# Efficient computations for high density ratio rising bubble flows using a diffused interface, coupled lattice Boltzmann-level set scheme

Mohammad Amin Safi*, Stefan Turek
*Corresponding author; amin.safi@math.tu-dortmund.de

Institute of Applied Mathematics (LSIII), TU Dortmund, Vogelpothsweg 87, D-44221 Dortmund, Germany

**Abstract**

A mass conserving, diffused interface, coupled Lattice Boltzmann-level set scheme is proposed and numerically studied for the simulation of high density and viscosity ratio multiphase flows. The approach is based on the pressure evolution formulation of the lattice Boltzmann equation, which is then coupled with the level set equation to capture a diffused level set function. Multiple relaxation time collision and isotropic force discretization techniques are employed to further reinforce the stability and accuracy of the LBM solver and a monolithic approach is employed to convect the level set function with minimal computations to preserve a smooth density profile. We present extensive investigations for numerical accuracy through performing benchmark simulations for rising bubble problems. It is observed that the proposed scheme is successful in producing accurate results as compared to those from high precision 2D finite element solutions in [5], offering a remarkable improvement in mass conservation and characteristic benchmark quantities upon our previous sharp interface coupled model described in [17]. Moreover, parallel scalability of the coupled LB scheme is shown to be preserved through performing efficient CPU- and GPGPU-based computations.

**Keywords** Rising bubbles; Lattice Boltzmann method; Level set method; High density ratio; GPGPU; Multiple relaxation time

## 1 Introduction

The Lattice Boltzmann Method (LBM) is widely known for its capability to facilitate capturing delicate transport phenomena, as well as its explicit structure of solution which allows for being efficiently accelerated on modern multi and many core compute architectures. Such desirable properties have in particular motivated researchers to export the methodologies, previously employed in the single- and multiphase solutions of the Navier-Stokes equations, to the framework of the LBM. A number of schemes are thus proposed with regards to the multiphase flow problems to bring the explicit realization of the interface capturing, as in the level set or volume of fluid methods, along to the lattice Boltzmann formulation as in [12, 17, 18]. Consequently, the computationally complex free energy or chemical potential description in conventional multiphase LB models are avoided.

Being one of the most established and conventional multiphase LB model for high density ratios, the chemical potential model, introduced by Lee an Liu [10], consists of solving two sets of LB equations; namely the lattice Boltzmann equation (LBE) for pressure and velocity based on the diffused chemical potential view of the interface forces, and the LBE for the order parameter which is derived from the Cahn-Hilliard equation

for the concentration (density) profile. While the method is known to be stable for high density ratios, the computational demand goes excessively beyond that of the early, weakly stable Shan-Chen or colour-fluid LB models, since a second $DnQm$ LBE has to be solved over the entire domain while both LBEs require spatial discretizations of force terms in all lattice directions. In addition, calculation of the chemical potential terms requires third and forth derivatives of the concentration profile, which subsequently implies having a high lattice resolution in the vicinity of the interface.

As a remedy to circumvent such high computational and numerical demands, one could re-design the LB method to solve for the pressure and velocity field and let a second differential equation capture the interface in a coupled fashion, which is analogous to that done in the multiphase Navier-Stokes framework. Examples are the works of Thomess et al. [18], Mehravarn [12] and Safi [17] using the level set equation and the work of Janssen et al. [7] using the volume of fluid technique.

While the coupled method of Thommes et al. is restricted to small viscosity differences, the early one-fluid model of Mehravaran and Hannani lacks isotropy and stability in force discretization and fails to repeat benchmark results for high density ratio problems. Recently, Safi and Turek [17] showed that the stability and accuracy of the one-fluid scheme could be profoundly enhanced by applying a consistent discretization to the interface forces as demonstrated for rising bubble benchmarks under high density and viscosity ratios. Moreover, the attained GPU-based parallel performances were shown to be significantly higher than those reported by similar implementations using the chemical potential LB models.

Nevertheless, derivation of the pressure in [17] is still based on the linear ideal gas equation of state (EOS) for a virtual fluid of density $\bar{\rho} = 1$, which limits the application of the method to slight dynamic pressure gradients, as also discussed by Mahravaran [12]. Additionally, coupled signed distance level set methods are generally associated with non-negligible mass losses of the bubble, which mainly occur during the course of reinitializing the level set function. Finally, the brute-force level set reinitialization employed in [17], although showed to be computationally superior to other schemes, becomes confusing to implement and realize when being applied to complex, bubbly flows which generally consist of multitudes of bubbles colliding and, possibly, merging together.

In this paper we propose to improve upon the coupled one-fluid approach of our previous work, through adopting the idea of diffused or pressure description of the interface forces as first proposed by He et al. [3] which takes the non-ideal part of the pressure into account and solves a pressure evolution LBE. The complex chemical potential form of the surface tension forces is then replaced with its conventional curvature-based definition. As opposed to the chemical potential LB models, a mass conserving level set equation is solved to capture a diffused interface which behaves similar to the concentration parameter in the Cahn-Hilliard equation, yet at much lower computational cost. This implies that the solution of a second LBE for resolving the concentration is avoided, while the additional computational effort brought by solving the level set equation is moderate and we show that only minimal implicit reinitializations are sufficient for accurate advection of the level set function.

As in our previous work we make use of multiple relaxation time collisions on the LBM side and employ robust force discretizations so as to guarantee the stability and isotropy of the solution. The accuracy of the solutions is then evaluated through performing benchmark comparisons against high precision finite element solutions of the Navier-Stokes equations coupled with level set methods for rising bubbles under moderate and high density and viscosity differences. As preserving the nice scalability of the algorithm is an indispensable part of any new LB-based model, we briefly point out the specific computational demands associated with the coupled algorithm and suggest parallel implementation outlines, particularly for many-core GPGPU architectures.

The rest of the paper is organized as follows; we present the derivation and discretization of the coupled pressure evolution LB model in section 2. Section 3 gives a brief overview of the special treatments for GPGPU implementation of the coupled scheme. The numerical results and validations for rising bubbles are then provided in section 4. Section 5 discusses the achieved parallel computational performances. Finally, the paper is closed with conclusions and discussions in section 6.

## 2 Pressure evolution coupled LB-level set scheme

At the presence of an external force $F$, the lattice Boltzmann equation in two dimensions with nine discrete velocities $e_k$, $k = 0, ..., 8$, describes the transport of the distribution functions $f_k$ as [4]

$$\frac{\partial f_k}{\partial t} + e_k.\nabla f_k = \sum_j \Lambda_{k,j}(f_j - f_j^{eq}(\rho, \mathbf{u})) + \frac{(e_k - \mathbf{u}) \cdot F}{\rho c_s^2} f_k^{eq} \tag{1}$$

in which the equilibrium state $f_k^{eq}$ is a function of density and velocity and is given by Maxwell equation as

$$f_k^{eq}(\rho, \mathbf{u}) = \omega_k \rho \left[1 + \frac{e_k \cdot \mathbf{u}}{c_s^2} + \frac{[e_k \cdot \mathbf{u}]^2}{2c_s^4} + \frac{[\mathbf{u} \cdot \mathbf{u}]}{2c_s^2}\right]. \tag{2}$$

In equations (1) and (2), $c_s = 1/\sqrt{3}$ is the lattice speed of sound and $\rho$ and $\mathbf{u}$ are the fluid density and velocity, respectively. Different moments of the distribution functions are then relaxed towards their respective equilibrium states in a multiple relaxation time fashion, i. e. $\Lambda = \mathbb{M}^{-1}\hat{\Lambda}\mathbb{M}$, where $\mathbb{M}$ is the transformation matrix

$$\mathbb{M} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -4 & -1 & 2 & -1 & 2 & -1 & 2 & -1 & 2 \\ 4 & -2 & 1 & -2 & 1 & -2 & 1 & -2 & 1 \\ 0 & 1 & 1 & 0 & -1 & -1 & -1 & 0 & 1 \\ 0 & -2 & 1 & 0 & -1 & 2 & -1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 0 & 0 & 1 & -2 & 1 & 0 & -1 & 2 & -1 \\ 0 & 1 & 0 & -1 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & -1 \end{bmatrix}. \tag{3}$$

and $\hat{\Lambda}$ is a diagonal relaxation matrix as in [2]

$$\hat{\Lambda} = \text{diag}\{0, s_1, s_2, 0, s_3, 0, s_3, 1/\tau, 1/\tau\}. \tag{4}$$

The relaxation time $\tau$ depends on the kinematic viscosity through $\nu = (\tau - 0.5)/3$. The parameters $s_1$, $s_2$ are typically set to 1.0 and $s_3$ is chosen in the range $0.3 < s_1 < 0.5$ as suggested in [17] for rising bubble problems.

