We propose a novel approach to the numerical simulation of thin film flows, based on the visco-capillary shallow water equations, integrated by means of a lattice Boltzmann method. We outline the basic features of the method, show how the correct hydrodynamic equations are recovered and perform validation tests. The numerical scheme is applied to the viscous Rayleigh-Taylor instability of a thin film and to the spreading of a sessile drop towards its equilibrium contact angle configuration. We show that the Cox-Voinov law is satisfied, and that the effect of a tunable slip length on the substrate is correctly captured. We address, then, the problem of a droplet sliding on an inclined plane, finding that the Capillary number scales linearly with the Bond number, in agreement with experimental results. At last, we demonstrate the ability of the method to handle heterogenous and complex systems by showcasing the controlled dewetting of a thin film on a chemically structured substrate.

I. INTRODUCTION

Thin layers of liquids on solid surfaces are prevalently encountered in a host of natural and technological settings [1, 2]. Therefore, understanding and controlling their stability and dynamics is a central problem for fundamental physics, as well as for applied research in process engineering and nanotechnology [3, 4]. Coating processes, for instance, rely crucially on the mutual affinity of liquid and surface (i.e. on wettability properties). When the liquid film is sufficiently thin, in fact, it can become unstable, leading to the dewetting of the coated area [5]. From the modelling point of view, the challenge consists in the fact that the physics of thin films is intrinsically multiscale, for it involves phenomena ranging from the molecular scale at the three phase contact line, to the micro-/nano-metric size of the film thickness, to the size of the film as a whole, extending over the coated substrate area.

A fully resolved bottom-up atomistic approach would be, obviously, unfeasible, if hydrodynamic regimes are to be explored. It clearly appears that some degree of model order reduction is required. Most hydrodynamic models of thin liquid films, in the framework of the lubrication theory, simplify the complexity of the full 3D Navier-Stokes equations to one scalar transport equation (the lubrication equation) for the film thickness field $h(x,t)$ [3, 6, 7]:

$$\partial_t h = \nabla \cdot (Q(h)\nabla p_{film}).$$

(1)

Here, $Q(h)$ is the mobility function, whose explicit form depends on the boundary condition for the velocity at the surface (For a no-slip boundary, $Q(h) = h^3/(3\eta)$, with $\eta$ being the viscosity), and $p_{film}$ is the film pressure at the free liquid surface. Stable and reliable numerical solutions of Eq. (1) require complex numerical methods, whose simulation is often computationally expensive [8]. Moreover, an ever-growing number of microfluidic problems requires to cope with complex fluids rather than simple liquids, i.e. fluids with non-trivial internal microstructure and/or complex non-Newtonian rheological behaviour (e.g. colloidal suspensions, polymer solutions, etc.). It is not straightforward to generalize equation (1) to such situations, deriving it rigorously from the full hydrodynamic equations of motion. The quest for an efficient multiscale numerical method for simulating thin film hydrodynamics, versatile for the inclusion of multiphysics features, is, thus, an ongoing endeavour.

In this paper, we present a novel approach to the numerical study of thin liquid films, based on a lattice Boltzmann (LB) solver for the shallow water equation. Due to the built-in properties of LB models, our method enjoys an outstanding computational performance, especially on parallel architectures and graphics processing units.

The paper is organized as follows. We first discuss the equations of motion and their derivation from the full Navier-Stokes equations (section II). Then, we introduce...
the numerical method for their simulation (section III). In section IV we present validation results including the
Rayleigh-Taylor instability of thin fluid films, the spreading of a sessile droplet on a substrate and the sliding of a
droplet on an inclined plane. After showcasing the ability of our method to handle large and heterogeneous sub-
strates, we present some computational aspects including the performance of our implementation for Graphics Pro-
cessing Units (GPUs).

II. THE SHALLOW WATER EQUATIONS FOR THIN LIQUID FILMS

When a layer of fluid is characterized by a vertical length scale $H$ much smaller than the longitudinal one $L$, the equations of motion can be simplified by depth-
averaging under the approximation that the length scales ratio, $\varepsilon \equiv H/L$, is small ($\varepsilon \ll 1$, see Fig. 1).

