a spurious source term in the mass conservation equation:

$$\frac{\partial \rho^{*\prime}}{\partial t} + \nabla \cdot (\rho^* \boldsymbol{u}) = \frac{\lambda^2 - 1}{\lambda^2} \rho^* \left(\nabla \cdot \boldsymbol{u} \right) \simeq \frac{\lambda^2 - 1}{\lambda^2} \rho_{\text{ref}} \left(\nabla \cdot \boldsymbol{u} \right). \tag{17}$$

Since λ is supposed to be close to unity and $\nabla \cdot \boldsymbol{u} \simeq 0$ in the weakly-compressible regime, this term is expected to remain small in practice.

3 The lattice Boltzmann model

Besides the physical consequences of *adaptive time-stepping* on the continuous Boltzmann equation (developed in Sec. 2), the discrete model requires some additional rescaling operations. For the sake of clarity, we shall start from the dimensional discrete-velocity Boltzmann equation with external force term

$$\frac{\partial f_\alpha}{\partial t} + (\boldsymbol{c}_\alpha \cdot \nabla) f_\alpha = -\frac{1}{\tau} \left(f_\alpha - f_\alpha^{eq} \right) + F_\alpha \tag{18}$$

where $\alpha = 1 \cdots q-1$ spans a discrete set of microscopic velocities (for a $DdQq$ lattice) and

$$f_\alpha^{eq} = \omega_\alpha \rho \left(1 + \frac{\boldsymbol{c}_\alpha \cdot \boldsymbol{u}}{c_0^2} + \frac{(\boldsymbol{c}_\alpha \cdot \boldsymbol{u})^2}{2c_0^4} - \frac{u^2}{2c_0^2} + \mathcal{O}(\text{Ma}^3) \right) \tag{19}$$

is the discrete second-order (in Mach number) Maxwellian distribution that is associated with isothermal lattice molecules; ω_α are lattice weights. According to [GZS02] the additional term that arises from the external force (per unit volume) $\boldsymbol{f}_{ext}$ may be written as

$$F_\alpha = \omega_\alpha \left(\frac{\boldsymbol{c}_\alpha - \boldsymbol{u}}{c_0^2} + \frac{\boldsymbol{c}_\alpha \left(\boldsymbol{c}_\alpha \cdot \boldsymbol{u} \right)}{c_0^4} \right) \cdot \boldsymbol{f}_{ext}. \tag{20}$$

The velocity moments are expressed as discrete sums such that

$$\sum_{\alpha=0}^{q-1} f_\alpha = \sum_{\alpha=0}^{q-1} f_\alpha^{eq} = \rho \tag{21}$$

$$\sum_{\alpha=0}^{q-1} \boldsymbol{c}_\alpha f_\alpha = \sum_{\alpha=0}^{q-1} \boldsymbol{c}_\alpha f_\alpha^{eq} = \rho \boldsymbol{u} \tag{22}$$

$$\sum_{\alpha=0}^{q-1} \boldsymbol{c}_\alpha \boldsymbol{c}_\alpha f_\alpha = \boldsymbol{P} + \mathcal{O}(\mathrm{Ma}^3) \tag{23}$$

where $\boldsymbol{P}$ is the momentum flux tensor. Similarly, the zeroth and first-order velocity moments of the forcing term are given by

$$\sum_{\alpha=0}^{q-1} F_\alpha = 0 \quad \text{and} \quad \sum_{\alpha=0}^{q-1} \boldsymbol{c}_\alpha F_\alpha = \boldsymbol{f}_{ext}. \tag{24}$$

Concerning the discretization in space and time, the left-hand-side of Eq. (18) is viewed as a substantial derivative, which is integrated (for each α) over a time-step Δt along the characteristics $\mathrm{d}\boldsymbol{x}_\alpha(t) = \boldsymbol{c}_\alpha \mathrm{d}t$. The collision operator and the force term are then approximated by a trapezoidal rule, which leads to a semi-implicit scheme. In order to render this scheme explicit, the following change of variable is applied

$$g_\alpha = f_\alpha + \frac{\Delta t}{2\tau}\left(f_\alpha - f_\alpha^{eq}\right) - \frac{\Delta t}{2} F_\alpha. \tag{25}$$

It can be noted that F_α is at least first-order in the Chapman-Enskog multiple time-scale expansion. Therefore, it could formally be added to the non-equilibrium part of the distribution functions. Here, we refrain from this manipulation and treat separately the three contributions to g_α as

$$g_\alpha = g_\alpha^{eq} + g_\alpha^{neq} - \frac{\Delta t}{2} F_\alpha \tag{26}$$

with $g_\alpha^{eq} = f^{eq}$ and $g_\alpha^{neq} \equiv (1 + \Delta t/2\tau)(f_\alpha - f_\alpha^{eq})$. In consequence, the moments of g_α^{eq} and g_α^{neq} remain unaffected by the external force, whereas

$$\sum_{\alpha=0}^{q-1} g_\alpha = \rho \quad \text{and} \quad \sum_{\alpha=0}^{q-1} \boldsymbol{c}_\alpha g_\alpha = \rho \boldsymbol{u} - \frac{\Delta t}{2} \boldsymbol{f}_{ext} \tag{27}$$

according to Eq. (24).

The resulting discrete Boltzmann equation then reads

$$\begin{aligned} g_\alpha(\boldsymbol{x}, t + \Delta t) &= g_\alpha(\boldsymbol{x} - \boldsymbol{c}_\alpha \Delta t, t) - \frac{\Delta t}{\tau_g}\left(g_\alpha(\boldsymbol{x} - \boldsymbol{c}_\alpha \Delta t, t) - g_\alpha^{eq}(\boldsymbol{x} - \boldsymbol{c}_\alpha \Delta t, t)\right) \\ &+ \left(1 - \frac{\Delta t}{2\tau_g}\right) \Delta t F_\alpha(\boldsymbol{x} - \boldsymbol{c}_\alpha \Delta t, t) \end{aligned} \tag{28}$$

which can here be simplified to

$$\begin{aligned} g_\alpha(\boldsymbol{x}, t + \Delta t) &= g_\alpha(\boldsymbol{x} - \boldsymbol{c}_\alpha \Delta t, t) - \frac{\Delta t}{\tau_g} g_\alpha^{neq}(\boldsymbol{x} - \boldsymbol{c}_\alpha \Delta t, t) \\ &+ \Delta t F_\alpha(\boldsymbol{x} - \boldsymbol{c}_\alpha \Delta t, t) \end{aligned} \tag{29}$$

with $\tau_g = \tau + \Delta t/2$ and g_α^{neq} defined by Eq. (26).

4 The algorithm

The algorithm that is presented in the following applies to the classical stream-and-collide formulation on an isothermal lattice, *e.g.* D2Q9, D3Q19, D3Q27. In this case, the time-step is given by

$$\Delta t = \frac{\Delta x}{\sqrt{3} c_0}. \tag{30}$$

In the *adaptive time-stepping* algorithm, the speed of sound is tailored in order to maintain a constant target Mach number Ma_t so that

$$c_0^*(t) = \frac{u_{\max}(t)}{\mathrm{Ma}_t} \tag{31}$$

with the maximum velocity $u_{\max}(t) \equiv \sup_{\boldsymbol{x}\in\mathcal{D}} \|\boldsymbol{u}(\boldsymbol{x},t)\|$ in the simulation domain. In unsteady simulations, this (artificial) numerical speed of sound is therefore a function of time. Nevertheless, at a particular instant, it is constant throughout the fluid and proportional to $u_{\max}$. This should not be confused with the (real-world) physical speed of sound $c_s(\boldsymbol{x},t)$ that may vary in time and space as a result of a non-constant temperature (due to compressibility effects).

The time-step (after tailoring the speed of sound) is then maximal for each iteration according to

$$\Delta t^*(t) = \frac{\mathrm{Ma}_t\ \Delta x}{\sqrt{3}\ u_{\max}(t)}. \tag{32}$$

In other words, the time-step is dynamically adapted so that the maximal Mach number of the flow reaches a desired target value. By considering the multiplicative factor λ (*cf.* Eq. (14)), we therefore obtain that the time-step is modified inversely as

$$\Delta t^* = \frac{1}{\lambda}\Delta t. \tag{33}$$

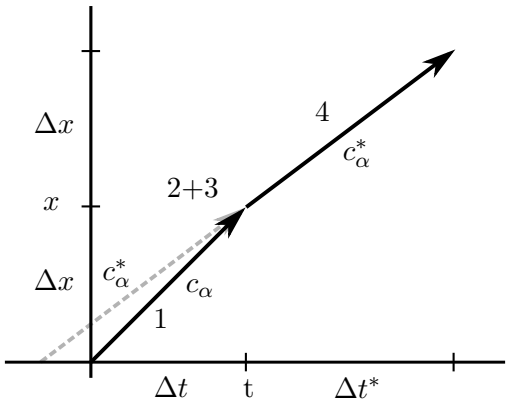


Figure 1: Sketch of the stream-and-collide algorithm with *adaptive time-stepping*. The different stages of the algorithm are (1) streaming (prior rescaling), (2+3) rescaling and collision – (4) streaming (after rescaling).

An illustration of the *adaptive time-stepping* algorithm is shown in Fig. 1. The successive stages of the algorithm are:

1. Standard streaming of the post-collision state $\widehat{g}_\alpha$ with the original time-step Δt

$$g_\alpha(\boldsymbol{x}, t) = \widehat{g}_\alpha(\boldsymbol{x} - \boldsymbol{c}_\alpha \Delta t, t - \Delta t) \tag{34}$$

2. Rescaling of the distribution functions as if they were streamed with velocity c^*_α (*cf.* the dashed arrow in Fig. 1) plus rescaling of the relaxation time
3. Collision of the rescaled distributions

$$\widehat{g}^*_\alpha(\boldsymbol{x}, t) = g^*_\alpha(\boldsymbol{x}, t) - \frac{\Delta t^*}{\tau^*_g} g^{*neq}_\alpha(\boldsymbol{x}, t) + \Delta t^* F^*_\alpha(\boldsymbol{x}, t) \tag{35}$$

4. Streaming of the distribution functions over the adapted time-step Δt^*

$$g^*_\alpha(\boldsymbol{x} + c^*_\alpha \Delta t^*, t + \Delta t^*) = \widehat{g}^*_\alpha(\boldsymbol{x}, t) \tag{36}$$

This sequence can *a priori* be repeated at each or more iterations leading to an optimized time step with respect to a target Mach number.

The remainder of this section will detail the algorithm. It should be noted that the above algorithm uses quantities in dimensional units, whereas the stream-and-collide algorithm is usually solved in non-dimensional lattice units for which $\Delta\tilde{x} = \Delta\tilde{t} = 1$. Somewhat against our intuition, the speed of sound in this framework remains constant and equal to $\tilde{c}_0 = 1/\sqrt{3}$. By simple reasoning it thus follows that the fluid velocity $\tilde{\boldsymbol{u}}$ (entering in the definition of the equilibrium distribution) requires rescaling to keep $\tilde{c}_0 \cdot \mathrm{Ma}_t$ constant when the time-step is adapted. We recall that Ma_t is the target Mach number that shall not be exceeded during a simulation. In other words, the rescaled maximum fluid velocity $\tilde{u}^*_{\max}$ (in lattice units) must remain constant.