For flows involving multiple phases, the diffused interface formulation of the LBE requires the force on the right hand-side of the LBE to account for the non-ideal effects emerging in the vicinity of the interface as well as for the interfacial surface tension forces as proposed by He et al. [3]

$$F = \nabla(\rho c_s^2 - P) + F_s + \mathbf{G} \tag{5}$$

where $\mathbf{G}$ is the gravity force and P represents the total pressure which is the sum of dynamic and thermodynamic pressures. In [3] the following form of surface tension force $F_s$ is adopted

$$F_s = \rho\gamma\nabla\nabla^2\rho \tag{6}$$

where $\gamma$ is related to the interface thickness $\varepsilon$ and the surface tension coefficient $\sigma$. Li and Liu [10] later proposed a modified form which transfers the chemical potential from of the thermodynamics pressure to the surface tension and leaves $P$ to account for the dynamic pressure only, such that the total force takes the following form

$$F = \nabla(\rho c_s^2 - P) - C\nabla\mu + \mathbf{G} \tag{7}$$

where the chemical potential $\mu$ is a function of the concentration $C$ as well as its Laplacian $\nabla^2 C$. The concentration $C$ is then solved using a second LBE for the Cahn-Hilliard equation which requires the thermodynamics part of the pressure. It is obvious that both equations (6) and (7) need three successive derivatives of the density profile. Furthermore, the second LBE in [10] will also require the forth derivative of $C$ so as to recover the diffusion term of the Cahn-Hilliard equation. Such multilayer approximations, not only leads to more computational complexity but also requires higher lattice resolutions near the interface to provide an adequately smooth density profile for numerical integration. In the present work, however, we aim to couple the LBM with the level set scheme and need not to involve any thermodynamic correlation. Consequently, the pressure $P$ in equation 5) is considered to be the dynamic pressure while we propose to replace the chemical potential form of the interface force with its classical curvature-based approximation

$$F_s = -\sigma\kappa\delta\mathbf{n} \tag{8}$$

where $\delta$ is the Dirac delta function and $\kappa$ and $\mathbf{n}$ are the mean curvature and normal vector to the interface, respectively. We will see later in this section that in order to obtain $\kappa$, one only needs the second derivative of a diffused level set function, hence removing the extra regularity and complexity imposed by equation (6) or (7).

In order to diminish the adverse numerical effect of the steep density changes $\nabla\rho$ arising in equation (5), He et al. suggested to convert the mass and momentum LBE (1) into a pressure evolution equation using the following transformation

$$g_k = f_k c_s^2 + (P - \rho c_s^2)\omega_k. \tag{9}$$

Taking the total derivative $D_t = \partial_t \cdot \nabla$ of $g_k$ and inserting the force term, one ends up with the following equation for the evolution of the pressure distribution function $g_k$

$$\frac{\partial g_k}{\partial t} + e_k.\nabla g_k = \sum_j \Lambda_{k,j}(g_j - g_j^{eq}) + (e_k - \mathbf{u}) \cdot \left[\nabla\rho c_s^2(\Gamma_k(\mathbf{u}) - \omega_k) + (F_s + \mathbf{G})\omega_k\right] \tag{10}$$

where $\Gamma_k(\mathbf{u}) = f_k^{eq}/\rho$ and $g_k^{eq} = f_k^{eq} c_s^2 + (P - \rho c_s^2)\omega_k$. We note that $\nabla P$ is eliminated from the right hand-side as $\nabla P \sim \mathcal{O}(Ma^2)$ and hence the term $(e_k - \mathbf{u}) \cdot \nabla P$ is reduced to $\mathcal{O}(Ma^3)$ [9]. Applying backward Euler and trapezoidal rule for time integration of the collision and forcing terms, respectively, we have

$$g_k(x + e_k\Delta t, t + \Delta t) = g_k(x,t) - \sum_j \Lambda_{k,j}(g_j(x,t) - g_j^{eq}(x,t)) - \frac{\Delta t}{2} S_k^U(x,t) + \frac{\Delta t}{2} S_k^C(x + e_k\Delta t, t + \Delta t) \tag{11}$$

with central and upwind forcing terms, $S_k^C$ and $S_k^U$ defined as

$$S_k^U = (e_k - \mathbf{u}) \cdot \left[\nabla^U \rho c_s^2(\Gamma_k(\mathbf{u}) - \omega_k) + (F_s + \mathbf{G})\omega_k\right] \tag{12}$$

$$S_k^C = (e_k - \mathbf{u}) \cdot \left[\nabla^C \rho c_s^2(\Gamma_k(\mathbf{u}) - \omega_k) + (F_s + \mathbf{G})\omega_k\right] \tag{13}$$

and $e_k \cdot \nabla^C$, $e_k \cdot \nabla^B$ are central and upwind differencing in the lattice direction $k$, as defined for any scalar $\phi$ [10]

$$\begin{aligned} e_k \cdot \nabla^C \phi &= (\phi(x + e_k \Delta t) - \phi(x - e_k \Delta t))/2 \\ e_k \cdot \nabla^U \phi &= (-\phi(x + 2e_k \Delta t) + \phi(x + e_k \Delta t) - \phi(x))/2. \end{aligned} \tag{14}$$

Equation (11) could then be rendered to an explicit form

$$\bar{g}_k(x + e_k \Delta t, t + \Delta t) = \bar{g}_k(x, t) - \sum_j \Lambda_{k,j}(\bar{g}_j(x, t) - \bar{g}_j^{eq}(x, t)) - \Delta t S^{Ave}(x, t) \tag{15}$$

where $S^{Ave} = (S^U + S^C)/2$ and the following transformations are used

$$\bar{g}_k = g_k - \frac{\Delta t}{2} S_k^C \tag{16}$$

and

$$\bar{g}_k^{eq} = g_k^{eq} - \frac{\Delta t}{2} S_k^C. \tag{17}$$

In order to track the interface, the second LBE in [3] or [10] is replaced here with a convection equation for a phase-field level set function $\psi$ as originally suggested by Olsson [14],

$$\partial_t \psi + \mathbf{u} \cdot \nabla \psi = 0. \tag{18}$$

In a gas-liquid multiphase system, the function $\psi$ gets the values of $\psi = 0$ inside the gas and $\psi = 1$ on the liquid side, while $\psi = 0.5$ represents the interface location. For a single circular gas bubble surrounded by the liquid, the level set function could be initialized as,

$$\psi = 1 - \frac{1}{1 + e^{d/\varepsilon}} \tag{19}$$

where $d$ is the signed distance from the interface $\Gamma$. In order to preserve a smooth profile of $\psi$ while it is convected, one may add an artificial diffusion along with a balancing compression to the right hand side of equation (18). Eventually, we propose to employ the following monolithic equation

$$\partial_t \psi + \mathbf{u} \cdot \nabla \psi = -\nabla \cdot (\varepsilon \psi (1 - \psi) \mathbf{n}) + \varepsilon \nabla^2 \psi \tag{20}$$

where the interface thickness $\varepsilon$ calibrates both the numerical diffusion and compression. This differs from the method suggested in [14], where $\varepsilon$ controls the diffusion term only and the reinitialization is carried out iteratively via solving a second PDE with small virtual time steps. On the other hand, assuming the same time and length scale for the level set equation as for the LBE, and considering the very small time steps of the LBE, a mere single iteration of equation (20) suffices to keep the smooth profile of $\psi$ and preserve a constant interface thickness. The very small time steps also suggest that one could treat both the convective term and the right-hand side of equation (20) explicitly, applying the backward Euler rule in time

$$\psi(x, t + \Delta t) = \psi(x, t) + \Delta t \left[-\mathbf{u}(x, t) \cdot \nabla \psi^n(x, t)\right] - \nabla \cdot (\varepsilon \psi(x, t)(1 - \psi(x, t))\mathbf{n}) + \varepsilon \nabla^2 \psi(x, t). \tag{21}$$

The small coefficient $\varepsilon$ could be chosen as a function of the physical resolution $h$ of the problem as also suggested in [14]

$$\varepsilon = \frac{h^\alpha}{2} \tag{22}$$

where the exponent $\alpha$ is chosen to be close to 1. We will discuss in the next section that the value of $\alpha$ could be adjusted to achieve minimum mass loss as the solution proceeds in time.

To discretize the convective term $\mathbf{u} \cdot \nabla\psi$, we use the weighted essentially non-oscillatory (WENO) scheme which extends the first-order upwind differencing to fifth-order spatial accuracy based on the smoothest possible interpolation of the level set function (see [15]).