The resulting system is known as Shallow Water (or Saint Venant) Equations (SWE) [9]. The SWE are widely
employed for geophysical applications, from oceanography to atmospheric science, i.e., in flow regimes at very
high Reynolds number. Therefore, classical derivations of the SWE typically start from the Euler equations. We
are interested, instead, in the dynamics of liquid films in the micrometric or sub-micrometric scale, where viscous
effects with the substrate. Assuming a parabolic profile to the vertical coordinate; it embeds, therefore, friction
cous term containing the second derivative with respect of the vertical coordinate; it embeds, therefore, friction
cous term containing the second derivative with respect to the vertical coordinate; it embeds, therefore, friction
cous term containing the second derivative with respect to the vertical coordinate; it embeds, therefore, friction
cous term containing the second derivative with respect to the vertical coordinate; it embeds, therefore, friction
cous term containing the second derivative with respect to the vertical coordinate; it embeds, therefore, friction
III. NUMERICAL MODEL

Numerical simulations of the lubrication equation (1), be they performed by Finite Differences of Finite Element methods, suffer from important technical difficulties stemming from the high order and strongly nonlinear character of the partial differential equation. In particular, any implementation has to assure that the positivity of the mobility \( Q(h) \) (and \( h \)) is preserved [17]. As a consequence, numerical studies of thin film hydrodynamics within the framework of lubrication theory are, computationally, demanding [18]. Moreover, in the lubrication approach there is only one dynamical variable (the film thickness \( h \)), whereas the velocity is effectively slaved to gradients of the pressure and therefore limited to scenarios of vanishing Reynolds numbers. However, these cases might emerge in certain technological applications involving thin films, such as in spin-casting deposition [3, 19, 20] which violate this assumption. The SWE system explicitly includes inertial terms in the equation for the two dimensional depth-averaged velocity field. Equations like (6) are amenable of numerical integration by means of a lattice Boltzmann method (LBM) [21–24]. This possibility brings in the added values of excellent scalability of the corresponding algorithm on parallel architectures and, as we shall see in the next sections, on GPUs. Studying thin film hydrodynamics with the SWE, as compared to the fully resolved (three-dimensional) case, allows to employ a two-dimensional lattice Boltzmann algorithm to simulate 3D problems, thus heavily reducing the computational effort.

In standard LBM approaches, one considers the lattice dynamics of discrete distribution functions \( f_l(x, t) \), representing the probability density to find particles at site \( x \) at time \( t \) carrying a velocity \( c^{(l)} \). In the case of SWE this concept is generalized to that of a film thickness distribution function, as it is done, for instance, in thermal models for the energy density distribution function [25]. The lattice Boltzmann equation for the \( f_l \) reads [26]

\[
\frac{df_l(x, t + \Delta t)}{dt} = (1 - \omega) f_l(x, t) + \omega f^{(eq)}_l(x, t) + \frac{\Delta t}{c_s^2} c^{(l)} \cdot \mathbf{F},
\]

(10)

Algorithmically, this equation can be seen as made up of two steps. A local collision step where the \( f_l(x, t) \) “relax” towards the local equilibrium distributions \( f^{(eq)}_l(x, t) \) with rate \( \omega = \Delta t / \tau \) (where \( \tau \), the relaxation time, is proportional to the viscosity \( \eta \)): the distribution functions are substituted by their weighted average (with weights \( \omega \) and \( 1 - \omega \)) with the equilibria. A non-local streaming step where the updated distribution functions are scattered to the nearest neighbouring sites. The last term on the rhs of (10) includes the total force \( \mathbf{F} \) which can be in general the sum of bulk and internal (possibly space and time dependent) contributions. The parameters \( c_s \) (the lattice speed of sound) and \( w_l \) (the so called “weights”) depend on the geometry of the lattice and are determined under the requirement of recovering isotropy up to a certain order of tensors of the lattice velocities [27].