4.1 The rescaling

The rescaling of g_α is achieved by separately adapting g^{eq}_α, F_α and g^{neq}_α. In lattice units the second-order equilibrium function of the lattice Boltzmann equation reads

$$g^{eq}_\alpha = \omega_\alpha \rho \left(1 + 3\left(\boldsymbol{e}_\alpha \cdot \tilde{\boldsymbol{u}}\right) + \frac{9}{2}\left(\boldsymbol{e}_\alpha \cdot \tilde{\boldsymbol{u}}\right)^2 - \frac{3}{2}\tilde{u}^2 + \mathcal{O}(\mathrm{Ma}^3)\right) \tag{37}$$

where $\tilde{\boldsymbol{u}} = \boldsymbol{u}\Delta t/\Delta x$ and $\boldsymbol{e}_\alpha = \tilde{\boldsymbol{c}}_\alpha$ represents the lattice link in the direction α. Changing the time-step from Δt to Δt^* thus changes the non-dimensional fluid velocity. The rescaling of g^{eq}_α is therefore achieved in three steps:

(i) The computation of the fluid velocity from Eq. (27)

$$\tilde{\boldsymbol{u}} = \frac{1}{\sum_{\alpha=0}^{q-1} g_\alpha} \left(\sum_{\alpha=0}^{q-1} g_\alpha \boldsymbol{e}_\alpha + \frac{\tilde{\boldsymbol{f}}_{ext}}{2}\right)$$

(ii) The rescaling of the fluid velocity $\tilde{\boldsymbol{u}}^* = \tilde{\boldsymbol{u}}/\lambda$ with λ defined by $c^*_0 = \lambda c_0$

(iii) The computation of the rescaled equilibrium function $g^{*\,eq}_\alpha(\rho^*, \tilde{\boldsymbol{u}}^*)$. At this point it is possible to consider either that the mass density is not affected, *i.e.* $\rho^* = \rho$, or that the mass density is modified according to Eq. (15) to preserve the continuity of the pressure force (per unit mass).

The contribution of an external force is taken into account similarly. In lattice units, the forcing term expresses as

$$\tilde{F}_\alpha = \omega_\alpha \left(3(\boldsymbol{e}_\alpha - \tilde{\boldsymbol{u}}) + 9(\boldsymbol{e}_\alpha(\boldsymbol{e}_\alpha \cdot \tilde{\boldsymbol{u}}))\right) \cdot \tilde{\boldsymbol{f}}_{ext} \tag{38}$$

In addition to updating the fluid velocity, the external force needs to be modified in the previous equation. This is done by considering

$$\tilde{\boldsymbol{f}}^*_{ext} = \boldsymbol{f}_{ext} \frac{(\Delta t^*)^2}{\Delta x} = \frac{1}{\lambda^2} \tilde{\boldsymbol{f}}_{ext}. \tag{39}$$

The rescaling of g^{neq}_α is not straightforward since its projection onto moment space includes non-hydrodynamic moments, whose rescaling is not intuitive. A possible solution would be to reconstruct a regularized non-equilibrium distribution, as suggested in [LC06; Lat07]. Rescaling by regularization relies on the continuity of $\sum_\alpha g^{neq}_\alpha \boldsymbol{c}_\alpha \boldsymbol{c}_\alpha$. Strictly speaking, this expression does not lead to any particular properties at the macroscopic level; only the second-order moment of the *genuine distributions* f^{neq}_α yields the deviatoric stress (at second order in the Knudsen number). In addition, regularization should be applied to each (local) time step in order not to mix regularized and non-regularized distributions, which can induce a significant computational overload, in particular when it comes to composite multi-resolution lattices [TRL14].

Here an alternative to the rescaling by regularization is favored by exploiting the continuity of the rate-of-strain tensor. This assumption is again motivated by the preservation of the velocity field and its spatial derivatives in Eq. (13).

Before detailing this method, we first establish the rescaling of the non-dimensional relaxation time $\tilde{\tau}_g = \tau/\Delta t + 1/2$. Clearly one possibility would be to replace τ by Eq. (9) and substitute for Δt^* and c^*_0. However, in order to avoid the return to dimensional space it may be preferable to compute $\tilde{\tau}^*_g$ directly from $\tilde{\tau}_g$. Therefore, considering that the viscosity must remain unaltered with

$$\nu = \left(\tilde{\tau}_g - \frac{1}{2}\right) \frac{c_0 \Delta x}{\sqrt{3}}, \tag{40}$$

and changing the speed of sound from c_0 to λc_0 directly leads to

$$\tilde{\tau}^*_g = \frac{1}{\lambda}\left(\tilde{\tau}_g - \frac{1}{2}\right) + \frac{1}{2}. \tag{41}$$

The rescaling of g^{neq}_α can now be processed as follows. A Chapman-Enskog analysis establishes that

$$\sum_{\alpha=0}^{q-1} g^{neq}_\alpha \boldsymbol{c}_\alpha \boldsymbol{c}_\alpha = -2\rho\tau_g c_0^2 \boldsymbol{S} + \mathcal{O}(\mathrm{Ma}^3) \tag{42}$$

where $\boldsymbol{S}$ represents the rate-of-strain tensor [Del01; Krü+17]. The continuity of $\boldsymbol{S}$ then gives

$$\frac{\sum_{\alpha=0}^{q-1} g^{neq}_\alpha \boldsymbol{c}_\alpha \boldsymbol{c}_\alpha}{\rho\tau_g c_0^2} = \frac{\sum_{\alpha=0}^{q-1} g^{*\,neq}_\alpha \boldsymbol{c}^*_\alpha \boldsymbol{c}^*_\alpha}{\rho^*\tau^*_g {c^*_0}^2} \tag{43}$$

which is equivalent (in lattice units) to

$$\frac{\sum_{\alpha=0}^{q-1} g^{neq}_\alpha \boldsymbol{e}_\alpha \boldsymbol{e}_\alpha}{\rho\tilde{\tau}_g \Delta t} = \frac{\sum_{\alpha=0}^{q-1} g^{*\,neq}_\alpha \boldsymbol{e}_\alpha \boldsymbol{e}_\alpha}{\rho^*\tilde{\tau}^*_g \Delta t^*} \tag{44}$$

and eventually yields

$$g_\alpha^{*\,neq} = \frac{1}{\lambda}\frac{\rho^*}{\rho}\frac{\tilde{\tau}_g^*}{\tilde{\tau}_g}\, g_\alpha^{neq} \tag{45}$$

by assuming that all the g_α^{neq}'s are rescaled by a same factor. This assumption is consistent with the idea that changing c_0 expands or contracts the velocity space isotropically. Therefore, there is a priori no reason to favor a particular direction. Let us clarify that in the case of a continuous density field $\rho^* = \rho$ in the above equation.

4.2 Summary of the algorithm

As seen previously, the implementation of *adaptive time-stepping* can be achieved in several ways that differ by the physical arguments used. An important issue is the entanglement of p and ρ via the speed of sound c_0, which unavoidably introduces a (small) discontinuity in either the mass or momentum transport equation. While previous studies are based on the default setting, which ensures the continuity of ρ but not of p, it is questionable whether this is always a good choice in particular when it comes to numerical accuracy. The main objective of this study is to specifically address this question . Attention has also been paid to verify that varying c_0 does not deteriorate the stability of the LB scheme itself. In the following, ATS will refer to *adaptive time-stepping* with the continuity of the density field, whereas "ATS with correction" will indicate the continuity of the pressure force by accounting for an additional step (*cf.* Eq. (16)) in the algorithm. An overview of the algorithm employed here is provided in Table 1.

Table 1: Stream-and-collide algorithm with *adaptive time-stepping*. The different operations are presented in lattice units (indicated by a tilde).

step (1)	Update g_α via streaming by using Eq. (34)
step (2)	Compute ρ, $\tilde{\boldsymbol{f}}_{ext}$, $\tilde{\boldsymbol{u}}$, and g_α^{eq} to obtain g_α^{neq} by using Eq. (26)
step (3)	Compute $\tilde{u}_{\max}$ and $\lambda = \tilde{u}_{\max}$ / $\tilde{c}_0 \mathrm{Ma}_t$
step (4)	Compute ρ^* by using Eq. (16) for ATS with correction
step (5)	Compute $\tilde{\boldsymbol{u}}^* = \tilde{\boldsymbol{u}}/\lambda$ and $\tilde{\boldsymbol{f}}_{ext}^* = \tilde{\boldsymbol{f}}_{ext}/\lambda^2$
step (6)	Compute $g_\alpha^{*\,eq}$ and $\tilde{F}_\alpha^*$ by applying Eqs. (37) and (38) with the rescaled variables
step (7)	Compute $\tilde{\tau}_g^*$ by using Eq. (41)
step (8)	Compute $g_\alpha^{*\,neq}$ using Eq. (45) together with g_α^{neq} from step (2)
step (9)	Compute $\widehat{g}_\alpha^*$ by using Eq. (35)

5 Validation

Our first two test cases are chosen to highlight the various benefits of the ATS algorithm. Let us recall that the algorithm presented in this study relies on a fixed Mach number that is defined prior to each simulation. This target Mach number is a trade-off between performance gain and stability. The classical BGK scheme is said to

remain stable for Mach numbers up to 0.4 [SC96]. Nevertheless, the time-step does not necessarily have to be at the stability limit for this method to be highly efficient. The strength of the algorithm is to adapt to the current maximum flow velocity. This is of interest for simulations, where the maximum velocity is either unknown or varies significantly. Explicit examples are natural convection phenomena in thermal flows and oscillating flows, respectively. Therefore, we present a thermal plume and a Womersley channel flow in two dimensions for both, the ATS and the "ATS with correction" algorithm. The former test case shall provide the reader with an intuitive understanding of this numerical technique, while the latter will examine in more detail the influence and the accuracy of the additional correction step for different simulation setups. In the absence of an oscillating pressure gradient, the Womersley channel flow becomes a Poiseuille channel flow, which constitutes the third test case. Designed as a channel entrance flow with a uniform inlet velocity, it features the standard boundary conditions used for many industrial applications and is worth being assessed in the context of an adaptive time-step. The correction of the density is found essential in that situation.

5.1 Natural convection in thermal flow

Although an isothermal lattice Boltzmann model is used in this study, it is possible to introduce a temperature-driven buoyancy force into the modelling via the Boussinesq hypothesis

$$\boldsymbol{f}_b(\boldsymbol{x},t) = \rho(\boldsymbol{x},t)\boldsymbol{g}\beta\left(T(\boldsymbol{x},t) - T_0\right) \tag{46}$$

where β is the coefficient of thermal expansion of the fluid, $\boldsymbol{g}$ is the gravitational acceleration and T_0 is the temperature at rest. We use a simple two-dimensional Gaussian distribution to introduce a variation in the temperature field, *i.e.*

$$T(\boldsymbol{x},t_0) = T_0 + exp\left(-\frac{x^2+y^2}{R^2}\right)\Delta T \tag{47}$$

where R is the radius of the hot spot and ΔT is the maximum temperature difference with respect to the background temperature T_0.

