



HAL
open science

Does the multiresolution lattice Boltzmann method allow to deal with waves passing through mesh jumps?

Thomas Bellotti, Loïc Gouarin, Benjamin Graille, Marc Massot

► To cite this version:

Thomas Bellotti, Loïc Gouarin, Benjamin Graille, Marc Massot. Does the multiresolution lattice Boltzmann method allow to deal with waves passing through mesh jumps?. *Comptes Rendus. Mathématique*, inPress. hal-03235133v1

HAL Id: hal-03235133

<https://hal.science/hal-03235133v1>

Submitted on 25 May 2021 (v1), last revised 18 Jan 2022 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Does the multiresolution lattice Boltzmann method allow to deal with waves passing through mesh jumps?

Thomas Bellotti^{Ⓢ a}, Loïc Gouarin^b, Benjamin Graille^{Ⓢ c} and Marc Massot^{Ⓢ d}

^a CMAP, CNRS, Ecole polytechnique, Institut Polytechnique de Paris, 91128 Palaiseau Cedex, France.

^b CMAP, CNRS, Ecole polytechnique, Institut Polytechnique de Paris, 91128 Palaiseau Cedex, France.

^c Université Paris-Saclay, CNRS, Laboratoire de mathématiques d'Orsay, 91405, Orsay, France.

^d CMAP, CNRS, Ecole polytechnique, Institut Polytechnique de Paris, 91128 Palaiseau Cedex, France.

E-mails: thomas.bellotti@polytechnique.edu (T. Bellotti),
loic.gouarin@polytechnique.edu (L. Gouarin),
benjamin.graille@universite-paris-saclay.fr (B. Graille),
marc.massot@polytechnique.edu (M. Massot)

Abstract. We consider an adaptive multiresolution-based lattice Boltzmann scheme, which we have recently introduced and studied from the perspective of the error control and the theory of the equivalent equations. This numerical strategy leads to high compression rates, error control and its high accuracy has been explained on uniform and dynamically adaptive grids. However, one key issue with non-uniform meshes within the framework of lattice Boltzmann schemes is to properly handle acoustic waves passing through a level jump of the grid. It usually yields spurious effects, in particular reflected waves. In this paper, we propose a simple mono-dimensional test-case for the linear wave equation with a fixed adapted mesh characterized by a potentially large level jump. We investigate this configuration with our original strategy and prove that we can handle and control the amplitude of the reflected wave, which is of fourth order in the space step of the finest mesh. Numerical illustrations show that the proposed strategy outperforms the existing methods in the literature and allow to assess the ability of the method to handle the mesh jump properly.

2020 Mathematics Subject Classification. 65M99, 65M50, 76M28.

Funding. The first author is supported by a PhD funding (year 2019) from the Ecole polytechnique.

This article is a draft (not yet accepted!)

1. Introduction

In [2, 3] we have proposed a novel approach to perform time-dynamic grid adaptation and to devise lattice Boltzmann schemes to be deployed on such adaptive meshes using multiresolution analysis. This framework guarantees reduced memory footprints for problems with steep solution and potential gains in terms of computational time. Moreover, the preeminent feature of this strategy is that it yields a precise bound on the additional numerical error caused by the compression of the solution on a dynamically adapted grid coupled with the numerical scheme, thanks to the information on the local smoothness of the solution provided by multiresolution. These peculiarities have been thoroughly investigated in these two works. The next question was to clarify – besides the previously elucidated error control – the potential alteration of the physics of the simulated system by the adaptive method, which has been fully studied in [1], concluding that our method of choice does not alter the reference lattice Boltzmann scheme up to local contributions of order four in the space step. This performance seems unprecedented in the literature. This analysis has been carried out on uniform or locally uniform grids for smooth solutions using the equivalent equations [4]. When non-smooth solutions are obtained, one must utilize the time-adaptive refinement to allow for the representation of singularities at the finest level of resolution.

The question which stimulated the present work has been raised by Pierre Lallemand and François Dubois during a seminary at the IHP in Paris and is the following: how does the scheme introduced in our previous works [1–3] behave once an acoustic wave is forced to pass through grids of different resolutions, like the one on Fig. 1? We emphasize that in the context of dynamic mesh adaptation, the only case in which waves could penetrate through a level jump is when they are locally very smooth. It is known in the lattice Boltzmann community that a wave passing through a grid transition normally splits into two parts: the first one correctly propagating through the interface and the second one which is reflected back. The phenomenological reason for this is the different acoustic impedance of two media made up of grids at different levels of refinement. Generally speaking, the aim is to devise numerical approaches which minimize the amplitude of the reflected waves.