Having the updated values of $\psi$, the fluid properties could finally be obtained based on the diffused level set function

$$\begin{cases} \rho(\psi) = \rho_g + (\rho_l - \rho_g)\psi \\ \mu(\psi) = \mu_g + (\mu_l - \mu_g)\psi \end{cases} \tag{23}$$

where the subscripts $g$ and $l$ stand for the gas and liquid properties, respectively. The interface curvature $\kappa$ and normal vector $\mathbf{n}$ could also be determined as functions of $\psi$

$$\mathbf{n}(\psi) = \frac{\nabla\psi}{|\nabla\psi|} \qquad , \qquad \kappa(\psi) = \nabla \cdot \mathbf{n} = \nabla \cdot \left( \frac{\nabla\psi}{|\nabla\psi|} \right). \tag{24}$$

Choosing the diffused Dirac delta as $\delta = |\nabla\psi|$, and using equations (8) and (24), $F_s$ is reduced to

$$F_s = -\sigma \nabla\psi \nabla \cdot \left( \frac{\nabla\psi}{|\nabla\psi|} \right). \tag{25}$$

Our numerical experiments show that applying the central differencing along the main $X$ and $Y$ directions would suffice for evaluating $\nabla\psi$ in equation (24), provided that the surface tension force is much smaller than the pressure forces due to the relatively small values of $\sigma$. The fluid pressure and the velocity are obtained using the moments of the distribution function $\bar{g}_k$

$$\rho\mathbf{u} = \sum_k e_{i,k}\bar{g}_k + \frac{\Delta t}{2} F_s \qquad , \qquad P = \sum_k \bar{g}_k + \frac{\Delta t}{2} \mathbf{u} \cdot \nabla^C(\rho c_s^2 - P). \tag{26}$$

Eventually, the overall algorithm for the coupled scheme looks like the following:

1. Calculate the directional derivatives of the density as well as the interface normal and its curvature;
2. Calculate the force terms based on equations (12) and (13) and equation (25);
3. Perform collision and streaming to update $\bar{g_k}$ using LB equation (15);
4. Solve the monolithic level set equation (21) to convect $\psi$ and update the density $\rho$;
5. Having the new density $\rho$ and the new $\bar{g_k}$, calculate velocity and pressure using equation (26).

## 3 GPGPU considerations

Efficient techniques for extracting the maximum power of GPGPUs for LBM solvers have been widely discussed in the literature. Optimized implementations for 2D and 3D single phase problems could be found in the works of Kuznik et. al [8] and Obrecht et. al [13] as well as for two phase LBM solvers in [6] and [11], which are mostly based on the early colour-fluid or Shan-Chen models for low density ratio systems.

For multiphase LB models based on the chemical potential or coupled level set schemes aimed at high density ratio systems, however, a significant increase in the computational workload emerges during the evaluation of the force term. This is because the force discretizations in the form of averaged and central directional derivatives are essential elements of such models. In the current coupled model using the D2Q9 stencil, this

means that one has to calculate 8 central and 8 upwind directional derivatives. For the chemical potential LB model of Lee and Liu [10] one has to add to this a second group of 16 differencings for $\nabla\mu$, along with 8 differencings for $\nabla P$ for the second LBE. An immediate, yet naive, implementation pattern will then be to launch a kernel to read the distribution functions, calculate 16 values and write them back to the memory. On the other hand, the most effective strategy to prevent performance losses in GPU-based implementations is to keep the unnecessary reads and writes at the lowest possible level, as the memory accesses account for the majority of the computational time in memory-bound algorithms, e. g. LBM. Therefore, instead of launching a kernel for the calculation of the force terms, we instead evaluate them on-the-fly inside the kernel for collision and streaming, where the distribution functions are already retrieved from the memory. Consequently, the averaged or central forms of the force will be available right before the collision step and only the weighted summed forces in the X and Y directions are saved to the memory to be used later in an individual kernel for updating velocity and pressure. Thanks to the vastly extended computational resources (e. g. shared memory size) per streaming multiprocessor (SM) in the modern Kepler GPGPUs, one could be sure that the inclusion of force computations alongside the collision and streaming would not impose a severe obstruction on the instruction throughput of the kernel.

# 4 Results

## *4.1 Test case definition*

In order to provide a comprehensive measure of the numerical accuracy of the presented multiphase LB schemes, we pick the well-established rising bubble benchmarks introduced by Hysing et al. [5], where two distinct rising bubble test cases are introduced and the incompressible multiphase Navier-Stokes equations coupled with the level set equation are solved using the finite element method. The problem configuration is illustrated in figure 1. The no-slip boundary condition is applied on the horizontal walls using the second order bounce-back scheme [19], while we impose periodic boundary conditions on the vertical boundaries which is almost identical, in this particular case, to assuming the slip boundary condition ($\mathbf{u}\cdot\mathbf{n}=0$) as in [5].

At $T=0$ the initially static bubble is allowed to rise due to the gravity force $\mathbf{G}=\mathbf{g}(\rho(\phi)-\rho_l)$, acting only upon the bubble. A summary of the specific inputs for the two test cases are given in LBM units in table 1, while the corresponding macroscopic values could be found in [5].

Table 1: Physical parameters and dimensionless numbers for test case 1 and 2, applied to static and rising bubble

| test case | $\rho_l$ | $\rho_g$ | $\mu_l$ | $\mu_g$ | $Re$ | $Eo$ | $\rho_l/\rho_g$ | $\mu_l/\mu_g$ |
|---|---|---|---|---|---|---|---|---|
| Test case 1 | 10 | 1 | 0.1 | 0.01 | 35 | 10 | 10 | 10 |
| Test case 2 | 250 | 0.25 | 2 | 0.02 | 35 | 125 | 1000 | 100 |

The surface tension coefficient $\sigma$ and the the gravity $\mathbf{g}$ are calculated in accordance with the desired values for the non-dimensional $Eo$ and $Re$ numbers

$$Eo=\frac{4\rho_l\mathbf{g}(r_0L)^2}{\sigma} \tag{27}$$

$$Re=\frac{\rho_1\sqrt{\mathbf{g}}(2r_0L)^{3/2}}{\mu_l} \tag{28}$$

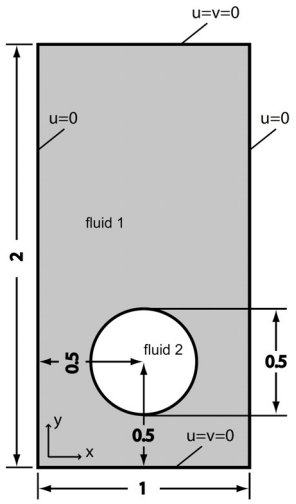



Fig. 1: Initial configuration and boundary conditions for the rising bubble problem [5]

where $r_0$ is the bubble radius as a fraction of 1 and $L$ is the number of lattice cells covering the $X$ dimension of the domain. Additionally, in order to have a consistent accordance with the time measurement in [5], we use the following equation to calculate the macroscopic time

$$T = t\sqrt{\frac{\mathbf{g}}{\mathbf{g}_0 L}} \tag{29}$$

were $\mathbf{g}_0 = 0.98$ is the macroscopic gravity used in [5] and $t$ is the time in LB units (iteration number). We note that the value of the phase densities has to be chosen with care, i. e. a large density leads to a very small physical time step according to equations (27) and (29) and hence implies a large number of lattice iterations to reach a certain physical time. On the other hand, choosing a small liquid density for the second test case leads to excessively large time steps, very small relaxation time inside the bubble and eventually high Mach numbers which in turn undermines the accuracy of the LBM.

To measure the bubble velocity $\mathbf{U_c}$ and its centroid position $X_c$, the following equations are used

$$\mathbf{U_c} = \frac{\int_{\Omega_b} \mathbf{u}\ dx}{\int_{\Omega_b} 1\ dx}$$

$$X_c = \frac{\int_{\Omega_b} \mathbf{x} dx}{\int_{\Omega_b} 1 dx}$$

where $\Omega_b$ represents the area inside the bubble. Additionally, the bubble deformation degree is described by the circularity $\circlearrowleft$ defined as

$$\circlearrowleft = \frac{P_a}{P_b} = \frac{\text{perimeter of area-equivalent circle}}{\text{perimeter of bubble}} = \frac{\pi d_a}{P_b}$$

### *4.2 Calibrating the interface thickness ε*

As discussed in section 2, the choice of the interface thickness $\varepsilon$ calibrates the amount of reinitialization and hence directly affects the conservation of mass inside the bubble, as well as its deformation rate. Figure 2 shows the effect of different values of $\alpha$ in equation (22), on the temporal value of the bubble mass and its circularity in test case 1 using a lattice resolution of $1/h = 320$. One could see that small values of $\alpha$ (i. e. larger $\varepsilon$) cause an adverse expansion in the initial course of rise. On the other hand, for $\alpha = 1$, the bubble undergoes an non-physical compression in the early iterations. Both expansion and compression effects mean that the bubble undergoes large variations of mass as apparent in figure 2(a), which has a subsequent impact on the bubble shape as reflected in the bubble circularity in figure 2(b),(c). One also should note that assuming $\alpha > 1$ leads to a too small interface thickness where the variation of $\psi$ is not smooth enough to resolve the interface gradients and results in early divergence of the solution. A favourable compromise is thus obtained for $\alpha = 0.87$ where the total mass change is less than 0.15%. Therefore, we apply $\alpha = 0.87$ in all simulations presented in this paper. Figure 2(a) also shows how effective the present level set function preserves the mass as compared to the variation of mass using the signed distance level set function in [17].