We choose a two-dimensional lattice with lattice constant \( \Delta x \) and nine-velocities. For simplicity, we keeps \( \Delta x = \Delta y = \tau = 1 \) throughout this paper and follow the standard notation, where \( c_s = 1 / \sqrt{3} \) and the \( c^{(l)} = (c^{(l)}_x, c^{(l)}_y) \), \( l = 0, 1, \ldots, 8 \), are [28, 29]

\[
f^{(l)} = \begin{cases} (0, 0) & l = 0 \\
\frac{\cos((l-1)\pi/4), \sin((l-1)\pi/4)}{\sqrt{2}} & l = 1, 3, 5, 7 \\
\frac{\cos((l-1)\pi/4), \sin((l-1)\pi/4)}{\sqrt{2}} & l = 2, 4, 6, 8
\end{cases}
\]

(11)

with the corresponding weights \( w_l \)

\[
w_l = \begin{cases} \frac{1}{36} & l = 0 \\
\frac{1}{9} & l = 1, 3, 5, 7 \\
\frac{1}{2} & l = 2, 4, 6, 8
\end{cases}
\]

(12)

The equilibrium distribution functions \( f^{(eq)}_l \) have to be determined to recover the desired SWE (6) in the hydrodynamic limit. They have, therefore, to fulfill the following relations with the hydrodynamic fields of the liquid height

\[
h = \sum_{l=0}^{8} f^{(eq)}_l,
\]

(13)

momentum

\[
h u_i = \sum_{l=0}^{8} c^{(l)}_i f^{(eq)}_l
\]

(14)

and momentum flux tensor

\[
\frac{1}{2} h^2 g \delta_{ij} + h u_i u_j = \sum_{l=0}^{8} c^{(l)}_i c^{(l)}_j f^{(eq)}_l
\]

(15)

With the usual ansatz for the \( f^{(eq)}_l \) based on the polynomial expansion of the Maxwellian equilibrium up to second order in the Mach number \( Ma = u/c_s \), the equilibrium distribution functions read

\[
f^{(eq)}_l = \begin{cases} h - \frac{5}{6} h g^2 + \frac{2}{3} h^2 \frac{u^2}{c_s^2} & l = 0 \\
\frac{1}{6} \left( h + \frac{c^{(l)}_i u}{c^{(l)}_j} \right) + \frac{h (c^{(l)}_i u)^2}{2 c_s^2} - \frac{h u^2}{6 c_s^2} & l = 1, 3, 5, 7 \\
\frac{1}{12} h + \frac{h c^{(l)}_i u}{6 c_s^2} + \frac{h (c^{(l)}_i u)^2}{8 c_s^2} - \frac{h u^2}{24 c_s^2} \quad & l = 2, 4, 6, 8
\end{cases}
\]

(16)

where \( u^2 = |u|^2 \) is the magnitude of the velocity. The multiscale Chapman-Enskog expansion of such an LBM recovers the correct transport equation for the height field [22, 23, 30]

\[
\partial_t h + \nabla \cdot (hu) = 0,
\]

(17)

whereas the velocity equation is given by (for small \( Ma \) and inverse Froude number \( Fr = c_s / \sqrt{gH} \), i.e. negligible gravitational effects as [22])

\[
\partial_t (hu) + \nabla \cdot (hu \otimes u) = -gh \nabla h + \eta \nabla^2 (hu) + \mathbf{F}.
\]

(18)
Eq. (18) differs from the one appearing in (6). The missing terms, namely the viscous friction term $-\eta \alpha_3(h) \mathbf{u}$, the film pressure $-h \nabla p_{\text{film}}$ and the bed terms, are, then, added through the forcing $\mathbf{F}(x, t)$. It can be noticed that the longitudinal viscous term in (18), $\eta \nabla^2(h \mathbf{u})$, still differs from the correct one as in (6). The due counter-term can be as well introduced in $\mathbf{F}$. Nevertheless, let us stress that such viscous terms are subdominant as compared to the bed friction $-\eta \alpha_3(h) \mathbf{u}$, being of order $\varepsilon^2$ smaller in the ration of length scales.

Special care has to be taken when evaluating the forcing term, since it contains higher order derivatives (the gradient $p_{\text{film}}$, which in turn embeds the Laplace pressure $\gamma \nabla^2 h$, see Eq. (7) and, hence, spurious lattice effects may arise. We noticed, for example, that a centered scheme to calculate gradients [21] does not guarantee the sufficient degree of isotropy on the lattice as, e.g., for the relaxation of a droplet (discussed in section IV), where it led to unphysical droplet shapes. Therefore, we use the following expressions to compute the gradients

$$\nabla \phi = 3 \sum_{l=0}^{8} w_l \mathbf{c}^{(l)} \phi(x + \mathbf{c}^{(l)}) + O(\varepsilon^3),$$

(19)

and the Laplacian

$$\nabla^2 \phi = 6 \sum_{l=1}^{8} w_l \phi^{(l)} + (w_0 - 1) \phi^{(0)},$$

(20)

respectively [31], for a generic scalar field $\phi$ (be it the height field $h$, the pressure $p_{\text{film}}$ or the topography of the substrate $b$). Besides the higher degree of the isotropy, the scheme (19-20) has the advantage of employing directly the set of LB speeds.