Many strategies, both for the representation of an adaptive grid and for the construction of adaptive lattice Boltzmann methods have been explored. For a review on the different techniques and the issues linked with grid transitions, the reader can consult [7] and [6]. There is no widespread consensus on a standard test case to analyze the issue of reflected waves: we therefore consider a basic one-dimensional configuration which already introduces the main difficulties of more realistic multidimensional systems (*e.g.* the D2Q9 scheme) to be found in the available literature, arising from the applications.

2. Numerical method

Let us briefly sketch the structure of our numerical method in order to make the paper self-contained. The interested reader can consult [1–3] for more precision.

2.1. Space and time discretization

For the sake of presentation, we consider a one-dimensional setting on the bounded domain $\Omega = [0, 3]$. As in our previous work [1] and references therein, we take cells $C_{\ell,k} = [2^{-\ell}k, 2^{-\ell}(k+1)]$ for levels of resolution $\ell \in \{\ell_{\min}, \dots, \ell_{\max}\}$ and $k \in \{0, \dots, 3 \times 2^{\Delta\ell} - 1\}$, where $\Delta\ell = \ell_{\max} - \ell$. The cell center is denoted by $x_{\ell,k}$. Therefore, the space step of the grid at finest level of resolution is

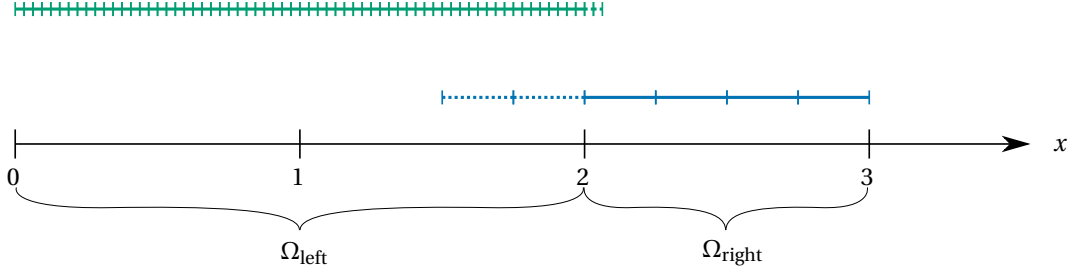


Figure 1. Example of domain $\Omega = \Omega_{\text{left}} \cup \Omega_{\text{right}}$ with $\Omega_{\text{left}} = [0, 2]$ finely meshed (green full line) and $\Omega_{\text{right}} = [2, 3]$ coarsely meshed (blue full line). Dashed lines represent ghost cells, where the solution needs to be updated to deploy the adaptive scheme.

$\Delta x = 2^{\ell_{\text{max}}}$. Cells with different levels ℓ can be employed to cover the domain Ω of interest as in Fig. 1, which also illustrates the experimental configuration used in the sequel.

The time discretization is uniform with step $\Delta t = \Delta x / \lambda$, where $\lambda > 0$ is the so-called “lattice velocity”. The time-step is the same for all the levels $\ell \in \{\ell_{\text{min}}, \dots, \ell_{\text{max}}\}$ and determined by ℓ_{max} .

2.2. Lattice Boltzmann method

The lattice Boltzmann method is an algorithmic “recipe” based on $q \in \mathbb{N}^*$ velocities $(e_\alpha)_{\alpha=0}^{\alpha=q-1} \subset \lambda\mathbb{Z}$ compatible with the lattice, with dimensionless counterparts $c_\alpha := e_\alpha / \lambda \in \mathbb{Z}$, $\alpha = 0, \dots, q-1$. The distribution function of the population moving with velocity e_α shall be denoted f^α . The algorithm is made up of two phases at each time step.

2.2.1. Collision phase

For every cell $C_{\ell,k}$ belonging to the mesh, the collision phase is performed locally and is a diagonal relaxation in the space of the moments

$$\begin{cases} \overline{m}_{\ell,k}(t) &= \mathbf{M} \overline{f}_{\ell,k}(t), \\ \overline{f}_{\ell,k}^*(t) &= \mathbf{M}^{-1} \left((\mathbf{I} - \mathbf{S}) \overline{m}_{\ell,k}(t) + \mathbf{S} \mathbf{m}^{\text{eq}}(\overline{m}_{\ell,k}^0(t), \dots, \overline{m}_{\ell,k}^{q_c-1}(t)) \right), \end{cases}$$

where $\mathbf{M} \in \text{GL}(q, \mathbb{R})$, $\mathbf{S} \in \mathcal{M}_q(\mathbb{R})$ diagonal with $\text{rank}(\mathbf{S}) = q - q_c$ where the non-zero entries belong to $]0, 2]$. Here, q_c is the number of conserved moments and the vector of moments at equilibrium \mathbf{m}^{eq} is a function of these conserved moments.