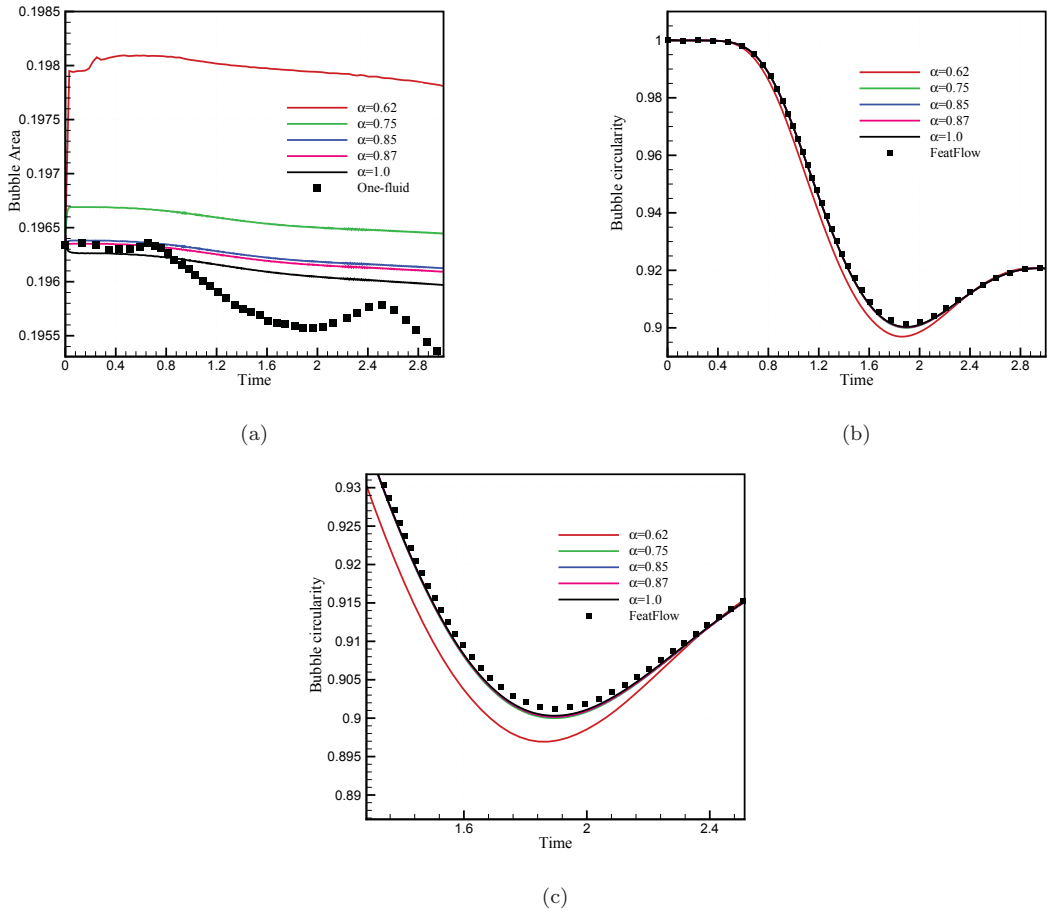
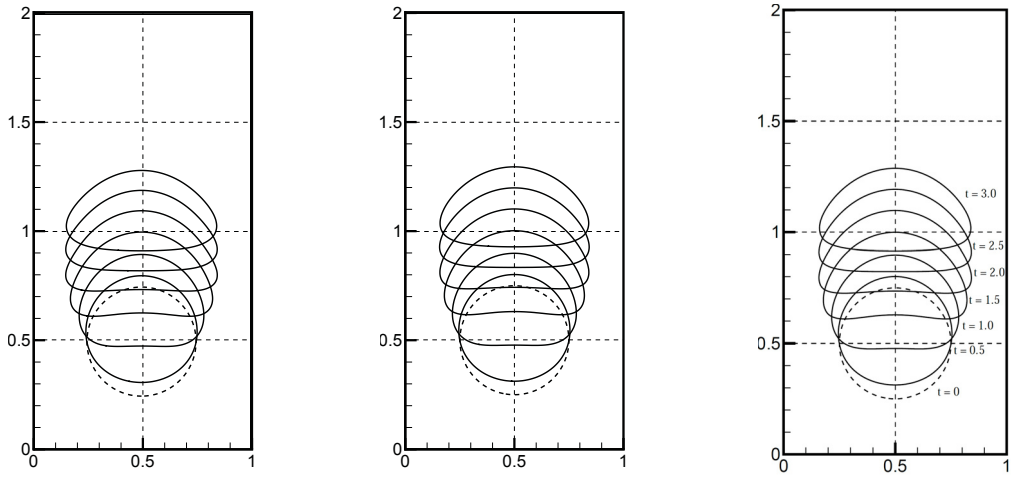



Fig. 2: Effect of choosing $\alpha$ on the solution of test case 1; (a) bubble area, (b) bubble circularity, (c) bubble circularity enlarged view

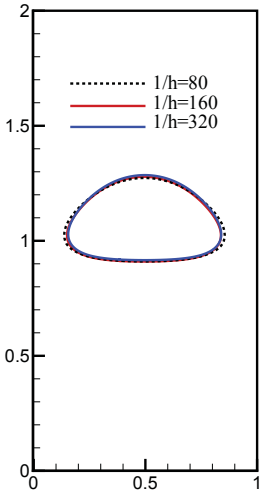
### *4.3 Test case 1*

Choosing the Eo number to a relatively low value of $Eo = 10$ in the first test case means that the surface tension effects are dominating the inertial ones. Considering a moderate Re number of $Re = 35$, the circular bubble is expected to evolve into an ellipsoid [1]. Figure 3 shows the time-lapsed evolution of the bubble interface using the current pressure evolution coupled LBM-level set and the one-fluid coupled scheme in [17], along with the reference finite element solution of the FeatFlow package in [5]. All methods use the same grid spacing of $h = 1/320$. From a qualitative perspective, one could see that both coupled schemes are successful in retrieving the expected bubble shape in different time stamps. Figure 4 also depicts the effect of using different resolutions on the interface shape at $T = 3$.




Press. Evol. LB-level set | One-fluid LB-level set | FeatFLOW

Fig. 3: Time evolution of the bubble shape for test case 1




Press. Evol. LB-level set

Fig. 4: Bubble interface shape at $T = 3$ for test case 1 on different lattice levels

Figures 5 shows the convergence trend in the solution using the present coupled scheme, for the centroid position, rising velocity and circularity of the bubble up to the nearly steady state time of $T = 3$. Figure 6 further presents the convergence behaviour of the $L_1, L_2$ and $L_{max}$ norm of error in the rising velocity with reference to $1/h = 640$, integrated over $NTS = 480$ time steps as

$$L_1 \quad error: \qquad ||e_1|| = \frac{\sum_{t=1}^{NTS} |q_t - q_{t,ref}|}{\sum_{t=1}^{NTS} |q_{t,ref}|}$$

$$L_2 \quad error: \qquad ||e_2|| = \left( \frac{\sum_{t=1}^{NTS} |q_t - q_{t,ref}|^2}{\sum_{t=1}^{NTS} |q_{t,ref}|^2} \right)^{1/2}$$

$$L_{\max} \; error: \qquad ||e_\infty|| = \frac{\max_t |q_t - q_{t,ref}|}{\max_t |q_{t,ref}|}$$

These errors are then calculated for the lattice sizes of $1/h = 40, 80, 160, 320$. One could see that a nearly second order of convergence is preserved on most lattice levels. The relatively lower convergence order for $1/h = 40$ is most likely caused by the lack of resolution in the post processing stage of calculating the mean bubble velocity. Finally, the obtained solutions using different schemes are compared in figures 7 for $1/h = 320$, revealing a closer agreement between the pressure evolution formulation and the benchmark data, as more discernible in the circularity diagram of figure 7(a)