The interest of this test case lies in the time-dependent variation of the flow velocity. Initialized in a fluid at rest, *i.e.* $\boldsymbol{u}(\boldsymbol{x},0) = \boldsymbol{0}$, the thermal plume will accelerate until buoyancy and drag forces are balanced. The velocity field is taken into account by the Boltzmann equation with the force $\boldsymbol{f}_b$, while the development of the temperature field is governed by a simple advection-diffusion equation

$$\frac{\partial T}{\partial t} + (\boldsymbol{u}\cdot\nabla)\,T = \kappa\nabla^2 T \tag{48}$$

where κ is the thermal diffusivity of the fluid (air). The above equation is discretized using a finite-difference approach on the same spatio-temporal grid as the discrete Boltzmann scheme. Commonly referred to as an hybrid LB method, it is here easily brought in line with the *adaptive time-stepping* algorithm. In non-dimensional velocity space, a discrete form of Eq. (48) writes as

$$T(\boldsymbol{x},t+1) = T(\boldsymbol{x},t) - (\tilde{\boldsymbol{u}}\cdot\nabla_h)\,T + \tilde{\kappa}\Delta_h T, \tag{49}$$

where ∇_h and Δ_h stand for finite-difference gradient and Laplacian operators. Given that the lattice Boltzmann scheme already provides the rescaled bulk velocity $\tilde{\boldsymbol{u}}^*$, the only modification required is the rescaling of thermal diffusivity, *i.e.*

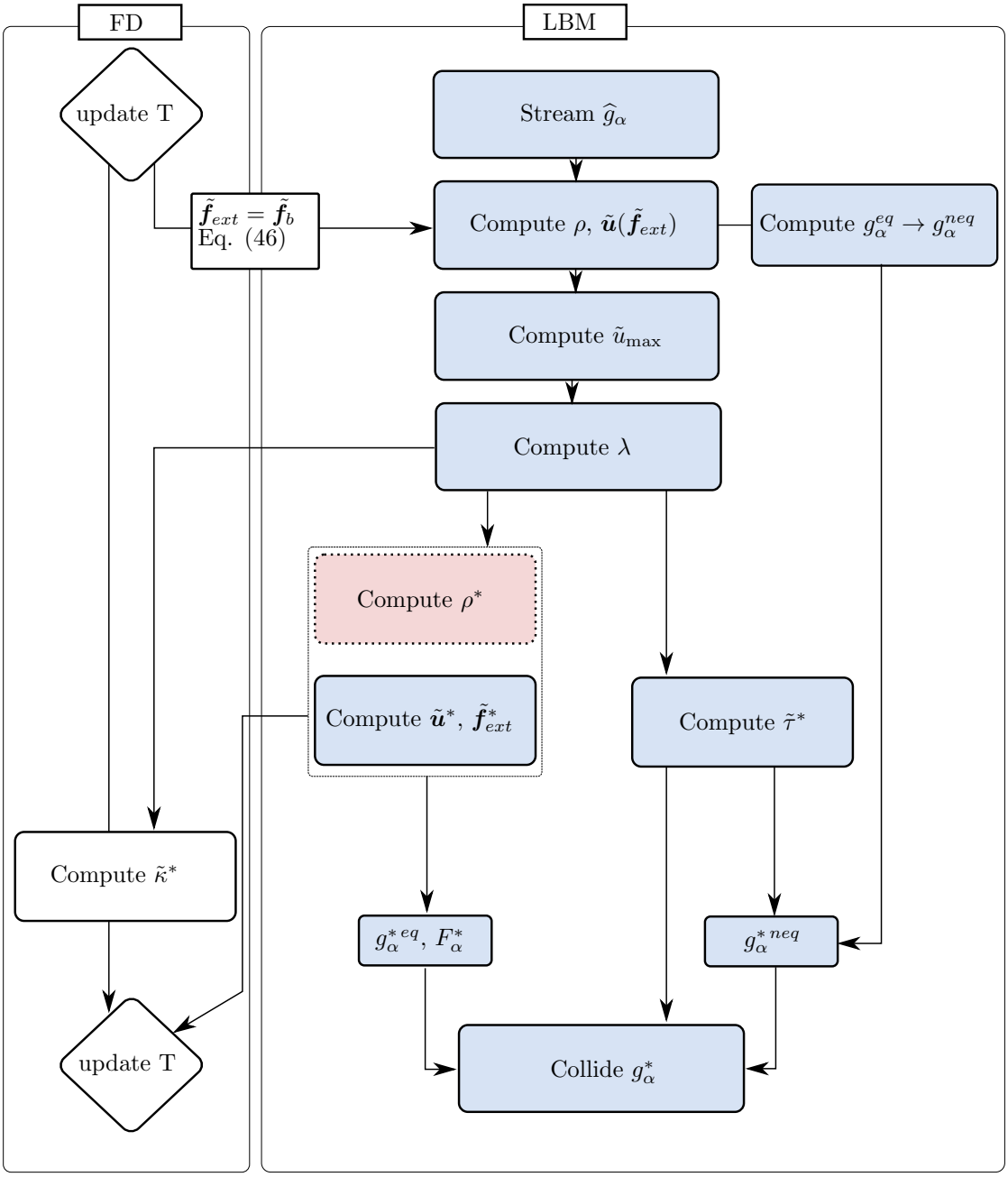


Figure 2: Diagram of the *adaptive time-stepping* (ATS) algorithm in lattice units. Blue boxes coincide with the original algorithm found in [Thü+06] that is extended here by accounting for an external force. White boxes indicate the extension to thermal dynamics using finite differences to solve a advection-diffusion equation for the temperature. Finally, the red box indicates the operation required on the density field (to ensure the continuity of the pressure force) that leads to the "ATS with correction" algorithm.

$\tilde{\kappa}^* = \nu\Delta t^*/Pr\Delta x^2$ with the Prandtl number $Pr = \nu/\kappa$. The complete algorithm is illustrated in Fig. 2.

The hot spot (or plume) is initialized in the center of a $(n_x \times n_y) = (250 \times 500)$ periodic domain. The spatial resolution is $\Delta x = 0.4$ m and $R = 50\Delta x$. The initial speed of sound is 343.2 ms^{-1} and the target Mach number was fixed at $\mathrm{Ma}_t = 0.15$. This target Mach number was reached gradually in order to avoid abrupt changes in the speed of sound at the beginning of the simulation. Specifically, we have used a caped decrease of c_0 according to

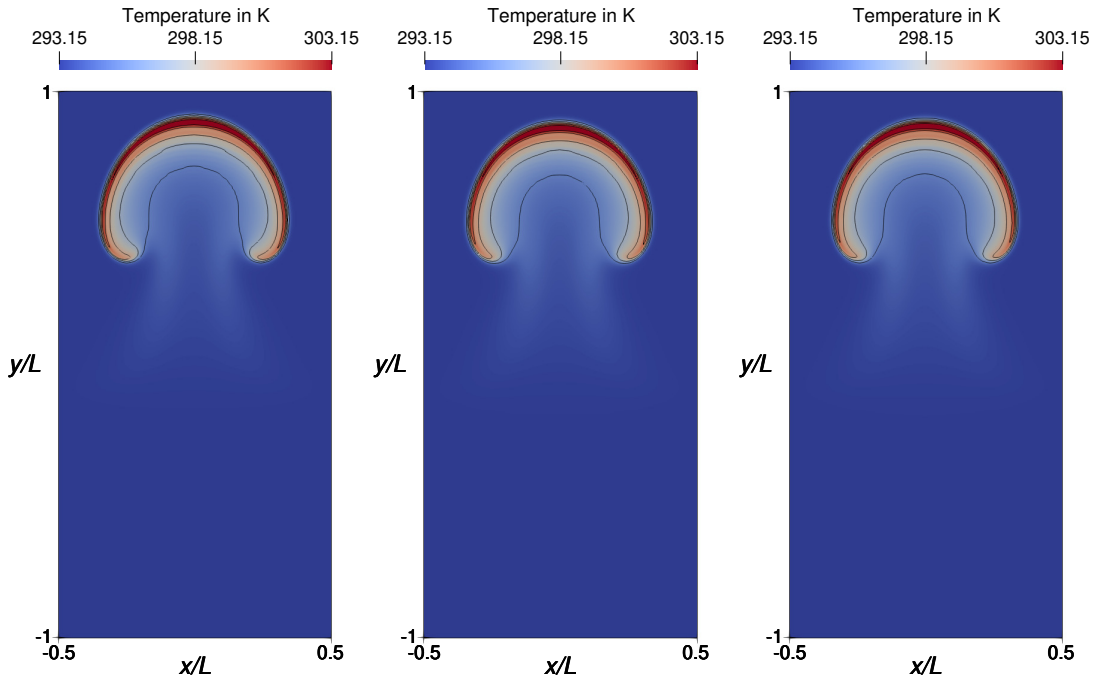
$$c_0^* = \max\left(\frac{u_{\max}}{\mathrm{Ma}_t}, c_0^{0.9}\right). \tag{50}$$

At first, the results of the two *adaptive time-stepping* algorithms are compared to a reference simulation with a constant time-step and for an initial temperature difference of $\Delta T = 10$ K at a reference temperature of $T_0 = 293.15$ K. The time-step is reevaluated every 10 iterations. Due to the variable time-step, the simulations are out of sync with each other. At 45.2 seconds, the simulation times match by four decimals corresponding to 3690 and 67310 iterations for the adaptive and constant time-stepping algorithms, respectively. The corresponding temperature fields are shown in Fig. 3. Qualitatively, the solutions agree with each other. The plume has reached the same elevation in the three simulations despite the (very) different time-steps. The contour lines of the temperature field indicate that the deformation of the rising plume is suitably captured as well. Let us mention that some unavoidable slight differences stem from the artificial compressibility of the accelerated simulations. However, these differences remain here very small at $\mathrm{Ma}_t = 0.15$. The velocity norms are compared in Fig 4. The maximum velocity is slightly higher in case of ATS, whereas the results from ATS with correction better agree with the reference (as evidenced later). Since the plume has covered the same distance in the three cases, it can only follow that the velocity field oscillate around the reference.

The evolution of the parameters related to the time-stepping is shown in Fig. 5. In the simulations with *adaptive time-stepping*, the target Mach number is reached after a transient of only about 100 iterations (Fig. 5a), during which the artificial speed of sound declines gradually according to the ramp defined in Eq. (50). After this transient, the speed of sound starts being controlled by $u_{\max}(t)$ and increases progressively as the plume rises (Fig. 5b). As a corollary, the time-step adapts itself during the transient to reach the target Mach number, then decreases to keep the Mach number constant as the flow develops (Fig. 5c). On the contrary, in the simulation with a constant time-step, the Mach number remains small reaching a maximum value of Ma = 0.013 at the end of the run. In Fig. 5c, the difference between the adapted and reference time-steps directly illustrate the significant gain obtained on the execution time for a given physical duration. Finally, let us notice that the ATS algorithm gives rise to some oscillatory behavior towards the end of the runs, particularly visible in the evolution of the time step. This behavior is much less pronounced when using ATS with correction.

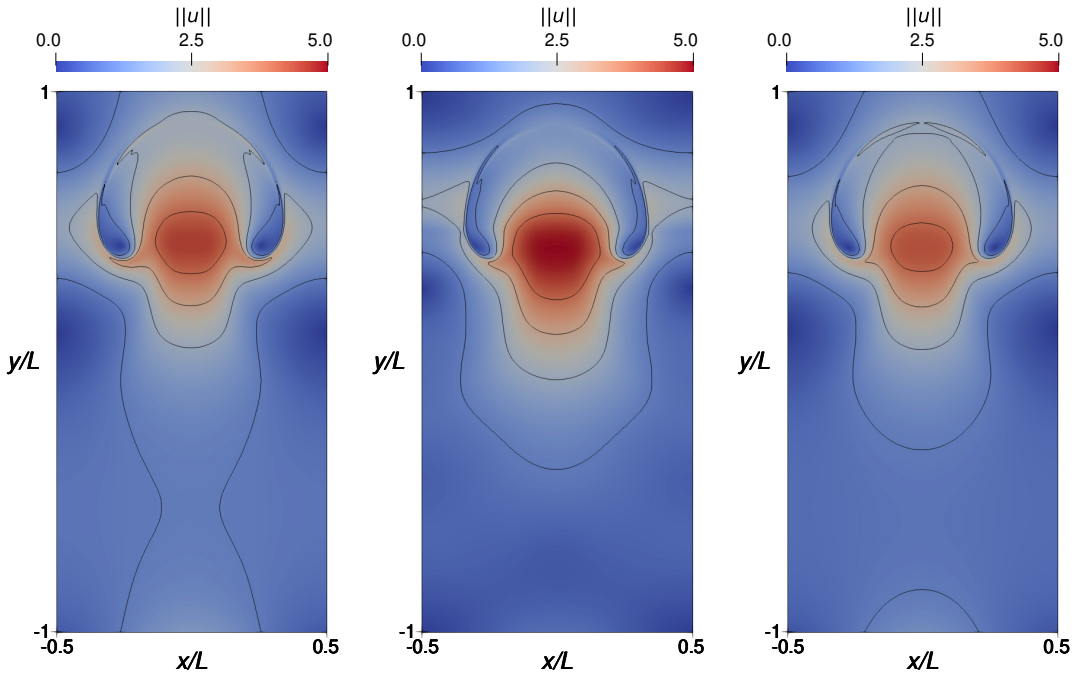
Due to the doubly periodic boundary conditions the thermal plume test case is also well suited to assess the mass conservation. By defining the relative mass error as

$$\frac{\frac{1}{n_x \times n_y}\sum_{x_i,y_j}\rho(x_i,y_j,t)}{\rho_{\mathrm{ref}}} - 1$$



(a) constant time-step (b) ATS (c) ATS with correction

Figure 3: Natural convection of a Gaussian spot of temperature elevation with $\Delta T = 10$ K after 45.3 s. $L = 100$ m and $\Delta x = 0.4$ m. *Left*: temperature field after 67310 iterations by using a constant time-step of $\Delta t = 6.729 \times 10^{-4}$ s. *Center*: after 3690 iterations by using ATS. *Right*: after 3690 iterations by using ATS with correction. The target Mach number is $\mathrm{Ma}_t = 0.15$.