IV. RESULTS

A. Validation

We validated our implementation against standard tests for shallow water equations solvers. The first one of which is the settling problem over an arbitrary bed topography. For the so-called lake at rest problem we require that under the influence of gravity the velocities approach zero and that the lake’s surface becomes flat. The velocities vanish if

$$gh \nabla (h + b) = 0,$$

(21)

assuming that the surface tension effects are subleading and can be neglected for this example. If surface tension is neglected, the only case where the above equation can be fulfilled requires the sum of $h$ and $b$ to be constant. To show that this is indeed the case we use a bed topography which is given by

$$b(x) = 2 \sin \left( \frac{5\pi x}{L_x} \right)^2 \sin \left( \frac{5\pi y}{L_y} \right)^2,$$

(22)

and a height field which is initially randomly perturbed

$$h(x, 0) = h_0(1 + \varepsilon(x)).$$

(23)

In this first example we set $h_0 = 8$, $g = 1.8 \cdot 10^{-3}$ and $\gamma = 10^{-8}$ (corresponding to a capillary length $\lambda_c$ such that $\lambda_c/h_0 \sim 10^{-4}$, to ensure that capillary effects are negligible), with $\varepsilon \in [-0.2, 0.2]$. In Fig. 2 we show the initial condition and the system after 20000$\Delta t$, which has relaxed to a constant liquid surface. The observed error of the settling simulations can be computed as

$$\epsilon_h = \max(h(x) - b(x) - H_{\text{ref}}) / H_{\text{ref}}.$$  

(24)

It is observed that $\epsilon_h$ decreases and then saturates at a value of $\sim 1 \cdot 10^{-6}$. Fig. 3 shows that the profile of the film thickness is exactly shifted by $\pi$. Thus, the two sine functions cancel each other precisely, as it is shown in the upper panel of Fig 3. While in the lower panel we put them into phase again to show that they match perfectly.

In the following subsections we focus on results from simulations of film instability and droplet dynamics.

B. The Rayleigh-Taylor instability

The Rayleigh-Taylor instability occurs when a denser fluid is accelerated against a less dense one [32]. This can be the case, for instance, for a liquid film coating a ceiling, under the action of gravity. In such a configuration gravity tends, of course, to deform (and eventually disrupt) the film, while surface tension has a stabilizing effect. As a result of these competing mechanisms, any surface perturbation is stable or unstable depending on whether its characteristic wavenumber is smaller or larger than a certain critical value $k_c$. Linear stability analysis calculations in the framework of lubrication theory provide

$$k_c = \sqrt{\frac{\rho |g| h_0}{\gamma}},$$

(25)

where $\rho$ is the liquid density and $h_0$ the initial (average) film thickness [3].

On a lattice of size $2048 \times 2048$ nodes, we initialize the film height according to

$$h(x, 0) = h_0(1 + \varepsilon(x)),$$

(26)

with $\varepsilon$ a random variable homogeneously distributed in $[0.0001, -0.0001]$ and $h_0 = 1$. Forcing should always be below a certain threshold. Thus, for the gravitational acceleration we choose $|g| = 0.0001$. Furthermore, we fix the fluid density $\rho = 1$ and the surface tension $\gamma = 0.01$. This results in a critical wavenumber of $k_c = 0.1$. Fig. 4 shows snapshots from various time steps, where the growth of the perturbations is shown as time increases. The last panel is already beyond the linear regime.
FIG. 2: (Color online) Relaxation of an out-of-equilibrium surface over a complex bed topography. The left panel shows the initial state with the randomly perturbed height field. In the right panel we show the simulation output after $20000\Delta t$. As expected we observe a flat film.

FIG. 3: (Color online) Shape of the computed film thickness. In the upper panel the thickness as well as the bed are shown as a cut, the thickness is subtracted by $H_{\text{ref}} = 6.5$ to be exactly the negative of the bed. In the lower panel we subtract the thickness by $H_{\text{ref}} = 8.5$ and take the absolute value. The computed thickness and the analytical shape of the bed match perfectly.