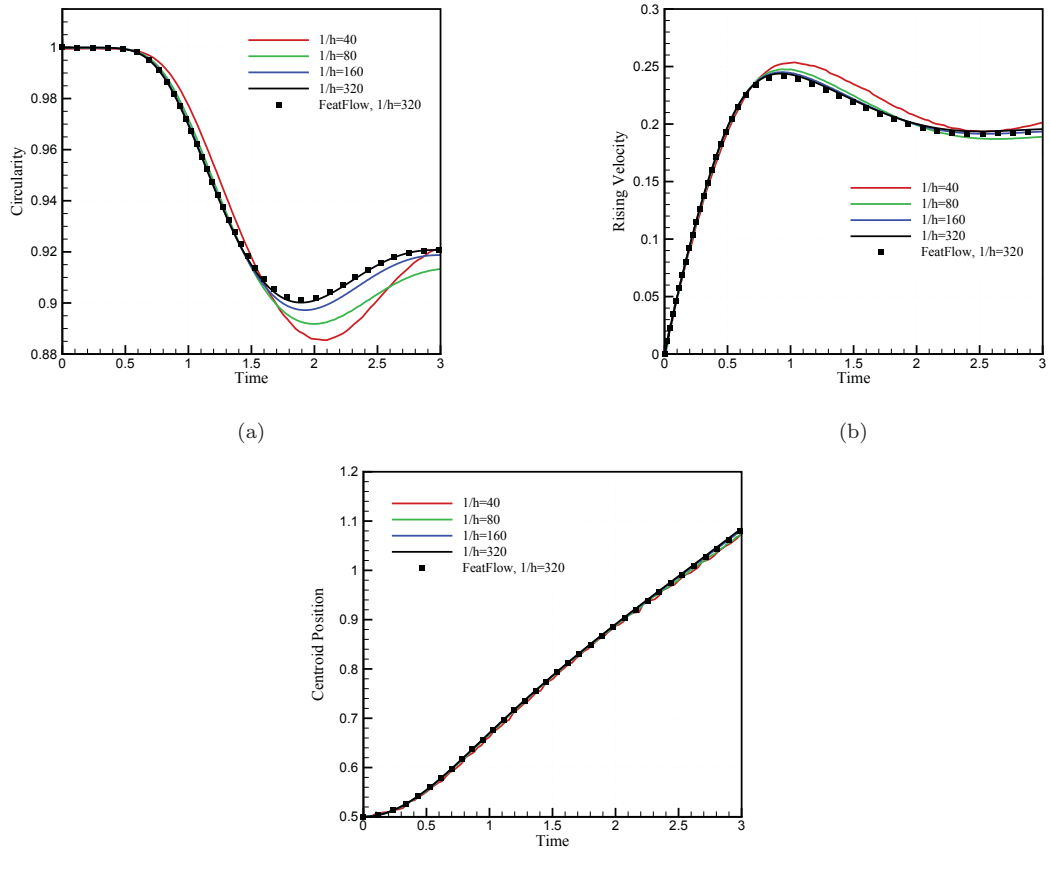



Fig. 5: Temporal values of bubble metrics on different grid levels using the pressure evolution coupled LB-level set scheme.

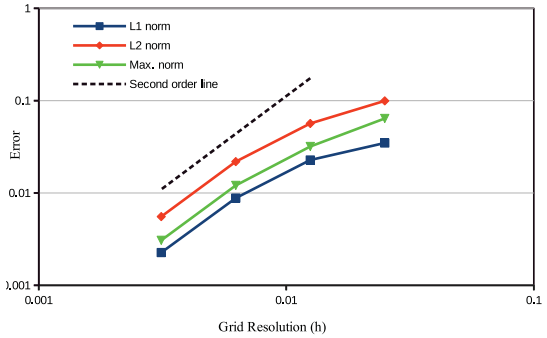



Fig. 6: Error convergence behaviour for rising velocity in test case 1

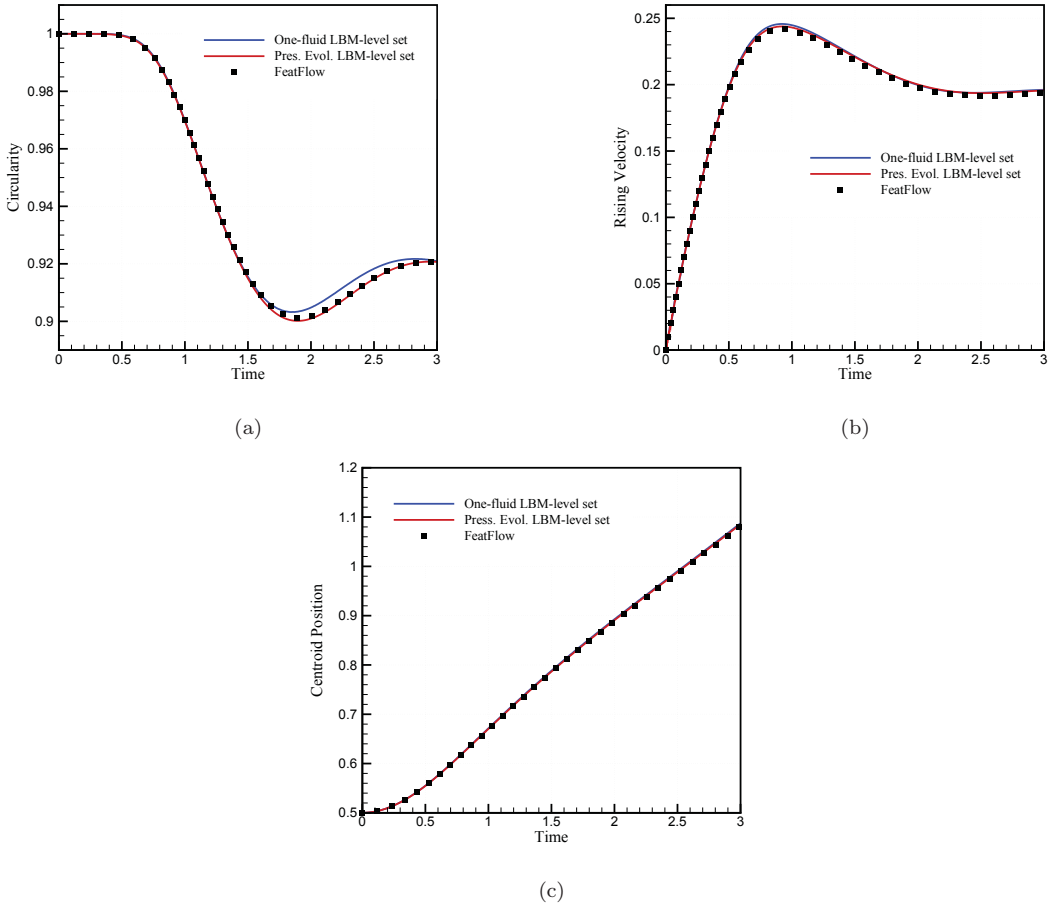
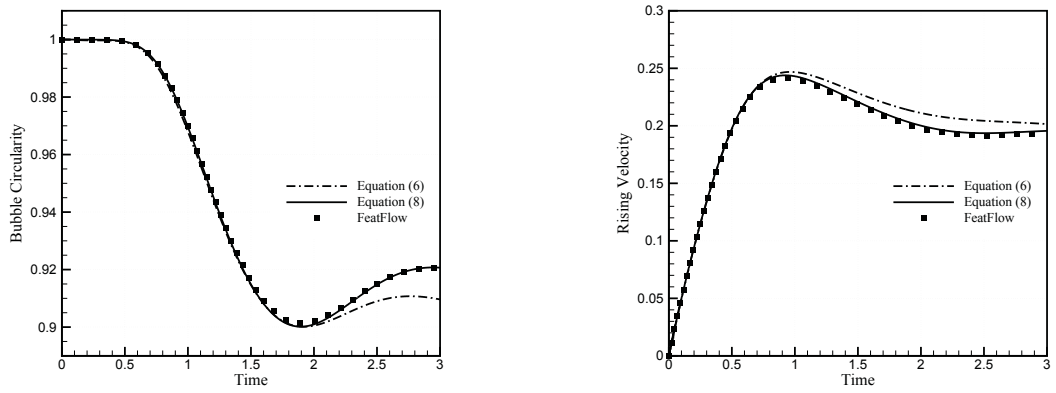