(a) constant time-step (b) ATS (c) ATS with correction

Figure 4: The field of velocity magnitude $||\boldsymbol{u}|| = \sqrt{u_x^2 + u_y^2}$ in ms^{-1} is displayed at the same instant as in Fig. 3

it can be seen in Fig. 5d that the mass remains exactly conserved (as expected) in simulations with a constant time-step or ATS. In the case of "ATS with correction", the global mass error varies between -0.008% and 0.017% and remains negligible. Let us note that the the mass variation shows a steeper increase and decline than the time step, which reflects the fact that typically $\rho'^*/\rho' \simeq (\Delta t^*/\Delta t)^2$ according to Appendix A.

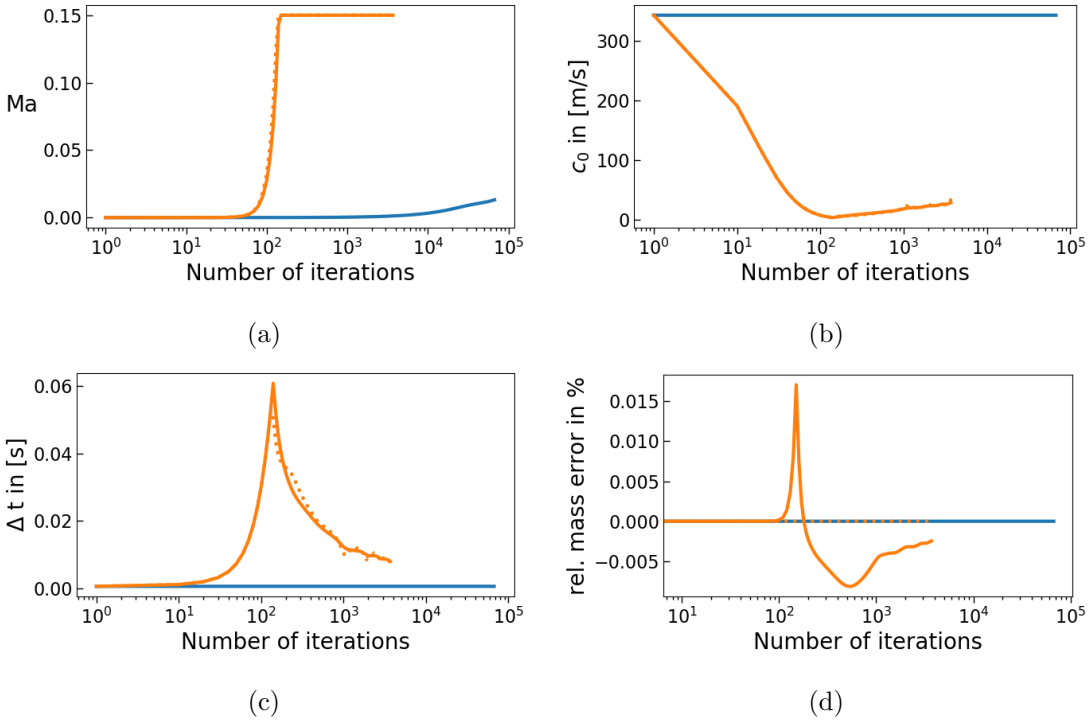
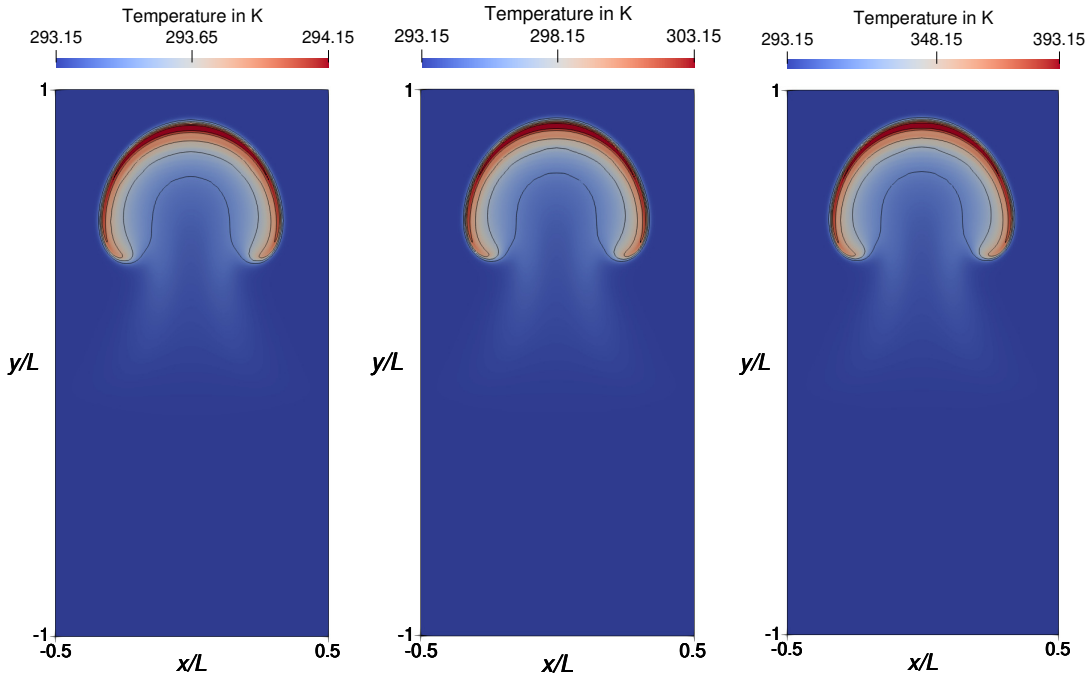


Figure 5: Evolution of the parameters related to the time-stepping during the rise of the thermal plume over a duration (physical time) of 45.3 s. (———): constant time-step; (·····): ATS; (———): ATS with correction. Let us notice that the number of iterations is presented on a logarithmic scale.

The robustness of ATS is examined in a second experiment by varying the initial temperature difference ΔT between the thermal plume and the surrounding fluid. In addition to $\Delta T = 10$ K, $\Delta T = 1$ K and $\Delta T = 100$ K are also considered. Each simulation was run for 3690 iterations as previously. The results are expected to be very similar with respect to the number of iterations. Qualitatively, a lower temperature difference leads to a slower rise of the plume, which is compensated for by a larger time-step. Conversely, a stronger difference has the opposite effect. More quantitatively, the dynamics results from the balance between the buoyancy and the drag, which justifies that $T - T_0$ varies typically as u. Therefore, the time-step adapts itself inversely to $T - T_0$, and the displacement of the plume remains (almost) constant with respect to the number of iterations. This behavior is correctly observed in Fig. 6, where the final temperature fields are displayed.

The adaptation of the speed of sound and time-step remain consistent for the



(a) $\Delta T = 1$ K (b) $\Delta T = 10$ K (c) $\Delta T = 100$ K

Figure 6: Natural convection of three Gaussian hot spots using ATS with initial temperature differences $\Delta T = 0$ K, 10 K and 100 K, after 3690 iterations. Iso-countours correspond for each figure to $T_0 + j\Delta T$ with $j = 0.2$, 0.4, 0.6, 0.8 and 1.

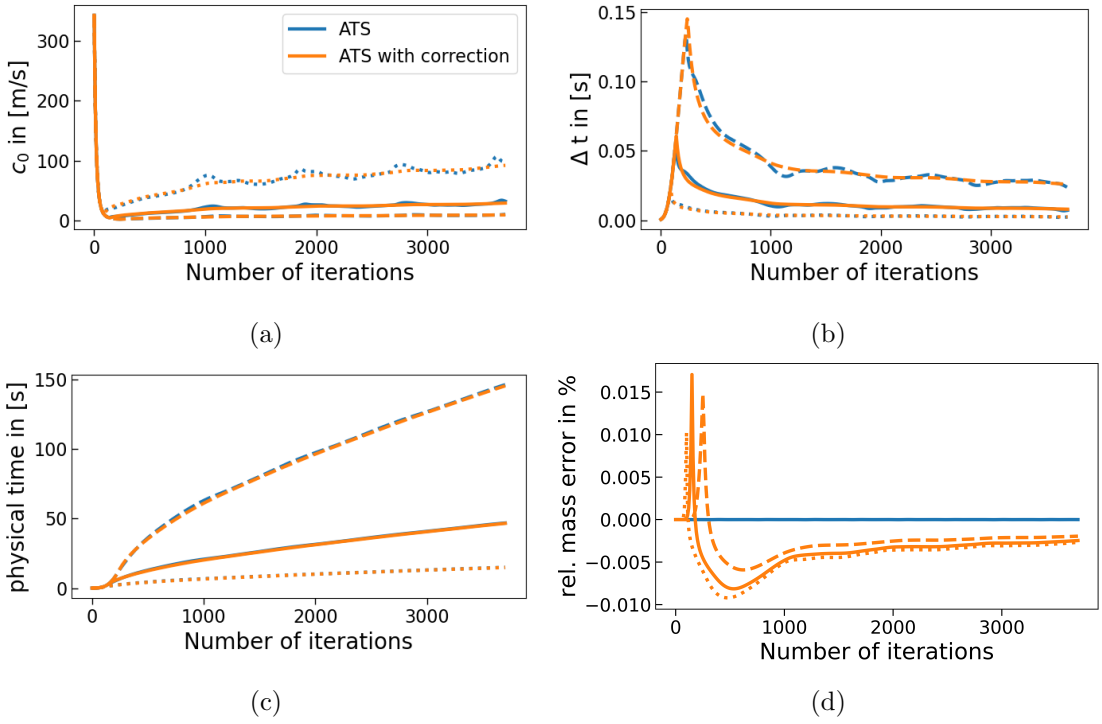


Figure 7: Evolution of the parameters related to *adaptive time-stepping* for various initial temperature differences of the thermal plume: (- - -): $\Delta T = 1$ K; (——): $\Delta T = 10$ K; (·····): $\Delta T = 100$ K.

different dynamics, as evidenced in Fig. 7. Finally, let us notice that the results obtained by ATS (without correction) contain once again a wavy perturbation, which is particularly pronounced in case of a high temperature difference. The relative mass error behaves differently at the beginning of the simulations but converges to a very small value (close to -0.002%) in all three scenarios.