2.2.2. Transport phase

For every cell $C_{\ell,k}$ belonging to the mesh, the transport phase does not mix different populations. For every $\alpha \in \{0, \dots, q-1\}$ it comes under the form

$$\overline{f}_{\ell,k}^\alpha(t + \Delta t) = \overline{f}_{\ell,k}^{\alpha,*}(t) + \frac{1}{2^{\Delta\ell}} \sum_{m=-2}^2 C_{\Delta\ell,m}^\alpha \overline{f}_{\ell,k+m}^{\alpha,*}(t), \quad (1)$$

with weights recursively defined by

$$\begin{pmatrix} C_{\Delta\ell,-2}^\alpha \\ C_{\Delta\ell,-1}^\alpha \\ C_{\Delta\ell,0}^\alpha \\ C_{\Delta\ell,1}^\alpha \\ C_{\Delta\ell,2}^\alpha \end{pmatrix} = \begin{pmatrix} 0 & -1/8 & 0 & 0 & 0 \\ 2 & 9/8 & 0 & -1/8 & 0 \\ 0 & 9/8 & 2 & 9/8 & 0 \\ 0 & -1/8 & 0 & 9/8 & 2 \\ 0 & 0 & 0 & -1/8 & 0 \end{pmatrix} \begin{pmatrix} C_{\Delta\ell-1,-2}^\alpha \\ C_{\Delta\ell-1,-1}^\alpha \\ C_{\Delta\ell-1,0}^\alpha \\ C_{\Delta\ell-1,1}^\alpha \\ C_{\Delta\ell-1,2}^\alpha \end{pmatrix}, \quad \text{and} \quad \begin{cases} C_{0,-c_\alpha}^\alpha &= 1, \\ C_{0,0}^\alpha &= -1, \\ C_{0,m}^\alpha &= 0, \quad m \notin \{0, -c_\alpha\}. \end{cases}$$

The transport phase (1) needs information stored on ghost cells (dashed in Fig. 1), which thus must be updated. This is done by averaging for the ghost cells underneath more refined cells followed by the use of interpolations (from which (1) is recovered, see [1] for the details) for ghost cells above coarser cells.

3. Target problem, results and discussion

We introduce a simple system which sustains travelling acoustic waves which are eventually sent against a level jump in the computational grid.

3.1. Equations and numerical scheme

The target equation is the linear wave equation with velocity $c > 0$ (taken $c = 1/2$ in the experiments) over the whole real line \mathbb{R}

$$\begin{cases} \partial_{tt}u - c^2\partial_{xx}u = 0, \\ u(t=0, x) = u_0(x), \\ \partial_t u(t=0, x) = 0, \end{cases} \Leftrightarrow \begin{cases} \partial_t u + \partial_x v = 0, \\ \partial_t v + c^2\partial_x u = 0, \\ u(t=0, x) = u_0(x), \\ v(t=0, x) = 0, \end{cases}$$

which has been recast under the form of first order system for simulating it. For this test we simulate on the bounded domain $\Omega = \Omega_{\text{left}} \cup \Omega_{\text{right}}$ with $\Omega_{\text{left}} = [0, 2]$ and $\Omega_{\text{right}} = [2, 3]$ with the initial datum $u_0(x) = \exp(-100(x - 3/2)^2)$. The most barebone lattice Boltzmann scheme to handle such equation – yet yielding the difficulties of more sophisticated ones – is a three velocity scheme $q = 3$ with two conserved moments ($u = m^0$ and $v = m^1$)

$$c_0 = 0, \quad c_1 = 1, \quad c_2 = -1, \quad \mathbf{M} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & \lambda & -\lambda \\ 0 & \lambda^2/2 & \lambda^2/2 \end{pmatrix}, \quad \mathbf{S} = \text{diag}(0, 0, p), \quad m^{2,\text{eq}} = \frac{c^2}{2} u.$$

We consider $\lambda = 1$, $p = 1.7$ and a final time $T = 1.5625$ for each simulation.

3.2. Results

To quantify the amplitude of the reflected wave, we simulate on different configurations:

- \bar{u}^{jump} is the solution obtained using the spatial discretization illustrated on Fig. 1, the left sub-domain Ω_{left} is finely meshed with level $\ell_{\text{left}} = \ell_{\text{max}}$, whereas the right sub-domain Ω_{right} is coarsely meshed with level $\ell_{\text{right}} = \ell_{\text{min}} \leq \ell_{\text{left}}$. We vary the jump gap $\ell_{\text{jump}} := |\ell_{\text{left}} - \ell_{\text{right}}|$. This configuration is the monodimensional equivalent of that presented by [7].
- \bar{u}^{ref} is the reference solution obtained on the uniform mesh at the finest level ℓ_{max} .
- \bar{u}^{coarse} is the solution obtained on the uniform mesh at the coarsest level ℓ_{min} .