Fig. 7: Temporal values of bubble quantities in test case 1 using the the one-fluid in [17] and the pressure evolution coupled LB-level set schemes along with the finite element FeatFlow solution in [5]

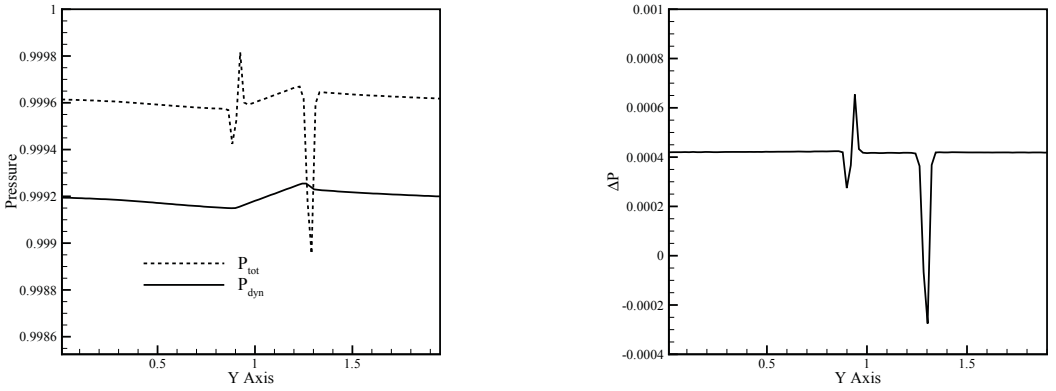
Lastly, one may question the accuracy of the surface tension description used by He et al. [3] given in equation (6) along with the total pressure, compared to the present formulation in equation (8) with the dynamic pressure. To this end and since the surface tension effects are dominating in our first test case, the temporal circularity and rising velocity arising from both equations are presented in figure 8 for test case 1. One could readily observe that deviations from the benchmark data obtained by equation (6) start to become apparent

as the bubble tries to retrieve its circular shape for $T > 1.5$. Such a discrepancy is partly due to the extra regularity required by the third derivative of the concentration in (6), but to a greater extent brought by the steep variations of the total pressure involved in the formulation of He et al. [3] as depicted in figure 9. In fact the highly non-monotonic changes of the thermodynamic part of the pressure, shown in figure 9(b), causes considerable discretization error near the interface and leads to erroneous prediction of the bubble quantities.




(a) (b)

Fig. 8: Temporal values of bubble circularity (a), and rising velocity (b), using different descriptions of surface tension force along with the benchmark solution in [5]




(a) (b)

Fig. 9: Variation of pressure along the Y axis at $X = 0.5$ ; (a) the total pressure ($P_{tot}$) arising from the He et al. formulation, and the dynamic pressure ($P_{dyn}$) using the the present formulation, (b) the difference between the total and dynamic pressure $\Delta P$.

#### 4.3.1 Test case 2

The problem configuration of the second test case becomes more numerically demanding as the density and viscosity ratios are increased to 1000 and 100, respectively. Moreover, by increasing the Eo number to $Eo = 125$ the bubble is allowed to exhibit more pronounced deformations since inertial effects take over the surface tension forces which typically tend to retain the bubble shape.

Figure 10 illustrates the temporal evolution of the bubble interface as it rises, produced with both the one fluid and the pressure evolution coupled schemes along with those predicted by finite element solutions of FeatFlow in [5]. All simulations are carried out on a grid resolution of $1/h = 320$. One could see that both methods reproduce the general bubble deformation pattern which eventually gives rise to a cap-shaped bubble with two trailing filaments (in case of LBM) or satellite bubbles (in case of FeatFlow). As contrary to the pressure evolution method, the one-fluid approach coupled with a sharp interface level set module predicts much thinner trailing filaments. It must be noted that unlike the LBM simulations, the finite element solution predicts the evolution of the trailing filaments into satellite bubbles. We emphasise that such a spin-off phenomena could not be physically verified since level set-based methods do not generally take into account the small-scale effects pertained to the chemical potential field in the vicinity of the interface. Consequently, the data provided in [5] for $T > 2$ are not considered as reference solutions.

Similar to the test case 1, a rather quantitative insight into the solutions is provided in figure 12 for bubble circularity, centroid position and rising velocity on different grid levels. We note that the sudden jump of the circularity values given by FeatFlow solutions points to the spin-off moment. A comparison of the rising velocity in figure 14 to those in figures 7 of test case 1 reveals that the bubble velocity in test case 2 attains two extreme values instead of one. The emergence of the second peak is believed to be due to the reduced drag force as the bubble sheds the trailing filaments and becomes more streamlined around $T = 2$. Consequently, the relatively stronger dependence of the diffused level set scheme on the lattice quality prevents the method to acquire the second peak even up to $1/h = 160$. Such an effect could be also noticed in figure 11 for the convergence of the terminal bubble shape, where the pressure evolution method requires a resolution of $1/h = 320$ to develop a deformation similar to that of a sharp interface solution.

Another key feature of the proposed scheme is the mechanism therein to resolve the pressure, as reflected in the temporal values of the rising velocity using the pressure evolution and one-fluid coupled schemes in figure 13. For relatively coarse lattices, e. g. $1/h = 80$, the one-fluid method shows overshoots in the rise velocity during the early times of rising as well as large deviation from the converged solutions for the entire course of rise. The primary reason behind such deviations is the limited accuracy of the one-fluid approach in resolving large pressure gradients. Instantaneous steep changes in the pressure often emerge around the stagnation point just above the bubble as it rises in the surrounding stationary liquid. In fact the ideal gas EOS used in [17] is expected to function accurately only for moderate pressure gradients. In contrast, the pressure evolution method removes the ideal gas contribution through defining the new distribution function in equation (9) and solves the LBE for the non-ideal part of the pressure. Consequently, one could observe a closer agreement between the converged solutions of the pressure evolution scheme and those of the benchmark data up to $T = 1.5$ in figure 14, which becomes more prominent for the rising velocity as in figure 14(c).

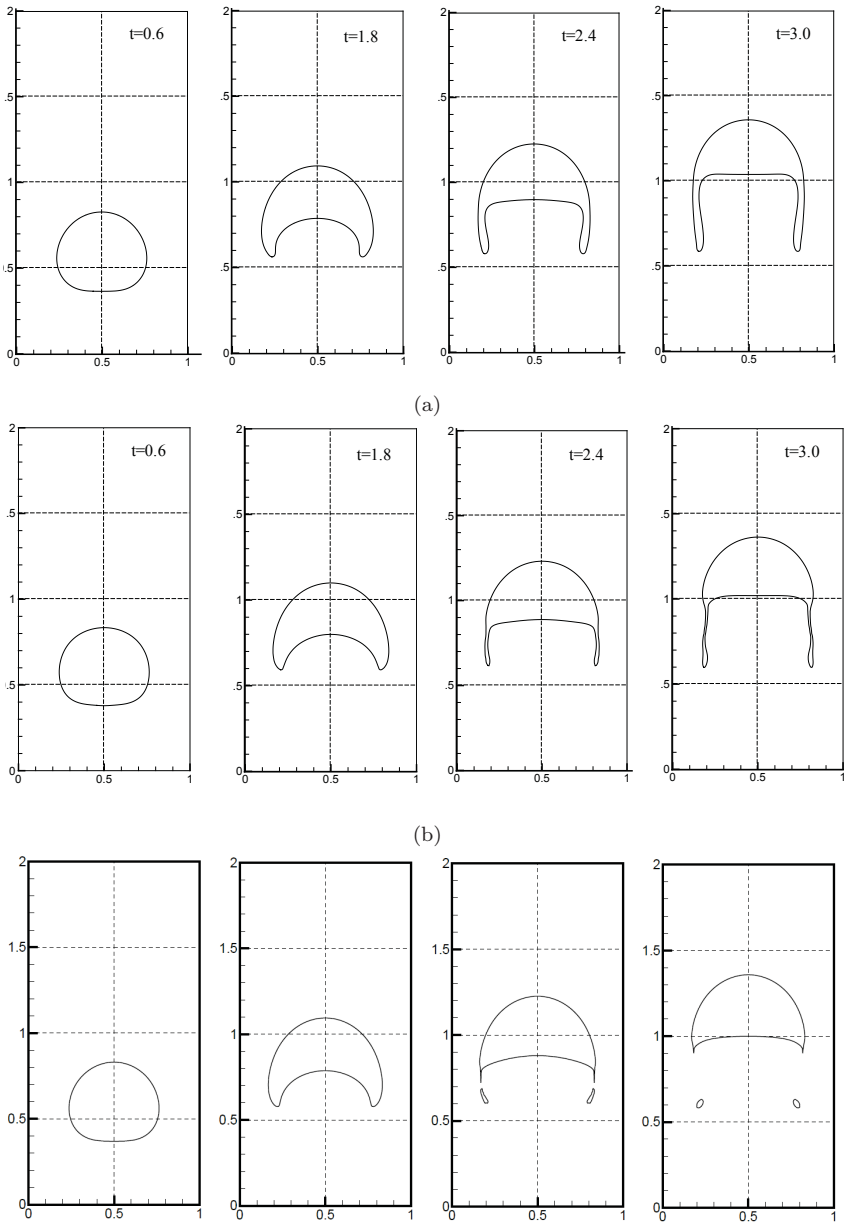
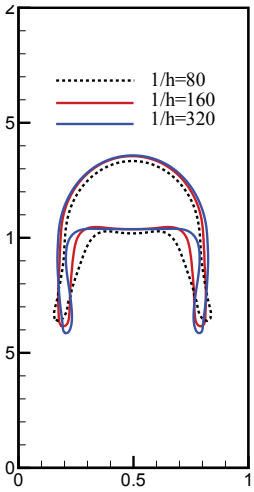
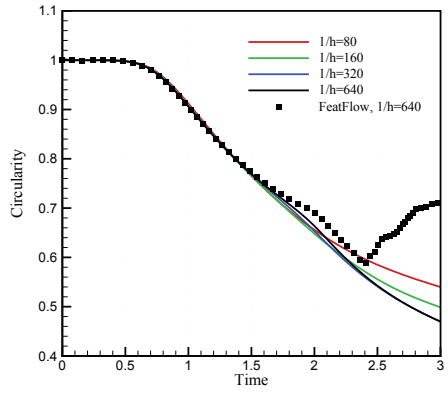