5.2 Womersley flow

The Womersley flow is a pulsating flow in a 2D channel. In contrast to a Poiseuille flow, the pressure gradient oscillates according to

$$\frac{\partial P}{\partial x} \;=\; A\cos(\omega t) \tag{51}$$

where A is the amplitude and ω the pulsation. The latter is related to the pulsating period T as

$$\omega = \; 2\,\pi\,f \;=\; \frac{2\,\pi}{T}. \tag{52}$$

This problem has an exact solution in the laminar regime [IGC13] as the flow stays parallel to the walls and only depends on time and on the wall normal coordinate. Precisely,

$$\begin{aligned} u_x(y,t) &= \Re\left\{ \imath\,\frac{A}{\rho\,\omega}\left(1-\frac{\cos(\Lambda\,(\frac{2y}{L_y}-\,1))}{\cos(\Lambda)}\right) e^{\imath\,\omega\,t}\right\} \\ u_y &= 0 \end{aligned} \tag{53}$$

where L_y is the channel height and Λ is related to the so-called Womersley number α with

$$\Lambda^2 \;=\; -\imath\,\alpha^2 \quad \text{and} \quad \alpha^2 \;=\; \frac{L_y^2\,\omega}{4\,\nu}.$$

Similar to the Reynolds number, α relates inertial to viscous forces. Velocity profiles for different Womersley numbers α at constant amplitude A and pulsation ω are shown in Fig. 8. The velocity is normalized with respect to the maximum velocity U_0 achieved in a Poiseuille flow, *i.e.* for $\omega = 0$. For low values of α, the velocity profiles remain parabolic. The maximum velocity of the Womersley flow is achieved at the center-line of the channel and is of the same order of magnitude as U_0. With increasing α, the reluctance of the fluid to change direction increases so that it lags behind the pressure gradient. As a result the maximum velocity decreases and its occurrence moves towards the walls of the channel.

This multi-parametric problem offers many different possibilities to be implemented. In contrast to the previous example, there exists a maximum velocity for each configuration, which can be obtained from the analytical solution of the flow. For the sake of simplicity, we shall restrict our consideration to Womersley numbers that yield an overall maximum flow velocity located at the center-line of the channel, *i.e.* $U_{\max} = \max\{u_x(L_y/2,t) \;:\; 0 \le t < T\}$. At this point, it is important to distinguish between the overall maximum velocity $U_{\max}$ that is achieved during a period T, and the current maximum velocity $u_{\max}(t) = \max\{u_x(y,t) \;:\; 0 \le y \le L_y\}$. According to Fig. 8, $u_{\max}(t)$ can become zero at small α, which would result here in an infinitely large time-step. We assume that the Womersley number is sufficiently high to avoid this pitfall.

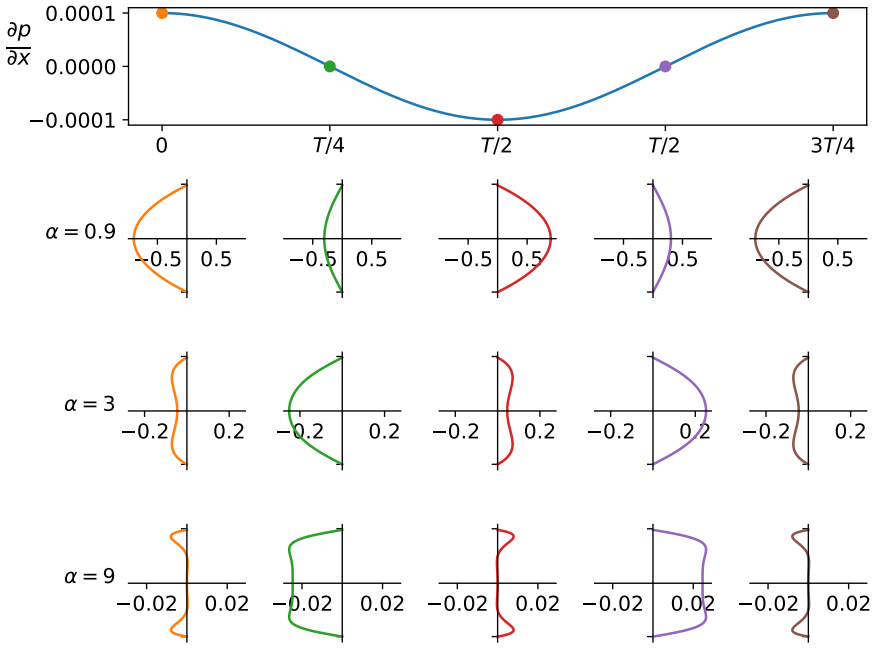


Figure 8: Temporal evolution of the pressure gradient (top) and the corresponding velocity profile $u_x(y,t)/U_0$ for different values of the Womersley number α.

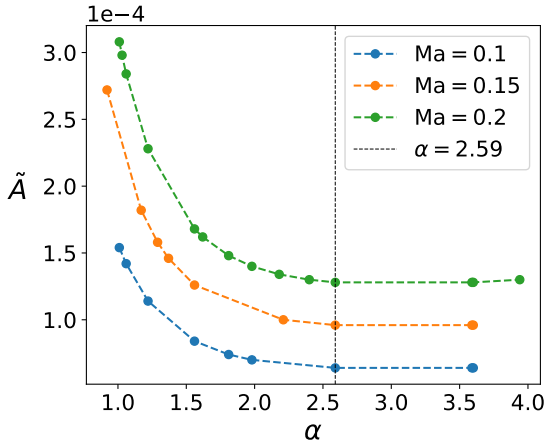


Figure 9: The non-dimensional amplitude $\tilde{A}$ as a function of the Womersley number α for different Mach numbers.

Our ATS algorithms require a target Mach number Ma_t as an input parameter. In that case, $\tilde{c}_0$ and consequently $\tilde{u}^*_{\max}$ (after rescaling) remain constant. In a non-dimensional framework, the scaling factor λ can therefore directly be obtained as $\lambda = \tilde{u}_{\max}/\tilde{U}_{\max}$ so that $\tilde{u}^*_{\max} = \tilde{u}_{\max}/\lambda = \tilde{U}_{\max}$.

In order to achieve three target Mach numbers $\mathrm{Ma}_t = 0.1$, 0.15 and 0.2, the remaining parameters are selected such that $\tilde{U}_{\max} = 0.1\tilde{c}_0$, $0.15\tilde{c}_0$ and $0.2\tilde{c}_0$, respectively. In accordance with [Nas+14], $Re = 100$ and $\tilde{T} = 5000$ to mimic real life flow phenomena that can be encountered in the smaller arteries of the human body. Next, we determine $\tilde{A}$ and α. According to Eq. (53), both parameters influence the velocity field and thus $\tilde{U}_{\max}$. In the ($\tilde{A}$ – α) space, the isolines corresponding the three target Mach numbers (with $\rho = 1$ and $\tilde{L}_y = 48$) are plotted in Fig. 9. In the light of our preceding considerations, we picked numerically the matching pairs for $\alpha = 2.59$ with an error tolerance of 0.02% on the Mach number. Finally, the viscosity is given by

$$\nu = \frac{\omega L_y^2}{4\alpha^2}. \tag{54}$$

In the case of the Womersley flow, the pressure gradient can either be established by an external body force, *i.e.*

$$\boldsymbol{f_{ext}}(t) = -A\cos(\omega t)\boldsymbol{e}_x \tag{55}$$

or by setting the respective inlet and outlet conditions with, by definition,

$$\frac{p_{\mathrm{in}}(t) - p_{\mathrm{out}}(t)}{L_x} = c_0^2 \frac{\rho_{\mathrm{in}}(t) - \rho_{\mathrm{out}}(t)}{L_x} = -A\cos(\omega t). \tag{56}$$

In the later case, by keeping the averaged density equal to the reference density, *i.e.*

$$\frac{\rho_{\mathrm{in}}(t) + \rho_{\mathrm{out}}(t)}{2} = \rho_{\mathrm{ref}}$$

the inlet and outlet boundary conditions for the mass density eventually write

$$\rho_{\mathrm{in}}(t) = \rho_{\mathrm{ref}} - \frac{AL_x}{2c_0^2(t)}\cos(\omega t) \quad \text{and} \quad \rho_{\mathrm{out}}(t) = \rho_{\mathrm{ref}} + \frac{AL_x}{2c_0^2(t)}\cos(\omega t). \tag{57}$$

For the sake of simplicity, these two strategies will be referred to as "body-force" and "inlet/outlet" from here onward. While the former applies directly a pressure gradient to every node of the fluid, the latter enforces the density at the inlet and outlet and induces the pressure gradient through the momentum equation. Therefore, it is expected that the correction of the pressure will have an impact on the results obtained by the inlet/outlet strategy. The simulation parameters are summarized in Table 2.

The domain size of the simulations is $(n_x \times n_y) = (24 \times 48)$ and the total number of iterations of each simulation is $5\tilde{T}$. The initial state is set to equilibrium with a zero velocity field and a reference density of unity. The same values are used to impose non-slip boundary conditions for the top and bottom of the domain [ZH97]. The initial speed of sound in all simulations is $U_{\max}/\mathrm{Ma}_t$. The *adaptive time-stepping* strategy is activated at $\tilde{t} = t/T = 0.75$ (after a brief transient to allow the oscillating flow to stabilize) and applied at every iteration.

Fig. 10 compares the evolution of the speed of sound and the maximum Mach number, as well as the flow velocity recorded at the center point ($L_x/2$, $L_y/2$) of the