We introduce the L^1 -norm of the exact solution u for the sake of normalizing: $\|u(t, \cdot)\|_{1, \Delta x} = \sum_k \Delta x |u(t, x_{\ell_{\text{max}}, k})| \simeq \|u_0\|_{1, \Delta x}$. We measure

$$\begin{aligned} E_{\text{ref}}(t) &= \frac{\|\bar{u}^{\text{ref}}(t, \cdot) - u(t, \cdot)\|_{1, \Delta x}}{\|u(t, \cdot)\|_{1, \Delta x}}, \\ E_{\text{coarse}}(t) &= \frac{\|\widehat{\bar{u}}^{\text{coarse}}(t) - u(t, \cdot)\|_{1, \Delta x}}{\|u(t, \cdot)\|_{1, \Delta x}}, \quad D_{\text{coarse}}(t) = \frac{\|\widehat{\bar{u}}^{\text{coarse}}(t) - \bar{u}^{\text{ref}}(t)\|_{1, \Delta x}}{\|u(t, \cdot)\|_{1, \Delta x}}, \end{aligned}$$

$$E_{\text{jump}}(t) = \frac{\|\widehat{\widehat{u}}^{\text{jump}}(t) - u(t, \cdot)\|_{1, \Delta x}}{\|u(t, \cdot)\|_{1, \Delta x}}, \quad D_{\text{jump}}(t) = \frac{\|\widehat{\widehat{u}}^{\text{jump}}(t) - \overline{u}^{\text{ref}}(t)\|_{1, \Delta x}}{\|u(t, \cdot)\|_{1, \Delta x}},$$

$$D_{\text{jump-refl}}(t) = \frac{\|(\overline{u})^{\text{jump}}(t) - \overline{u}^{\text{ref}}(t)\|_{\chi\{k : C_{\ell_{\text{left}}, k} \subset \Omega_{\text{left}}\}}\|_{1, \Delta x}}{\|u(t, \cdot)\|_{1, \Delta x}}.$$

Here the double hat operator $\widehat{\widehat{\cdot}}$ represents the application of the same interpolation used to update the ghost cells. This step is needed to compare solution on the same mesh. Using the analysis of [1], we can prove an estimate on the amplitude of the reflected wave $D_{\text{jump-refl}}(T)$:

Proposition 1. *Assume the solution of the the problem with level jump in the computational mesh to be smooth, namely of class \mathcal{C}^4 , for any time $t \in [0, T]$, then we have*

$$D_{\text{jump-refl}}(T) = \mathcal{O}(\Delta x^4).$$

Proof. Let us introduce the following indices for the cells close to the level jump

$$\begin{aligned} \overleftarrow{k} &= \{k \text{ such that } C_{\ell_{\text{left}}, k} \subset \Omega_{\text{left}} \text{ and } C_{\ell_{\text{left}}, k} \cap \partial\Omega_{\text{left}} \neq \emptyset\}, \\ \overrightarrow{k} &= \{k \text{ such that } C_{\ell_{\text{left}}, k} \subset \Omega_{\text{right}} \text{ and } C_{\ell_{\text{left}}, k} \cap \partial\Omega_{\text{right}} \neq \emptyset\}, \\ \underline{k} &= \{k \text{ such that } C_{\ell_{\text{right}}, k} \subset \Omega_{\text{right}} \text{ and } C_{\ell_{\text{right}}, k} \cap \partial\Omega_{\text{right}} \neq \emptyset\}, \end{aligned}$$

then performing the Taylor expansions detailed in [1], we claim that for every time t

$$\overline{f}_{\ell_{\text{right}}, \underline{k}}^{\text{jump}, \alpha}(t) = \overline{f}_{\ell_{\text{left}}, \overleftarrow{k}}^{\text{ref}, \alpha}(t) + \mathcal{O}(\Delta x^4).$$

The collision phase is linear, otherwise we may use a local Lipschitz property of the equilibria.

$$\overline{f}_{\ell_{\text{right}}, \underline{k}}^{\text{jump}, \alpha, \star}(t) = \overline{f}_{\ell_{\text{left}}, \overleftarrow{k}}^{\text{ref}, \alpha, \star}(t) + \mathcal{O}(\Delta x^4).$$