To measure the growth and damping of wavemodes we perform a Fast Fourier Transform (FFT) of the height field according to

$$
\hat{h}(\mathbf{k}, t) = \int e^{-2\pi i \mathbf{k} \cdot \mathbf{x}} h(x, t) \, d\mathbf{k},
$$

with $\mathbf{k} = (k_x, k_y)$ and compute the two dimensional power spectrum which is given by $|\hat{h}(\mathbf{k}, t)|^2$. Since the result is isotropic we compute the azimuthally averaged one-dimensional power spectrum which is shown in Fig. 5. The various colors and symbols of Fig. 5 relate to different time steps within the simulation as the arrow indicates.

Consistent with the random initialization, at initialisation the spectrum is a constant (blue circles). As the time passes, $|\hat{h}(\mathbf{k}, t)|^2$ develops a profile which is higher for $k < k_c$, while modes with $k > k_c$ are damped out, in agreement with the expectation from the theory.

The theory predicts that $k_c$ is different for different values of either $g$ or $\gamma$. To show that this behavior is captured we perform simulations with different values of $g$ but keep $\gamma$ constant. We expect that the critical wavenumber grows with increasing gravity $\propto \sqrt{g}$. Whenever the azimuthally averaged power spectral density (PSD) cuts the initial PSD the critical wavenumber is found. In Fig. 6 data obtained for five values of $g$ is shown. The dashed line indicates the position of the theoretical value of $k_c$.

C. Spreading droplet

Let us consider the problem of a droplet, deposited on a smooth substrate with an apparent contact angle $\theta > \theta_{eq}$, which spreads to relax to a shape dictated by the equilibrium contact angle $\theta_{eq}$. The equilibrium contact angle quantifies the wettability of a given substrate by a certain liquid and can be calculated using Young’s equation [33]

$$
\gamma \cos \theta_{eq} = \gamma_{SL} - \gamma_{SG},
$$

with $\gamma_{SL}$ and $\gamma_{SG}$ being the surface tensions between solid/liquid and solid/gas, respectively.

In our simulations we set the equilibrium contact angle through the disjoining pressure (Eq. (8)). To probe the spreading, on a $512^2$ lattice we initialize a droplet, whose surface is given by the expression

$$
h(x, y, 0) = \sqrt{R^2 - (x - x_0)^2 - (y - y_0)^2} - R \cos \theta,
$$

where $R$ is the initial radius of the droplet and $(x_0, y_0)$ is its center.

To probe the spreading, on a $512^2$ lattice we initialize a droplet, whose surface is given by the expression

$$
h(x, y, 0) = \sqrt{R^2 - (x - x_0)^2 - (y - y_0)^2} - R \cos \theta,
$$

where $R$ is the initial radius of the droplet and $(x_0, y_0)$ is its center.
with \( R \sin \theta \approx 20 \Delta x \) \((\theta > \theta_{eq})\) being the radius of the droplet with a spherical cap shape, and \((x_0, y_0)\) its center. The droplet is placed in the middle of the lattice, i.e. \( x_0 = y_0 = 256 \Delta x \). In Fig. 7 we show such an initial shape, with contact angle \( \theta = 4\pi/9 \), and the converged one, associated to the imposed equilibrium contact angle \( \theta_{eq} = \pi/6 \). The corresponding profiles, of a cut across the plane \( y = L_y/2 \), i.e. \( \tilde{h}(x, t) = h(x, y = L_y/2, t) \), are plotted in Fig. 8. Out of these profiles we estimate the contact angles, fitting the data with a polynomial function \( f(x) \). The fit is taken within the interval \( I = \{a, b\} \), with \( a, b \) being the abscissa of the numerically evaluated gradient of \( \tilde{h} \). To overcome the precursor film the fit is continued beyond the points \( a, b \) and intersected with the "substrate" such that \( f(x_{CL}) = 0 \), with \( CL \) being the abbreviation for contact line. The contact angle is then given by

\[
\theta = \arctan \left( \frac{df(x)}{dx} \bigg|_{x=x_{CL}} \right). \tag{30}
\]

Although Fig. 7 and Fig. 8 contain the same information we emphasize that the droplet has indeed a spherical cap shape. It is important to note that the droplet maintains its spherical cap shape along the evolution. Let us stress that this is not trivial: as remarked in Section III, in fact, a sufficiently accurate finite difference scheme is required, as the one in Eqs. (19-20) \cite{31}. In particular,
(a) Initial droplet surface with $\theta = 4/9\pi$.