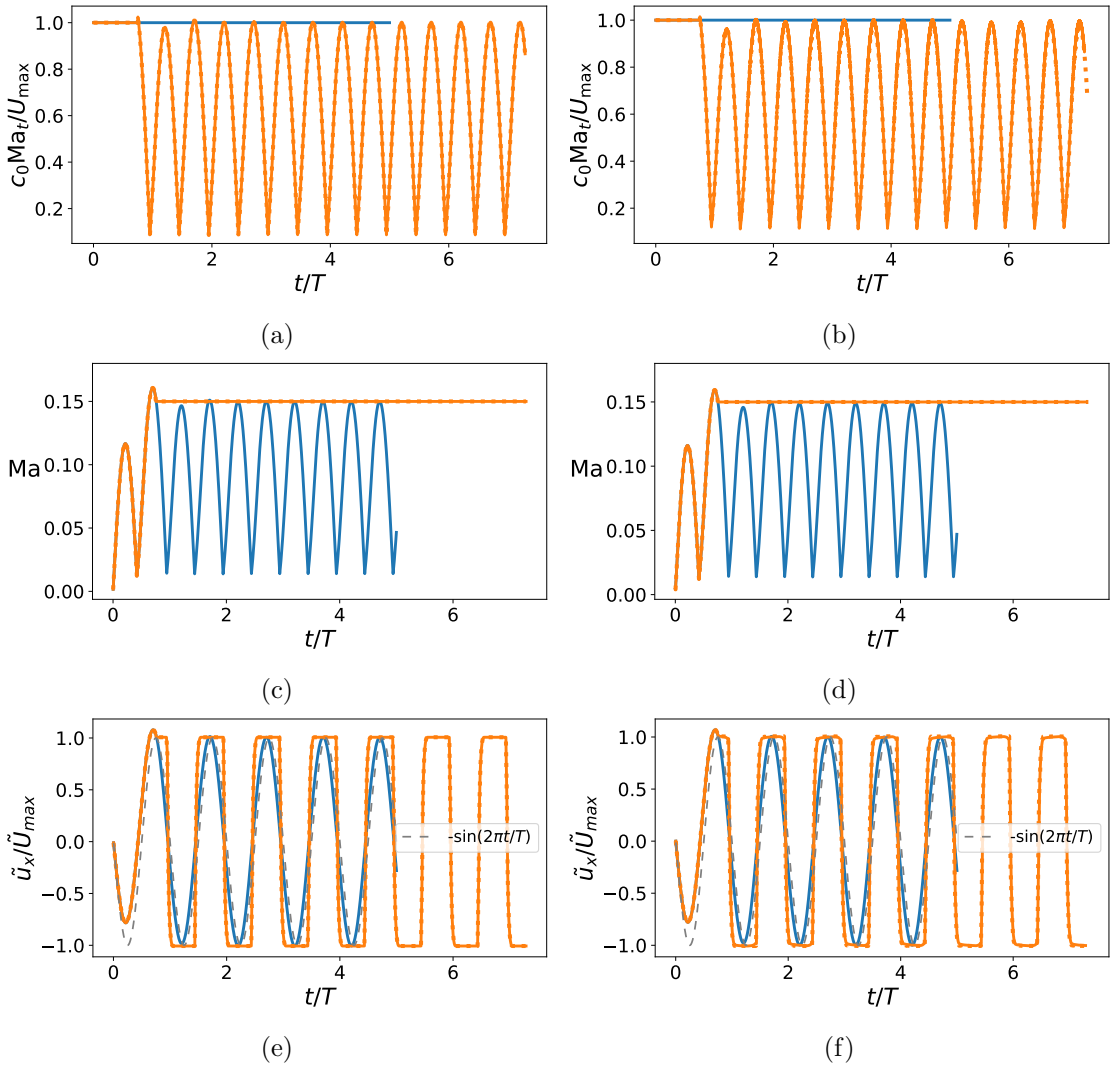


Figure 10: Temporal evolution of non-dimensional parameters related to to the *adaptive time-stepping* for the Womersley flow. The target Mach number is $\mathrm{Ma}_t = 0.15$. Recall that $U_{\max}$ is the (constant) overall maximum velocity and $u_x(t)$ refers to the probed velocity at the center of the oscillating flow. The Mach number is $\mathrm{Ma} = u_{\max}(t)/c_0^*$. *Left column:* body-force stirring; *Right column:* inlet/outlet density conditions. (———): constant time-step; (·····): ATS; (———): ATS with correction.

$\tilde{A}$	α	Re	Ma_t	$\tilde{T} = 2\pi/\tilde{\omega}$	$\tilde{T}_{\mathrm{sound}} = \tilde{L}_x/\tilde{c}_0$
6.4×10^{-5}	2.59	100	0.1	5000	26
9.6×10^{-5}	2.59	100	0.15	5000	26
1.28×10^{-4}	2.59	100	0.2	5000	26

Table 2: Parameters and characteristics of the simulations (in lattice units). $\tilde{T}$ is the period of oscillation of the pressure gradient. For comparison, $\tilde{T}_{\mathrm{sound}}$ is the time required by a local perturbation of density to propagate through the channel. The condition $\tilde{T}_{\mathrm{sound}} \ll \tilde{T}$ ensures that the pressure field has time to equilibrate for each value of the oscillating forcing.

domain. Only results for the intermediate Mach number ($\mathrm{Ma}_t = 0.15$) are presented for both the body-force and the inlet/outlet strategies. The results obtained for the two other target Mach numbers were very comparable.

In the non-adaptive case, the speed of sound remains constant at $c_0 = U_{\max}/\mathrm{Ma}_t$. When *adaptive time-stepping* is used, c_0 oscillates between $U_{\max}/\mathrm{Ma}_t$ and a minimum value, which is identical (within 5%) for the two forcing strategies (see Figs. 10a and 10b). As a corollary, the Mach number remains constant when *adaptive time-stepping* is used, whereas it oscillates when the time-step is kept constant (see Figs. 10c and 10d). The lowest value that $u_{\max}(t)$ attains during a cycle is approximately $U_{\max}/10$. Finally, the non-dimensional velocity $\tilde{u}_x$ at the center point of the domain normalized by $\tilde{U}_{\max}$ is shown in Figs. 10e and 10f. In the absence of *adaptive time-stepping*, $\tilde{u}_x$ closely follows a sinusoidal curve. Once the Mach number is locked to optimize the time-step, $\tilde{u}_x/\tilde{U}_{\max}$ behaves as a periodic step function. Therefore, the maximum gain attainable for the Womersley flow can be approximated by the ratio between the area of a step function and the half-period of a sine function, *i.e.* $\pi/2 \approx 1.57$. With the same number of iterations, the number of periods simulated with the proposed ATS algorithms is increased by a factor of 1.53, which is very close to the optimal gain.

So far, the forcing strategy (body-force or inlet/outlet conditions) and the correction of the pressure field have not revealed any notable differences in the results. In order to gain quantitative insights about the accuracy of our ATS algorithms, we measured the (normalized) L_2-norm of the velocity error defined as

$$\varepsilon_u(t) = \frac{\|u - \bar{u}\|_{L_2}}{\|\bar{u}\|_{L_2}} = \sqrt{\frac{\sum_{y_j=1}^{n_y} \left(u_x(L_x/2, y_j, t) - \bar{u}_x(y_j, t)\right)^2}{\sum_{y_j=1}^{n_y} \bar{u}_x(y_j, t)^2}} \tag{58}$$

where $\bar{u}_x$ is the analytical solution given by Eq. (53).

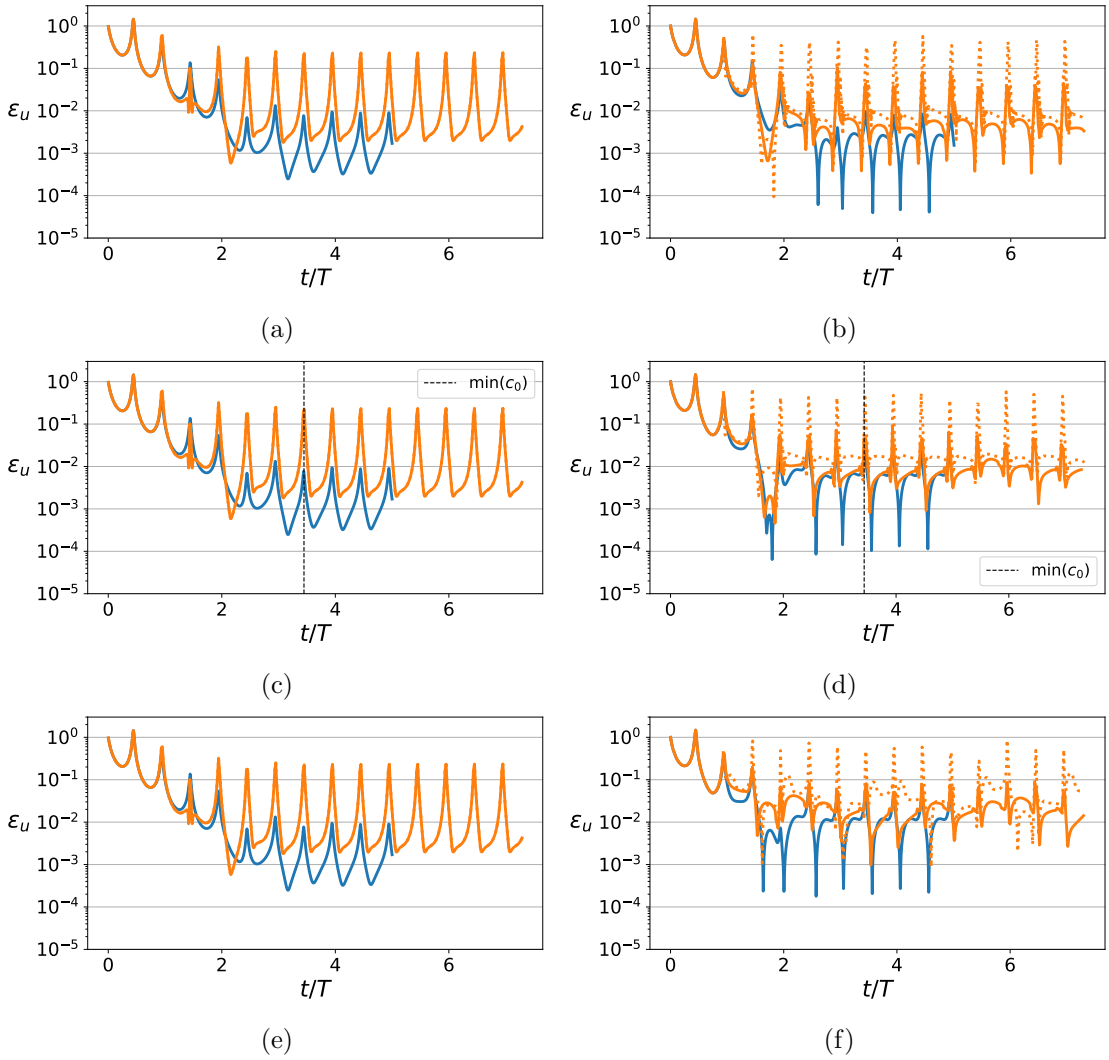


Figure 11: Temporal evolution of the L_2-norm of the velocity error for standard and *adaptive time-stepping* at different Mach numbers. *Top row:* $\mathrm{Ma}_t = 0.1$; *Middle row:* $\mathrm{Ma}_t = 0.15$; *Bottom row:* $\mathrm{Ma}_t = 0.2$. *Left column:* body-force stirring; *Right column:* inlet/outlet density conditions. (———): constant time-step; (·····): ATS; (———): ATS with correction.

Fig. 11 shows the error with respect to t/T for the different Mach numbers and forcing strategies. In all cases, a transient error decays rapidly over the first two cycles. When the flow is controlled via a (uniform) body force (see Figs. 11a, 11c and 11e), the error of the *adaptive time-stepping* does not depend on the Mach number and the correction of the pressure has no influence. In that case, the pressure gradient is mainly monitored by the body force and not by the density variations. The error is roughly one order of magnitude higher than the one achieved with a constant time-step, and peaks when the speed of sound is at a local minimum, *i.e.* during the

flow reversal. The situation is different when considering an inlet/outlet density (or pressure) condition. In that case, the error increases with the Mach number, which is understandable since the flow equilibrates through pressure waves. Importantly though, *adaptive time-stepping* is significantly improved when a correction of the pressure field is applied. As a result, the error remains of the same order as the error encountered in a simulation with a constant time-step, while being about 1.5 times faster. Let us notice that during the flow reversal (when c_0 reaches a minimum), the relative error spikes as the denominator in the error estimate (Eq. (58)) approaches zero.

In order to further assess the accuracy of our two ATS algorithms the simulations of the Womersley flow were repeated on a coarser (12×24) and a finer (48×96) grid with a period $\tilde{T}$ equal to 2500 and 10000 time-steps, respectively. The error is measured by integrating the L_2-norm of the difference with the analytical solution over five periods, starting at $t = 4T$ to exclude the transient phase. First, the target Mach number was kept at $\mathrm{Ma}_t = 0.15$ to remain consistent with the previous results. The convergence rates (as a function of the grid resolution) are shown in Fig. 12. In the case of a body force, the correction of the ATS algorithm has no influence on the error. The convergence rate compares well with that of the constant-time-step simulation, which is, however, lower than the second-order usually attributed to LB simulations. When the pulsatile flow is established through the inlet/outlet boundary conditions, the error at a target Mach number of 0.15 does not dependent on the grid resolution. This behavior arises from the fact that the analytical solution considers an incompressible fluid, while in our LB simulations the information travels from the boundaries through the domain at a finite speed. This latency establishes a phase shift that dominates the overall error. After each flow reversal the numerical solution overtakes the analytical one. To better estimate the convergence rate, and get ride of compressibility effects, the target Mach number was therefore reduced by a factor of ten. The results are shown in Figs. 12d and 12d. In the case of a body force this velocity reduction led to a shift of the error curves by the same factor, the convergence rate remaining unchanged. When inlet/outlet density conditions are used the situation is different. A second-order convergence is recovered in the case of a constant time-step. Importantly, when "ATS with correction" is applied, we also observe a convergence even though at a slightly smaller rate, whereas ATS without correction still contains an error that is independent of the grid resolution. In summary, these results show that the correction applied to the density improve the accuracy in all situations, and significantly increase the convergence rate of the *adaptive time-stepping* algorithm in the case of inlet/outlet boundary conditions.