The update of the ghost cell $C_{\ell_{\text{left}}, \overrightarrow{k}}$ is a linear combination of values between which we have those on $C_{\ell_{\text{right}}, \underline{k}}$, therefore

$$\overline{f}_{\ell_{\text{left}}, \overrightarrow{k}}^{\text{jump}, \alpha, \star}(t) = \overline{f}_{\ell_{\text{left}}, \overrightarrow{k}}^{\text{ref}, \alpha, \star}(t) + \mathcal{O}(\Delta x^4).$$

Concentrating on the stream phase of $c_2 = -1$, we have

$$\begin{cases} \overline{f}_{\ell_{\text{left}}, \overleftarrow{k}}^{\text{jump}, 2}(t + \Delta t) = \overline{f}_{\ell_{\text{left}}, \overleftarrow{k}}^{\text{jump}, 2, \star}(t) = \overline{f}_{\ell_{\text{left}}, \overleftarrow{k}}^{\text{ref}, 2, \star}(t) + \mathcal{O}(\Delta x^4), \\ \overline{f}_{\ell_{\text{left}}, \overleftarrow{k}}^{\text{ref}, 2}(t + \Delta t) = \overline{f}_{\ell_{\text{left}}, \overleftarrow{k}}^{\text{ref}, 2, \star}(t). \end{cases}$$

Thus by computing the moment u and taking the difference, we deduce that for every time

$$|\overline{u}_{\ell_{\text{left}}, \overleftarrow{k}}^{\text{jump}}(t + \Delta t) - \overline{u}_{\ell_{\text{left}}, \overleftarrow{k}}^{\text{ref}}(t + \Delta t)| = \mathcal{O}(\Delta x^4).$$

The CFL condition on the finest resolution imposes that information, thus the errors, propagate of one cell for one time step. The constant $C > 0$ carries the normalization

$$\begin{aligned} D_{\text{jump-refl}}(T) &= C \left(\sum_{\substack{k \neq \overleftarrow{k} \text{ s.t.} \\ C_{\ell_{\text{max}}, k} \subset \Omega_{\text{left}}}} \Delta x |\overline{u}_{\ell_{\text{max}}, k}^{\text{jump}}(T) - \overline{u}_{\ell_{\text{max}}, k}^{\text{ref}}(T)| + \Delta x |\overline{u}_{\ell_{\text{max}}, \overleftarrow{k}}^{\text{jump}}(T) - \overline{u}_{\ell_{\text{max}}, \overleftarrow{k}}^{\text{ref}}(T)| \right), \\ &\leq C \left(\frac{1}{C} D_{\text{jump-refl}}(T - \Delta t) + \mathcal{O}(\Delta x^5) \right) \leq \dots \leq \frac{T}{\Delta t} \mathcal{O}(\Delta x^5) = \frac{\lambda T}{\Delta x} \mathcal{O}(\Delta x^5), \\ &= \mathcal{O}(\Delta x^4). \end{aligned}$$

□

The results of the numerical simulations are provided in Table 1. We have also run comparisons with the method from Fakhari and Lee [5] and that of Rohde *et al.* [8] presented in Fig. 2.

Table 1. Results for the transition between fine and coarse mesh with the Gaussian as initial datum. Numerical convergence rates are reported between parenthesis.