Fig. 10: Time evolution of the bubble in test case 2 for $1/h = 320$ using (a) pressure evolution LB-levelset (b) one-fluid LB-level set (c) FeatFlow [5].

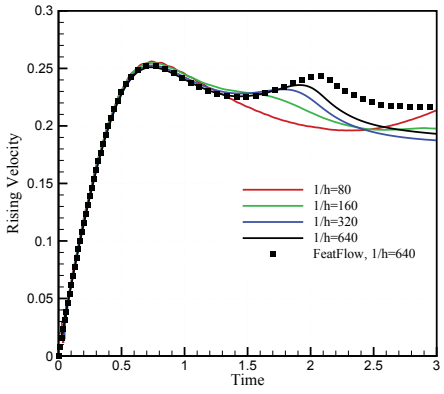



(a)

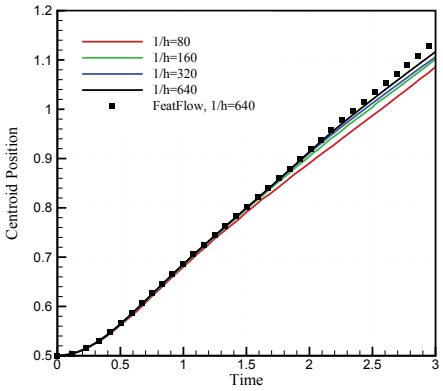
Fig. 11: Bubble interface shape at $T = 3$ for test case 2 on different lattice levels.




(a)

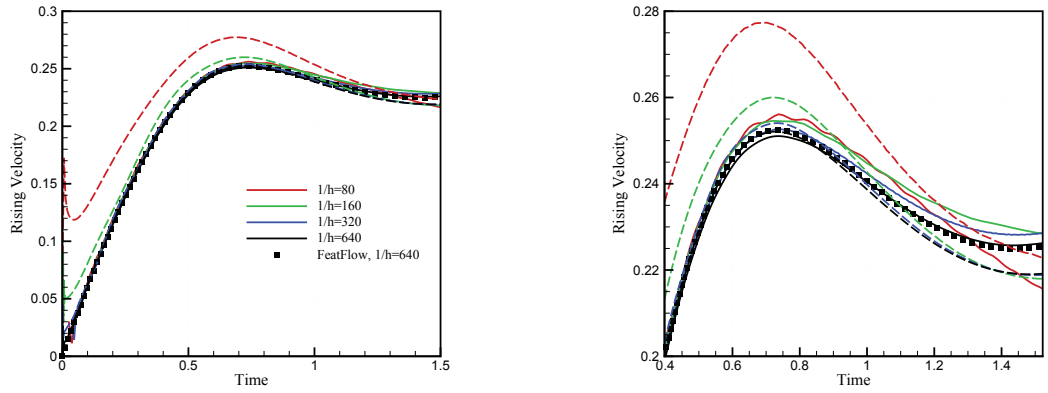



(b)



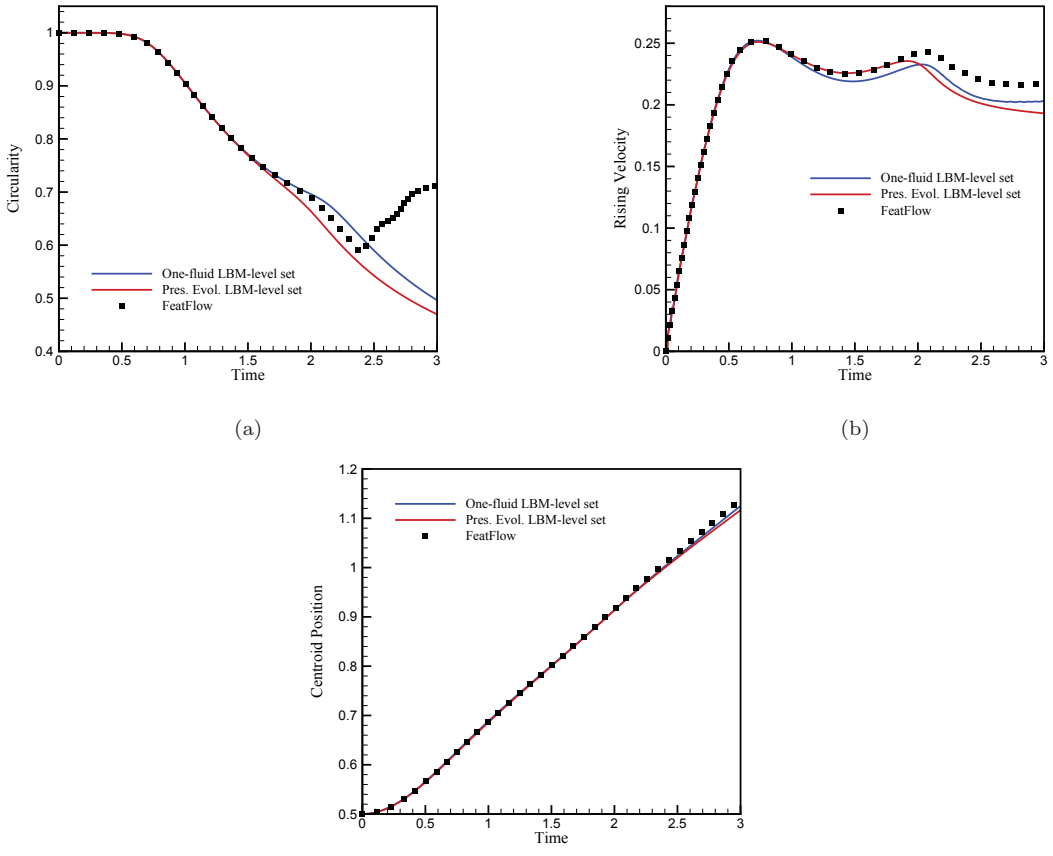

(c)

Fig. 12: Temporal values of bubble quantities in test case 2 using the pressure evolution coupled LB-level set scheme.




(a) (b)

Fig. 13: Bubble rising velocity on different lattice levels (a), and the enlarged view (b). Solid lines refer to the pressure evolution scheme while dashed lines refer to the one-fluid coupled method.




(a) (b)

(c)

Fig. 14: Temporal values of bubble metrics in test case 2 using different coupled schemes. The lattice resolutions is $1/h = 640$

### *4.4 Computational performance*

We use two distinct workstations for performing the parallel implementations; the parallel CPU code runs on a compute machine with Intel Xeon E5-2670 CPU having 32 processor cores at 2.60 GHz and peak memory bandwidth of 51 GB/sec. For GPGPU-based computations, we make use of another workstation equipped with a nVIDIA Kepler K20Xm GPU. It has 14 streaming multiprocessors (192 processor cores each) and a peak memory bandwidth of 250 GB/sec. The parallel shared memory The CPU code is based on the optimized parallel implementation described in [16] using OpenMP shared memory directives. The GPU code is compiled with nVIDIA's nvcc compiler under CUDA 6.0, where full optimization flags for both CPU and GPU codes are activated.