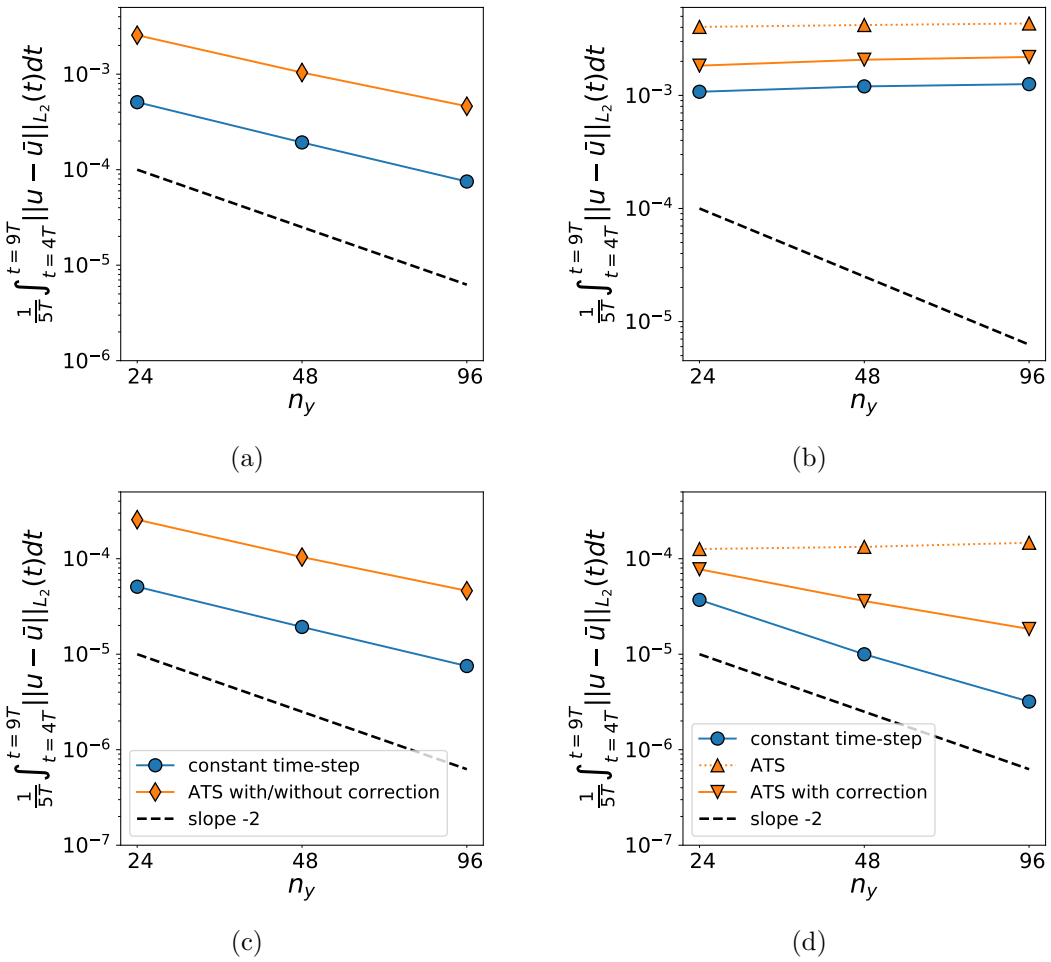


Figure 12: Convergence rate of the ATS algorithms with body-force *(left column)* and input/output boundary condition *(right column)*. The rows contain data from different Mach numbers. *Top row*: $\mathrm{Ma}_t = 0.15$, *bottom row*: $\mathrm{Ma}_t = 0.015$.

5.3 Channel entrance flow

The previous observations led us to examine a final numerical experiment that specifically addresses the issue of the pressure correction. By using the same channel height (L_y) and Reynolds number as in the previous setup, the length of the channel was increased by a factor 20 to simulate a channel entrance flow [Dur+05]. This flow scenario is achieved with a Dirichlet boundary condition for the velocity and the pressure at the inlet and the outlet, respectively. Measurements are performed at the position $x = 0.8L_x$ where the flow reaches a Poiseuille parabolic velocity profile. In order to improve the convergence and also to make the use of *adaptive time-stepping* relevant, the following ramp (in time) was considered for the velocity

at the inlet

$$
\begin{aligned}
U_{\text{in}}(t) &= \sin\left(\frac{\pi}{2}\frac{t}{T_{\text{ramp}}}\right) U_{\text{bulk}} \qquad \text{for} \quad t \leq T_{\text{ramp}} \\
U_{\text{in}}(t) &= U_{\text{bulk}} \qquad\qquad\qquad\qquad\quad \text{for} \quad t > T_{\text{ramp}}
\end{aligned}
\tag{59}
$$

with $U_{\text{bulk}} = 0.75U_0$ and $\tilde{T}_{\text{ramp}} = 10000$ being a fifth of the total number of iterations $\tilde{T}_{\text{tot}}$. Let us mention that U_{bulk} was fixed such that the maximum velocity at $x = 0.8L_x$ is $U_{max} = U_0$ in the asymptotic steady regime, *i.e.* when $t \gg T_{\text{ramp}}$. Initially, the velocity in the channel is zero and the pressure is uniform at an arbitrary reference value. This reference pressure is kept as the outlet boundary condition during the simulation. Finally, to avoid an impractically small speed of sound in the ATS algorithms, the time-step was locked until the Mach number at the inlet was greater than 0.03. Fig. 13 summarizes the time evolution of the Mach number, the

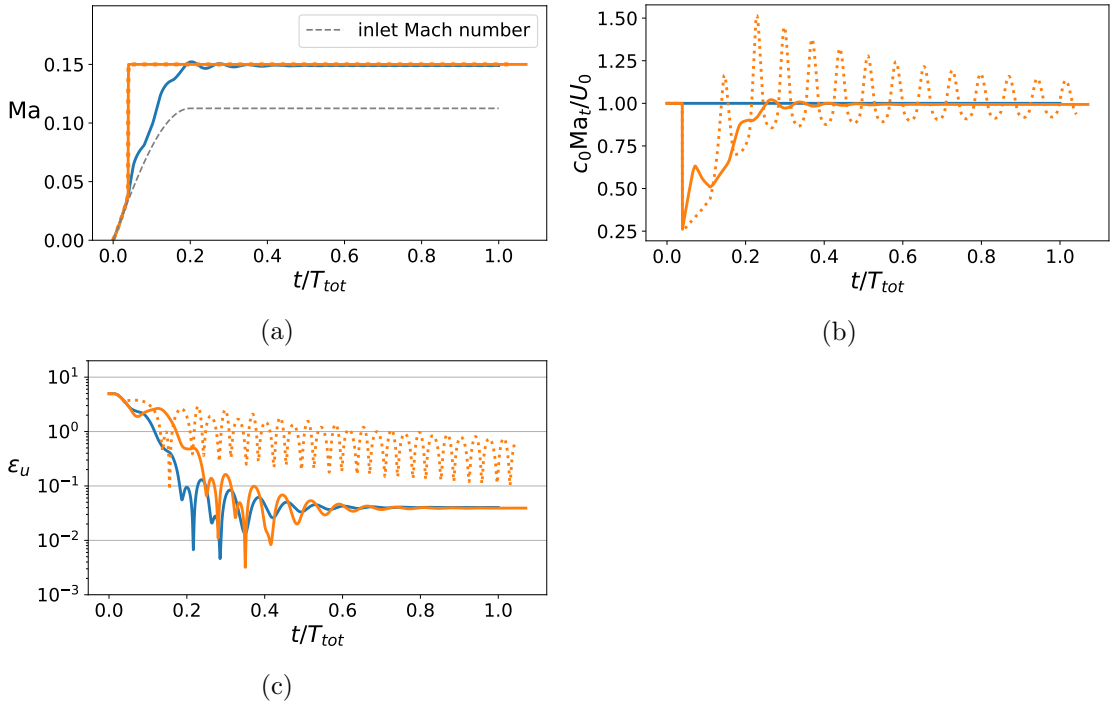


Figure 13: Time evolution of the Mach number (a), normalized speed of sound (b) and L_2-norm of the velocity error (c) for standard and *adaptive time-stepping* in a channel entrance flow simulation. Signals are recorded at $x = 0.8L_x$. (———): constant time-step; (·····): ATS; (———): ATS with correction.

normalized speed of sound and the L_2-norm of the velocity error defined by Eq. (58) with the Poiseuille profile $\bar{u}(y) = 4U_0(L_y - y)y/L_y^2$. At the beginning, the Mach number (at $x = 0.8L_x$) follows the Mach number ramp imposed at the inlet in the three simulations. When *adaptive time-stepping* is enabled, the Mach number jumps by construction to the target value $\text{Ma}_t = 0.15$. Conjointly, the parabolic Poiseuille profile begins to develop so that the Mach number in the standard algorithm starts

to deviate from the Mach number at the inlet (U_{bulk}/c_0) to converge to the asymptotic value $\text{Ma}_t = U_0/c_0$. The speed of sound remains constant in the standard approach, while it drops instantly once it is unlocked in the ATS simulations. When the pressure correction is used, the speed of sound consistently recovers its initial value rapidly. On the contrary, without the correction of the pressure, the speed of sound overshoots and enters an oscillatory pattern that slowly stabilizes but remains present throughout the remainder of the simulation. Such important artefact does not occur when the pressure correction is applied. Comparing the error, it becomes apparent that the "ATS with correction" simulation remains consistent with the constant time-step simulation, while the ATS simulation without the pressure correction does not converge at all. The normalized velocity profiles at $x = 0.8L_x$ at the end of

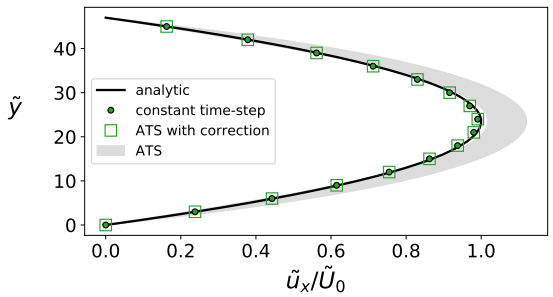


Figure 14: Normalized velocity profiles across the channel at $x = 0.8L_x$ obtained with constant and adaptive time-stepping. The grey area indicates the velocity interval, over which the solution of the ATS algorithm oscillates during the last period, *i.e.* between $t/T_{tot} = 1.013$ and $t/T_{tot} = 1.03$.

the simulation are displayed in Fig. 14). We observe that the results obtained with the standard and "ATS with correction" algorithms match the analytical Poiseuille profile. On the other hand, without the correction of the pressure, the discrepancy is much higher.

6 Conclusion

This study presents a physically consistent framework for *adaptive time-stepping* in the LB method, which constitutes an interesting technique to speed-up nearly-incompressible unsteady simulations. Despite being employed in a handful of studies, it is somewhat surprising that this technique has not been used to a wider extent by the LB community.