ℓ_{\max}	$E_{\text{ref}}(T)$	$E_{\text{coarse}}(T)$	$D_{\text{coarse}}(T)$	$E_{\text{jump}}(T)$	$D_{\text{jump}}(T)$	$D_{\text{jump-ref}}(T)$
$\ell_{\text{jump}} = 1$						
7	7.30E-02	8.19E-02	1.30E-02	7.49E-02	2.64E-03	1.27E-05
8	3.78E-02 (0.95)	3.89E-02 (1.07)	1.86E-03 (2.81)	3.80E-02 (0.98)	3.84E-04 (2.78)	8.90E-07 (3.83)
9	1.92E-02 (0.98)	1.93E-02 (1.01)	2.28E-04 (3.03)	1.93E-02 (0.98)	5.19E-05 (2.89)	5.95E-08 (3.90)
10	9.70E-03 (0.99)	9.70E-03 (0.99)	2.49E-05 (3.20)	9.71E-03 (0.99)	6.75E-06 (2.94)	3.86E-09 (3.95)
11	4.87E-03 (0.99)	4.87E-03 (0.99)	3.67E-06 (2.76)	4.87E-03 (0.99)	8.65E-07 (2.96)	2.46E-10 (3.97)
12	2.44E-03 (1.00)	2.44E-03 (1.00)	1.08E-06 (1.77)	2.44E-03 (1.00)	1.11E-07 (2.96)	1.55E-11 (3.99)
13	1.22E-03 (1.00)	1.22E-03 (1.00)	3.11E-07 (1.79)	1.22E-03 (1.00)	1.44E-08 (2.95)	9.76E-13 (3.99)
$\ell_{\text{jump}} = 2$						
7	7.30E-02	1.61E-01	9.47E-02	9.22E-02	2.21E-02	4.84E-04
8	3.78E-02 (0.95)	5.00E-02 (1.68)	1.68E-02 (2.50)	4.04E-02 (1.19)	3.50E-03 (2.66)	3.10E-05 (3.96)
9	1.92E-02 (0.97)	2.06E-02 (1.28)	2.24E-03 (2.90)	1.96E-02 (1.05)	4.71E-04 (2.89)	2.03E-06 (3.94)
10	9.70E-03 (0.99)	9.82E-03 (1.07)	2.63E-04 (3.09)	9.74E-03 (1.01)	6.07E-05 (2.96)	1.31E-07 (3.96)
11	4.87E-03 (0.99)	4.87E-03 (1.01)	2.78E-05 (3.24)	4.88E-03 (1.00)	7.70E-06 (2.98)	8.32E-09 (3.97)
12	2.44E-03 (1.00)	2.44E-03 (1.00)	4.67E-06 (2.57)	2.44E-03 (1.00)	9.70E-07 (2.99)	5.25E-10 (3.99)
13	1.22E-03 (1.00)	1.22E-03 (1.00)	1.39E-06 (1.75)	1.22E-03 (1.00)	1.22E-07 (2.99)	3.30E-11 (3.99)
$\ell_{\text{jump}} = 3$						
7	7.30E-02	4.33E-01	3.62E-01	1.83E-01	1.12E-01	9.81E-03
8	3.78E-02 (0.95)	1.25E-01 (1.79)	9.31E-02 (1.96)	5.79E-02 (1.66)	2.26E-02 (2.31)	7.10E-04 (3.79)
9	1.92E-02 (0.97)	3.02E-02 (2.05)	1.45E-02 (2.68)	2.16E-02 (1.42)	3.12E-03 (2.86)	4.13E-05 (4.10)
10	9.70E-03 (0.99)	1.07E-02 (1.49)	1.80E-03 (3.01)	9.98E-03 (1.12)	3.97E-04 (2.98)	2.64E-06 (3.97)
11	4.87E-03 (0.99)	4.94E-03 (1.12)	1.99E-04 (3.18)	4.91E-03 (1.02)	4.96E-05 (3.00)	1.68E-07 (3.97)
12	2.44E-03 (1.00)	2.44E-03 (1.02)	2.13E-05 (3.23)	2.45E-03 (1.00)	6.19E-06 (3.00)	1.06E-08 (3.99)
13	1.22E-03 (1.00)	1.22E-03 (1.00)	5.23E-06 (2.03)	1.22E-03 (1.00)	7.74E-07 (3.00)	6.66E-10 (3.99)
$\ell_{\text{jump}} = 4$						
7	7.30E-02	9.10E-01	8.43E-01	4.00E-01	3.29E-01	6.02E-02
8	3.78E-02 (0.95)	4.24E-01 (1.10)	3.88E-01 (1.12)	1.57E-01 (1.35)	1.20E-01 (1.45)	1.43E-02 (2.07)
9	1.92E-02 (0.97)	9.69E-02 (2.13)	8.24E-02 (2.24)	3.74E-02 (2.07)	2.02E-02 (2.58)	8.21E-04 (4.13)
10	9.70E-03 (0.99)	1.78E-02 (2.44)	1.08E-02 (2.93)	1.16E-02 (1.69)	2.42E-03 (3.06)	4.42E-05 (4.22)
11	4.87E-03 (0.99)	5.47E-03 (1.71)	1.21E-03 (3.15)	5.08E-03 (1.20)	2.89E-04 (3.07)	2.85E-06 (3.95)
12	2.44E-03 (1.00)	2.45E-03 (1.16)	1.25E-04 (3.27)	2.47E-03 (1.04)	3.51E-05 (3.04)	1.83E-07 (3.97)
13	1.22E-03 (1.00)	1.21E-03 (1.02)	1.98E-05 (2.66)	1.23E-03 (1.01)	4.34E-06 (3.02)	1.15E-08 (3.99)
$\ell_{\text{jump}} = 5$						
7	7.30E-02	1.24E+00	1.20E+00	6.46E-01	5.82E-01	1.22E-01
8	3.78E-02 (0.95)	9.30E-01 (0.41)	8.96E-01 (0.42)	3.88E-01 (0.74)	3.50E-01 (0.73)	7.35E-02 (0.73)
9	1.92E-02 (0.97)	4.51E-01 (1.04)	4.34E-01 (1.05)	1.45E-01 (1.42)	1.27E-01 (1.47)	1.86E-02 (1.98)
10	9.70E-03 (0.99)	8.05E-02 (2.49)	7.44E-02 (2.54)	2.58E-02 (2.49)	1.78E-02 (2.83)	8.87E-04 (4.39)
11	4.87E-03 (0.99)	1.06E-02 (2.93)	7.74E-03 (3.27)	6.32E-03 (2.03)	1.80E-03 (3.31)	4.31E-05 (4.36)
12	2.44E-03 (1.00)	2.68E-03 (1.98)	7.52E-04 (3.36)	2.58E-03 (1.29)	2.00E-04 (3.18)	2.84E-06 (3.93)
13	1.22E-03 (1.00)	1.18E-03 (1.18)	8.06E-05 (3.22)	1.24E-03 (1.06)	2.35E-05 (3.08)	1.86E-07 (3.93)