(b) The droplet surface spreads and relaxes to the expected shape, as dictated by the equilibrium contact angle.

FIG. 7: Relaxation of an out-of-equilibrium droplet.

FIG. 8: (Color online) Height profile at the center of the droplet. In blue we show the droplet at the initial state with $\theta > \theta_{eq}$. A later time step is shown in orange, where we observe $\theta = \theta_{eq}$. The $y$-axis shows the film height, while the $x$-axis denotes the position on the lattice.

we note that the isotropy of the pressure gradient is of utmost importance: a simple scheme with two-point centered derivatives [21] yields squared equilibrium droplet shapes.

The spreading dynamics can be investigated even more quantitatively in terms of the so-called Cox-Voinov law, relating the apparent contact angle to the velocity $U$ of the spreading front (the contact line), at various times, by $\theta^3 - \theta_{eq}^3 \propto Ca$. The capillary number $Ca$ is defined as $Ca = \eta U/\gamma$ [34]. In Fig. 9 we plot $\theta^3(t) - \theta_{eq}^3$ vs $Ca(t)$ from a numerical simulation of a spreading drop: a good linear scaling, in agreement with the Cox-Voinov law, is observed, as highlighted by the dashed line. Although the initial angle seems to be large it was shown in Ref. [5] that the Cox-Voinov law can be used up to angles of almost 180 degrees.

D. Sliding droplet

For a droplet to slide over an inclined plane, a minimum tilting angle $\alpha > 0$ is required [35], which in our case is due to the friction term Eq. (9). Until this critical angle is reached energy is stored in the deformation of the surface as the upper left inset in Fig. 10 shows. Above such a critical angle, a linear relation between the terminal sliding velocity $U_\infty$ and the gravitational force $m g \sin \alpha$ is observed [36–38]; in dimensionless numbers
such behaviour is expressed by

\[ Ca \propto Bo - Bo_c, \] (31)

where the capillary number is based on \( U_\infty \) and \( Bo \) is the so called Bond number, given by

\[ Bo = \left(3V/4\pi\right)^{2/3} \rho g \sin \alpha / \gamma. \] (32)

\( Bo_c \) is the critical Bond number, defined in terms of the critical titling angle \( \alpha_c \). In Fig. 10 we plot \( Ca \) vs \( Bo \) from our numerical simulations, showing that the phenomenology described by Eq. (31) is indeed reproduced, i.e. the onset of sliding takes place at a finite forcing, beyond which the linear scaling \( Ca \sim Bo \) is fulfilled.

E. Dewetting of liquid films

In order to show-case the capabilities of our method in handling more complex physics scenarios, we finally consider the dewetting of a chemically patterned substrate \[39, 40\]. This is easily made possible within the code by introducing a space-varying equilibrium contact angle, \( \theta_{eq}(x, y) \), in Eq. (8); in this way we can tune the local wettability of the substrate. Fig. 11 shows a liquid film which is initialized with thickness \( h(x, y, 0) \) randomly fluctuating in space around its mean value \( h_0 \), by a small percentage (\( \approx 10^{-2}\% \)) of it (panel (a)). A partially wettable substrate is patterned in such a way that the contact angle is lower on a region defining a logo. The total domain contains 512x512 lattice points. With this domain size a letter contains around 130 lattice points in y-direction and about 60 lattice points in x-directions. As the film dewets, liquid moves toward the letters of the logo, the surrounding film becomes thinner and eventually the logo becomes visible.

V. COMPUTATIONAL ASPECTS

We use OpenACC directives to allow our code to run on accelerator devices, such as Graphics Processing Units (GPUs), while being able, at the same time, to exploit the well known good scaling properties of LBM on parallel machines \[41\]. OpenACC is particularly versatile in terms of programmability since it only requires a few lines of code to allow us to harness the power of state of the art accelerators. Although the efficiency of a single precision (FP32) algorithm would perform much better on a GPU we choose double precision (FP64) in order to ensure both numerical accuracy and stability of the algorithm.