A critical aspect of *adaptive time-stepping* is the inevitable introduction of an error in either the mass or the momentum conservation due to an indirect entanglement of density and pressure variations through the time-step. A natural reluctance to violate the conservation of mass is probably one of the reasons why in previous studies the continuity of the mass density was preferred, and the error always passed into the momentum equation. Here, we investigate the opposite by preserving the pressure force (per unit mass) $-c_0^2 \boldsymbol{\nabla}(\log \rho)$ while sacrificing the continuity of the density field. Interestingly, it was shown that even in the case of natural convection, where the two driving forces are buoyancy and drag, the results are better when

the error is attributed to the mass density. In channel flows with classical velocity inlet and pressure outlet conditions, the correction of the pressure is essential for convergence. Otherwise, some oscillatory artefacts appear in the velocity field that greatly deteriorate the accuracy of the solution.

A physical explanation for these observations is not straightforward. Nevertheless, an indication may be given by comparing the errors made either on the density or the pressure force when varying the time step. Let us first introduce the small parameter

$$\varepsilon = \frac{\Delta t^* - \Delta t}{\Delta t} \tag{60}$$

quantifying the (relative) change of the time-step. The continuity of the density field ($\rho^* = \rho$) yields a relative error on the pressure force that varies as

$$\left(\frac{p^* - p}{p}\right) \sim \left(\frac{c_0^{*2} - c_0^2}{c_0^2}\right) \sim \varepsilon. \tag{61}$$

On the other hand, the continuity of the pressure force (per unit mass), which is equivalent to Eq. (16), gives at leading order a relative error on the density field

$$\left(\frac{\rho^* - \rho}{\rho}\right) \sim \varepsilon \, \frac{\rho - \rho_{\text{ref}}}{\rho} \sim \varepsilon \, \text{Ma}^2, \tag{62}$$

where the approximation $\rho' * \approx (1/\lambda^2)\rho'$ from appendix A is employed. Let us note that this qualitative estimation is consistent with the source term arising in Eq. (17) by further considering that $(\lambda^2 - 1)/\lambda^2 \sim \varepsilon$ and $(\nabla \cdot \boldsymbol{u}) \sim \text{Ma}^2$ in the weakly compressible regime. For a small Mach number, this relative error is therefore smaller than the one obtained by keeping the density continuous. Also, if ε is smaller or equivalent to Ma (which is a valid assumption) the error introduced by "ATS with correction" varies typically as $\mathcal{O}(\text{Ma}^3)$. This provides a plausible justification for the advantage of considering the continuity of the pressure force (per unit mass) in the *adaptive time-stepping*.

Using the here presented algorithm for high-Reynolds-number flows will certainly exceed the stability limit of the BGK collision model. Remedy may, for example, be provided by using advanced regularized collision models such as those obtained by *recursive regularization* [Mal15] or *hybrid recursive regularization* [JMS19]. These techniques rely on the reconstruction of first-order (in a multiple time-scale expansion) non-equilibrium distributions $g_\alpha^{(1)}$ by using their statistical moments. These later are not obtained by a summation over the velocity space but rather from a recursive relation between moments of increasing order, *i.e.* $\boldsymbol{a}_1^{(n)} = \boldsymbol{a}_1^{(n-1)}\boldsymbol{u}$ for $n \geq 3$. *Adaptive time-stepping* can be brought easily in line with these advanced models. Indeed, the second-order moment of $g_\alpha^{*\,neq}$ (*cf.* Eq. (45)) yields $\boldsymbol{a}_1^{*\,(2)}$ while $\tilde{\boldsymbol{u}}^* = \tilde{\boldsymbol{u}}/\lambda$. Higher moments are straightforwardly rescaled through the recursive relation.

To our best knowledge, this is the first study that proposes a correction of the pressure by accepting small compressibility errors in the solution. We believe that *adaptive time-stepping with correction* constitutes a valuable proposal to reduce the execution time of nearly-incompressible flows without notably altering the accuracy of the simulation. No prior knowledge about the maximum velocity is required and the speed up can be considerable by optimally adapting on the flow dynamics. This is particularly true for biological flows, transient thermal flows and oscillating flows in general, where the maximum velocity undergoes large variations.

Acknowledgment

This work is part of a scientific collaboration including CS-GROUP, Renault, Airbus, Ecole Centrale de Lyon, CNRS and Aix-Marseille University, and it is related to the development of the software ProLB^{TM}.

References

[AHS03] A. M. Artoli, A. G. Hoekstra, and P. M.A. Sloot. "Accelerated lattice BGK method for unsteady simulations through Mach number annealing". In: *International Journal of Modern Physics C* 14.6 (2003), pp. 835–845.

[Asi+12] Pietro Asinari et al. "Link-Wise Artificial Compressibility Method". In: *J. Comput. Phys.* 231.15 (2012), pp. 5109–5143.

[BGK54] P. L. Bhatnagar, E. P. Gross, and M. Krook. "A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems". In: *Physical Review* 94.3 (1954), pp. 511–525.

[CFL28] R Courant, K Friedrichs, and H Lewy. "Über die partiellen Differenzengleichungen der mathematischen Physik". In: *Mathematische Annalen* 100.1 (1928), pp. 32–74.

[Cho67] Alexandre Joel Chorin. "A numerical method for solving incompressible viscous flow problems". In: *Journal of Computational Physics* 2.1 (1967), pp. 12–26.

[CK06] Shyam S. Chikatamarla and Iliya V. Karlin. "Entropy and Galilean invariance of lattice Boltzmann theories". In: *Physical Review Letters* 97.19 (2006), pp. 1–4.

[Del01] Paul J Dellar. "Bulk and shear viscosities in lattice Boltzmann equations". In: *Pysical Review E* (2001).

[Del03] Paul J Dellar. "Incompressible limits of lattice Boltzmann equations using multiple relaxation times". In: *Journal of Computational Physics* 190 (2003), pp. 351–370.

[Dur+05] F. Durst et al. "The development lengths of laminar pipe and channel flows". In: *Journal of Fluids Engineering* 127.6 (2005), pp. 1154–1160.

[FH98] O. Filippova and D. Hänel. "Grid refinement for lattice-BGK models". In: *Journal of Computational Physics* 147 (1998), pp. 219–228.

[FST15] Yongliang Feng, Pierre Sagaut, and Wenquan Tao. "A Three Dimensional Lattice Model for Thermal Compressible Flow on Standard Lattices". In: *Journal of Computational Physics* (2015).

[GZS02] Zhaoli Guo, Chuguang Zheng, and Baochang Shi. “Discrete lattice effects on the forcing term in the lattice Boltzmann method”. In: *Physical Review E* 65 (2002), pp. 1–6.

[HDC02] Xiaoyi He, Gary D. Doolen, and T. Clark. “Comparison of the lattice Boltzmann method and the artificial compressibility method for Navier-Stokes equations”. In: *Journal of Computational Physics* 179.2 (2002), pp. 439–451.

[HL97] Xiaoyi He and Li-shi Luo. “Lattice Boltzmann Model for Incompressible Navier-Stokes Equation”. In: *Journal of Statistical Physics* 88 (1997).

[IGC13] I.G.Currie. *Fundamental Mechanics of Fluids*. 4th. CRC Press, 2013.

[JMS19] Jérôme Jacob, Orestis Malaspinas, and Pierre Sagaut. “A new hybrid recursive regularised bhatnagar–gross–krook collision model for lattice boltzmann method-based large eddy simulation”. In: *Journal of Turbulence* 19.11 (2019).

[Krü+17] Timm Krüger et al. *The lattice Boltzmann method*. 1st ed. Springer, Cham, 2017.

[Lat07] Jonas Latt. “Hydrodynamic limit of lattice Boltzmann equations”. PhD thesis. Université de Genève, 2007.

[LC06] Jonas Latt and Bastien Chopard. “Lattice Boltzmann method with regularized pre-collision distribution functions”. In: *Mathematics and Computers in Simulations* 72 (2006), pp. 165–168.

[Li+07] Q. Li et al. “Coupled double-distribution-function lattice Boltzmann method for the compressible Navier-Stokes equations”. In: *Physical Review E* 76.5 (Nov. 2007).

[Mal15] Orestis Malaspinas. “Increasing stability and accuracy of the lattice Boltzmann scheme: recursivity and regularization”. In: *arXiv preprint* (2015), pp. 1–31.

[Nas+14] Rupert W. Nash et al. “Choice of boundary condition for lattice-Boltzmann simulation of moderate-Reynolds-number flow in complex domains”. In: *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* 89.2 (2014), pp. 1–13.

[PS72] S. V. Patankar and D. B. Spalding. “A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows”. In: *International Journal of Heat and Mass Transfer* 15.10 (1972), pp. 1787–1806.

[Qia93] Y. H. Qian. “Simulating thermohydrodynamics with lattice BGK models”. In: *Journal of Scientific Computing* 8.3 (1993), pp. 231–242.

[SC96] James D Sterling and Shiyi Chen. "Stability Analysis of Lattice Boltzmann Methods". In: *Journal of Computational Physics* 123 (1996), pp. 196–206.

[Suc15] Sauro Succi. "Lattice Boltzmann 2038". In: *EPL (Europhysics Letters)* 109.5 (2015).

[Thü+06] Nils Thürey et al. "Optimization and stabilization of LBM free surface flow simulations using adaptive parameterization". In: *Computers and Fluids* 35.8-9 (2006), pp. 934–939.

[TRL14] Hatem Touil, Denis Ricot, and Emmanuel Lévêque. "Direct and large-eddy simulation of turbulent flows on composite multi-resolution grids by the lattice Boltzmann method". In: *Journal of Computational Physics* 256 (2014), pp. 220–233.

[ZH97] Qisu Zou and Xiaoyi He. "On pressure and velocity boundary conditions for the lattice Boltzmann BGK model". In: *Physics of Fluids* 9.6 (1997), pp. 1591–1596.

[Zou+95] Qisu Zou et al. "A improved incompressible lattice Boltzmann model for time-independent flows". In: *Journal of Statistical Physics* 81.1-2 (Oct. 1995), pp. 35–48.

A Continuity of the pressure force

The continuity of the pressure force per unit mass in the momentum equation expresses as

$$-\frac{c_0^2}{\rho}\nabla\rho' = -\frac{(\lambda c_0)^2}{\rho^*}\nabla\rho'^* \tag{63}$$

with $\rho = \rho_{\text{ref}} + \rho'$ and $\rho^* = \rho_{\text{ref}} + \rho'^*$ by definition. This equation may be rewritten as

$$\frac{1}{\rho^*}\nabla\rho^* = \frac{1}{\lambda^2}\frac{1}{\rho}\nabla\rho \tag{64}$$

which yields the general solution

$$\log(\rho^*) = \frac{1}{\lambda^2}\log(\rho) + C. \tag{65}$$

If $\rho' = \rho'^* = 0$ then $\rho = \rho^* = \rho_{\text{ref}}$. Therefore $C = \ln(\rho_{\text{ref}}) - (1/\lambda^2)\log(\rho_{\text{ref}})$, which eventually gives

$$\rho^* = \rho_{\text{ref}}\left(\frac{\rho}{\rho_{\text{ref}}}\right)^{\frac{1}{\lambda^2}}. \tag{66}$$

If one assumes that the Mach number is very small, ρ' and $\rho'^* \ll \rho_{\text{ref}}$. Therefore, $\rho'^* \approx (1/\lambda^2)\rho'$ as a first-order approximation , or equivalently

$$\frac{\rho'^*}{\rho'} \approx \left(\frac{\Delta t^*}{\Delta t}\right)^2. \tag{67}$$