3.3. Discussion

Commenting on Table 1, we see that the reference scheme converges linearly $E_{\text{ref}}(T) = \mathcal{O}(\Delta x)$ once refining as expected from the analysis by the equivalent equations [4]. The error $E_{\text{coarse}}(T) \leq E_{\text{ref}}(T) + D_{\text{coarse}}(T) = \mathcal{O}(\Delta x) + \mathcal{O}(\Delta x^3) = \mathcal{O}(\Delta x)$ converges linearly as well because the additional difference $D_{\text{coarse}}(T) = \mathcal{O}(\Delta x^3)$ does not influence the overall convergence, as pointed out in [1]. The same behavior is observed for the mesh with jump, namely for $E_{\text{jump}}(T)$ and $D_{\text{jump}}(T)$.

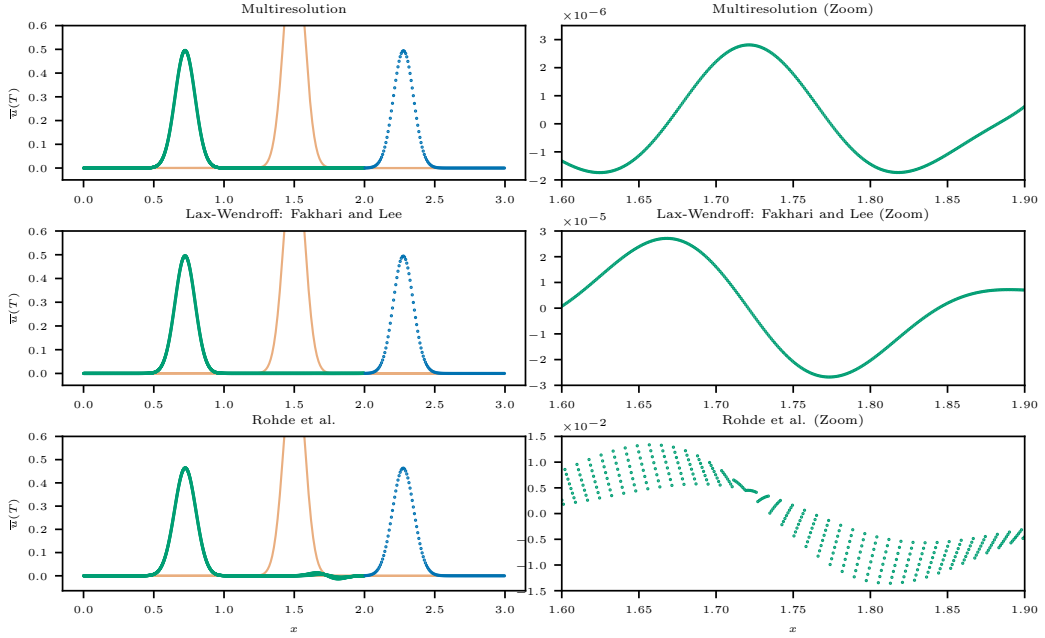


Figure 2. Results of the simulation on the mesh with jump (whole domain on the left, magnification on $[1.6, 1.9]$ and on the y axis on the right). Initial solution in pale orange and solution at $t = T$ in green (left subdomain) and blue (right subdomain). On the first row, we use our multiresolution scheme [1–3]. On the second row, we use the Lax-Wendroff scheme by Fakhari and Lee [5]. On the third row, the scheme with local time-stepping by Rohde *et al.* [8]. The simulation uses $\ell_{\text{left}} = 10$ and $\ell_{\text{jump}} = 3$.