A weak scaling of the computational performance is given in figure 15 for both CPU and GPGPU implementations. One could notice that for sufficiently large problem sizes, almost one order of magnitude speed up is gained over the CPU implementation for both schemes. Another noteworthy observation is that the parallel CPU-based pressure evolution code offers 20-40% higher performances for small to moderate problem sizes as compared to that of the CPU-based one-fluid approach. Such a discrepancy is due to the fact that the brute force reinitialization method used for the one-fluid approach is computationally expensive when implemented on CPU, while monolithic reinitializations of the pressure evolution method remove the need to carry out re-distancing computations. However, re-distancing computations could be efficiently implemented on GPU as described in [17] or could become affordable on CPU as the problem size increases and parallel scaling and memory bandwidth usage improves. This could be seen clearly in the CPU- and GPU-based performances of the one-fluid method as they start to take over those of the pressure evolution methods for large problem sizes. Nevertheless, at an expense of less than 20% performance loss, one could obtain a more accurate pressure resolution while benefiting from a flexible reinitialization which is compatible to arbitrary shape changes and boundary conditions.

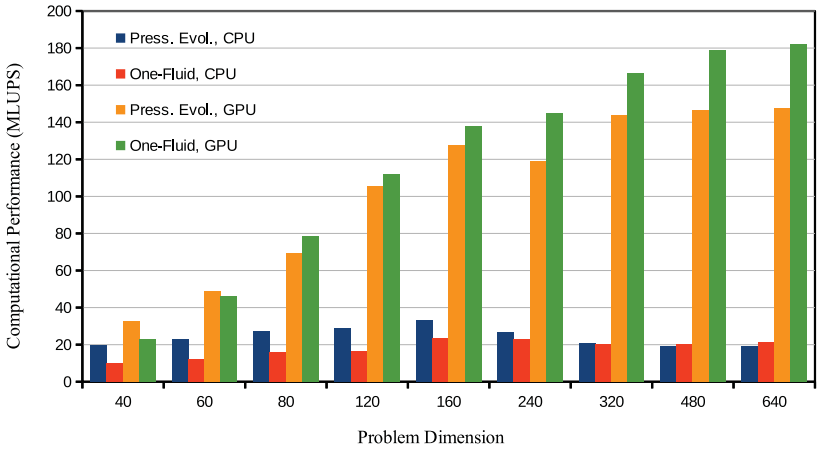



Fig. 15: Computational performance for various problem sizes using the two coupled schemes on CPU and GPGPU

## 5 Summary and conclusions

A diffused interface coupled LB-level set method for multiphase flow simulation was proposed and its numerical accuracy and efficiency were investigated for two benchmark problems of rising bubbles under moderate

to large density and viscosity ratios. By comparing the temporal quantities of the rising bubble to those of the reference solutions in [5] we show that the scheme provides promising accuracy and is able to handle complex bubble deformations. By combining the diffused interface capturing along with the pressure evolution LBE, a consistent coupled method is obtained where the pressure resolution is enhanced as compared to the coupled one-fluid approach and a higher accuracy with regards to benchmark data is achieved. Moreover, the employed mass conserving level set is reinitialized with minimal computational effort and provides flexibility for capturing more rigorous bubble deformations at a very negligible mass loss without the need to perform re-distancing computations or iterative PDE-based reinitializations. Yet, the computational performance of the coupled scheme is shown to be competitive to that of the previous one-fluid approach while the desired scalability of general LBM solvers is preserved as shown in the efficient parallel CPU and GPUGPU implementations.

The authors believe that a major improvement upon the present models could be achieved by employing adaptive mesh refinement (AMR) techniques to enhance the accuracy in resolving the pressure variations across the interface. This leads to higher accuracies for relatively less number of lattice points and saves considerable time and costs in the simulation if implemented efficiently on GPGPUs. In addition, in the case of less critical density ratios and hence more moderate pressure variations, one could improve the interface capturing quality by restricting AMR techniques to the level set grid only, as its underlying grid could be different than that for the LBE part. Early results in this regard are promising and are subject to our future publications.

**Acknowledgements**

The authors would like to acknowledge the support for this research from the Graduate School of Energy Efficient Production and Logistics of the state of Nordrhein-Westfalen at TU Dortmund, as well as the German Research Foundation (DFG) through the Priority Programme 1648 Software for Exascale Computing (SPPEXA): projects TU 10248-1 and GO 17582-1.

## References

[1] Grace J. R. Clift, R. and M. E. Weber. *Bubbles, Drops and Particles*. Academic Press, New York, First edition, 1978.

[2] A. Fakhari and T. Lee. Multiple-relaxation-time lattice boltzmann method for immiscible fluids at high reynolds numbers. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 87(2), 2013.

[3] X. He, S. Chen, and R. Zhang. A lattice boltzmann scheme for incompressible multiphase flow and its application in simulation of rayleigh-taylor instability. *Journal of Computational Physics*, 152(2):642–663, 1999.

[4] X. He, X. Shan, and G.D. Doolen. Discrete boltzmann equation model for nonideal gases. *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, 57(1):R13–R16, 1998.

[5] S. Hysing, S. Turek, D. Kuzmin, N. Parolini, E. Burman, S. Ganesan, and L. Tobiska. Quantitative benchmark computations of two-dimensional bubble dynamics. *International Journal for Numerical Methods in Fluids*, 60(11):1259–1288, 2009.

[6] M. Januszewski and M. Kostur. Sailfish: A flexible multi-gpu implementation of the lattice boltzmann method. *Computer Physics Communications*, 185(9):2350–2368, 2014.

[7] C.F. Janen, S.T. Grilli, and M. Krafczyk. On enhanced non-linear free surface flow simulations with a hybrid lbm-vof model. *Computers and Mathematics with Applications*, 65(2):211–229, 2013.

[8] F. Kuznik, C. Obrecht, G. Rusaouen, and J.-J. Roux. Lbm based flow simulation using gpu computing processor. *Computers and Mathematics with Applications*, 59(7):2380–2392, 2010.

[9] T. Lee and C.-L. Lin. A stable discretization of the lattice boltzmann equation for simulation of incompressible two-phase flows at high density ratio. *Journal of Computational Physics*, 206(1):16–47,

2005.

[10] T. Lee and L. Liu. Lattice boltzmann simulations of micron-scale drop impact on dry surfaces. *Journal of Computational Physics*, 229(20):8045–8063, 2010.

[11] X. Li, Y. Zhang, X. Wang, and W. Ge. Gpu-based numerical simulation of multi-phase flow in porous media using multiple-relaxation-time lattice boltzmann method. *Chemical Engineering Science*, 102:209–219, 2013.

[12] M. Mehravaran and S. Kazemzadeh Hannani. Simulation of buoyant bubble motion in viscous flows employing lattice boltzmann and level set methods. *Scientia Iranica*, 18(2):231 – 240, 2011.

[13] C. Obrecht, F. Kuznik, B. Tourancheau, and J.-J. Roux. A new approach to the lattice boltzmann method for graphics processing units. *Computers and Mathematics with Applications*, 61(12):3628–3638, 2011.

[14] E. Olsson and G. Kreiss. A conservative level set method for two phase flow. *Journal of Computational Physics*, 210(1):225–246, 2005.

[15] Fedkiw R. Osher, S. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, New York, First edition, 2003.

[16] A. Safi and M. Ashrafizaadeh. Entropic Lattice Boltzmann simulation of three dimensional binary gas mixture flow in packed beds using graphics processors. Technical report, Fakultät für Mathematik, TU Dortmund, September 2014. Ergebnisberichte des Instituts für Angewandte Mathematik, Nummer 504.

[17] A. Safi and S. Turek. Gpgpu–based rising bubble simulations using a mrt lattice Boltzmann method coupled with level set interface capturing. Technical report, Fakultät für Mathematik, TU Dortmund, December 2014. Ergebnisberichte des Instituts für Angewandte Mathematik, Nummer 508.

[18] G. Thommes, J. Becker, M. Junk, A.K. Vaikuntam, D. Kehrwald, A. Klar, K. Steiner, and A. Wiegmann. Numerical investigation of a combined lattice boltzmann-level set method for three-dimensional multiphase flow. *International Journal of Computational Fluid Dynamics*, 23(10):687–697, 2009.

[19] D. Yu, R. Mei, L.-S. Luo, and W. Shyy. Viscous flow computations with the method of lattice boltzmann equation. *Progress in Aerospace Sciences*, 39(5):329–367, 2003.