The performance of a LBM code is commonly measured in Million Lattice Updates Per Second (MLUPS), defined as

\[ \text{MLUPS} = \frac{A \times n}{t_{\text{sim}} \times 10^6}, \] (33)

with \( A = L_x \times L_y \) being the area of the lattice, where \( L_x, L_y \) are the number of lattice nodes in \( x \) and \( y \) directions. The number of iterations is given by \( n \). The time needed to compute the \( n \) iterations is called \( t_{\text{sim}} \) (in seconds). In Tab. I we provide benchmark data comparing the performance of a Nvidia GTX 1080TI, a Nvidia Quadro K2200 and a single core of an Intel i7-4790 @ 3.6GHz CPU. Due to the limited amount of memory available on the Quadro K2200, it is not possible to run a simulation of size \( 4096^2 \) on this card. Such a simulation requires about 4.8 GB local memory, while the Quadro K2200 only supplies 4 GB. In particular the speedup gained by using a GTX 1080TI is outstanding and corresponds to about 24-92 times the performance of a single core of the Intel CPU. Assuming perfect scaling on the CPU and using all 4 physical cores, the simulation on the GPU would be faster by a factor between 6 and 23. The speedup depends on the size of the lattice and in order to keep the pipelines on the GPU filled, a minimum loop size is needed. In addition, data transfer between host and device is a known bottleneck impacting the performance of GPU based simulations. This is obviously also the case for our code – even though such data transfer is only needed when files are written to disk.

VI. CONCLUSIONS

We have presented a novel lattice Boltzmann model for the numerical integration of the shallow water equations featuring explicitly relevant properties of interface physics, namely surface tension and disjoining pressure,
FIG. 11: (Color online) Time evolution of the free surface on a chemically patterned substrate on a 512x512 \( \Delta x^2 \) domain. Varying the contact angle between the letters and the rest of the substrate yields the shown dewetting pattern. The letters are more wettable then the rest. To emphasis the process we use a color gradient raging from dark blue to light blue. Starting from a randomly perturbed film height, the fluid starts to dewet the pattern (a) and after 2400\( \Delta t \) the letters and a surrounding rim structure are clearly visible. Towards the end of the simulation (c), the instability of the thin film also leads to film rupture. Holes form between the letters E, R and N.

TABLE I: Performance analysis based on a MLUPS measurement. The different columns relate to different lattice sizes, while the rows correspond to the two GPUs and one CPU used. All simulations are run for 100000\( \Delta t \) with FP64 double precision.

<table>
<thead>
<tr>
<th>Lattice/Accelerator</th>
<th>( 128^2 )</th>
<th>( 256^2 )</th>
<th>( 512^2 )</th>
<th>( 1024^2 )</th>
<th>( 2048^2 )</th>
<th>( 4096^2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>GTX 1080TI</td>
<td>157.6</td>
<td>279.2</td>
<td>382.6</td>
<td>414.9</td>
<td>404.7</td>
<td>395.6</td>
</tr>
<tr>
<td>Quadro K2200</td>
<td>33.5</td>
<td>42.9</td>
<td>46.6</td>
<td>48.2</td>
<td>49.0</td>
<td>X</td>
</tr>
<tr>
<td>i7-4790</td>
<td>6.4</td>
<td>5.8</td>
<td>4.5</td>
<td>4.6</td>
<td>4.5</td>
<td>4.3</td>
</tr>
</tbody>
</table>

thus being particularly suited for the simulation of thin liquid film flows.

We validated our implementation against various scenarios including the Rayleigh-Taylor instability, where the critical wavenumber as well as the growth and damping of wavemodes are correctly reproduced. Our simulations of droplets on substrates showed that droplets initiated out of equilibrium attain their equilibrium contact angle and that our method correctly reproduces the Cox-Voinov law. Furthermore, our approach allows to simulate the dynamics of sliding droplets and even complex dewetting scenarios.

Our OpenACC enabled simulation code allows for a massive improvement of the performance: with modern GPU cards at hand simulations using large lattice sizes and requiring many timesteps can be run on a single workstation without the need for access to high performance computing resources.

In the future we plan to extend our work towards systems which could hardly be tackled by traditional methods: from the dynamics of individual droplets on complex shaped substrates we plan to move to large numbers of droplets in order to understand the statistical properties of collective droplet motion on chemically structured substrates. Finally, a possible application of our method could be the simulation of full lab-on-chip devices with highly resolved channels, junctions, etc..

1[22], 1261 (2006).