Very interestingly $D_{\text{jump}}(T) \leq D_{\text{coarse}}(T)$: counterintuitively this is *a priori* not granted due to the possible formation of waves reflected at the jump, even though only a part of the domain is coarsened. This gives a first indication about the fact that the reflected waves are perfectly mastered. The second indication comes from $D_{\text{jump-refl}}(T) = \mathcal{O}(\Delta x^4)$. This means that with our method, we are able to decrease the amplitude of the reflected waves with fourth-order convergence in the space step, in accordance with Prop. 1. The supraconvergence compared to $D_{\text{jump}}(T)$ comes from the fact that at each time step, the reflected wave is generated only on the cell of Ω_{right} next to the interface, so that it eventually propagates to the left inside the fine medium without additional amplification of the error.

Compared to other methods, we can show with the same proof path than Prop. 1 that the Lax-Wendroff strategy by [5] yields $D_{\text{jump-refl}}(T) = \mathcal{O}(\Delta x^3)$, which is one order less than our method. This can also be qualitatively seen on Fig. 2. Concerning the approach by [8] where local time-stepping is used, we observe that it yields quite large reflected waves with a staggered structure (with $2^{\ell_{\text{jump}}}$ aligned points for each staggering section), as observed by the authors of this work. However, the local time-stepping prevents us from applying the same theoretical study to this scheme.

Mastering reflected waves at a high order of accuracy is important when our technique is extended to typical multidimensional applications. When simulating the incompressible Navier-Stokes equations *via* a quasi-incompressible D2Q9 scheme, spurious acoustic waves are of order $\mathcal{O}(\Delta x^2)$, thus controlling their reflection at order $\mathcal{O}(\Delta x^4)$ is a highly desirable feature of the

scheme.

4. Conclusions

In this contribution, we have briefly presented our adaptive lattice Boltzmann method, which is studied in Prop. 1 using the technique introduced in [1] to conclude that in case of a fixed mesh jump, the amplitude of the spuriously reflected waves is of order $\mathcal{O}(\Delta x^4)$. This fact is numerically verified and compared to the performance of other approaches available in the literature [5] and [8], showing that our method outperforms these traditional approaches.

It is worthwhile observing that the original method [2, 3] was conceived to be used with dynamically adapted meshes which automatically follow waves and fronts with finer discretizations once their lack of regularity justifies the depart from a coarse uniform mesh. Thus, in this case, we even do not expect the $\mathcal{O}(\Delta x^4)$ perturbation because fronts never cross level jumps but are precisely and successfully “chased” by the fine discretization.

For the sake of a quick and effective presentation, we have restrained the study to the one-dimensional setting. However, the generalization to higher spatial dimensions is straightforward and follows the indications of [1], since the operators involved in the multiresolution analysis are extended [2] by tensor product in the remaining directions of the space. We have proved in [1] that the multidimensional extension of the method retains the same accuracy levels in every spatial direction.

Acknowledgments

The authors thank Pierre Lallemand and François Dubois for the stimulating discussions during the “groupe de travail Schémas de Boltzmann sur réseau” at the IHP Paris.

References

- [1] BELLOTTI, T., GOUARIN, L., GRAILLE, B., AND MASSOT, M. High accuracy analysis of adaptive multiresolution-based lattice boltzmann schemes via the equivalent equations. *SMAI Journal of Computational Mathematics* (2021). Submitted - available on HAL : <https://hal.archives-ouvertes.fr/hal-03234120>.
- [2] BELLOTTI, T., GOUARIN, L., GRAILLE, B., AND MASSOT, M. Multidimensional fully adaptive lattice boltzmann methods with error control based on multiresolution analysis. *Submitted to the Journal of Computational Physics, arXiv preprint arXiv:2103.02903* (2021).
- [3] BELLOTTI, T., GOUARIN, L., GRAILLE, B., AND MASSOT, M. Multiresolution-based mesh adaptation and error control for lattice boltzmann methods with applications to hyperbolic conservation laws. *Submitted to SIAM SISC, arXiv preprint arXiv:2102.12163* (2021).
- [4] DUBOIS, F. Equivalent partial differential equations of a lattice boltzmann scheme. *Computers & Mathematics with Applications* 55, 7 (2008), 1441–1449.
- [5] FAKHARI, A., AND LEE, T. Finite-difference lattice boltzmann method with a block-structured adaptive-mesh-refinement technique. *Physical Review E* 89, 3 (2014), 033310.
- [6] HORSTMANN, J. *Hybrid numerical method based on the lattice Boltzmann approach with application to non-uniform grids*. PhD thesis, Université de Lyon, 2018.
- [7] LAGRAVA SANDOVAL, D. W. *Revisiting grid refinement algorithms for the lattice Boltzmann method*. PhD thesis, University of Geneva, 2012.
- [8] ROHDE, M., KANDHAI, D., DERKSEN, J., AND VAN DEN AKKER, H. E. A generic, mass conservative local grid refinement technique for lattice-boltzmann schemes. *International journal for numerical methods in fluids* 51, 4 (2006), 439–468.