Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

An enhanced momentum conservation treatment for FDM simulation of two-phase flows with large density ratio

Xin Wang [a], Min Luo [b], Harshinie Karunarathna [a], Dominic E. Reeve [a,*]

[a] *Zienkiewicz Centre for Computational Engineering, College of Engineering, Swansea University, Swansea SA1 8EN, United Kingdom*
[b] *Ocean College, Zhejiang University, Zhoushan 316021, Zhejiang, China*

ARTICLE INFO

Article history:
Received 20 February 2022
Received in revised form 6 December 2022
Accepted 15 January 2023
Available online 20 January 2023

Keywords:
Two-phase flow
Momentum conservation
Fluid interface
Free surface
Computational methods
REEF3D

ABSTRACT

The differences of the fluid properties across a fluid interface in two-phase flow often bring difficulties into computational simulations, as the conservation of mass, momentum and energy requires careful consideration at the interfacial region. Velocity advection and unsynchronised variables lead to loss of conservation of momentum across the interface, which results in an unphysical interface deformation and spurious interfacial currents. In this study, we investigate the numerical errors and instabilities in the interfacial region, and propose a new algorithm with strong temporal coupling and momentum-based velocity reconstruction, to enhance its conservation properties. The capability of the proposed algorithm is demonstrated with two idealised cases including a one-dimensional convection case of a dense droplet and a standing wave case, and one laboratory dambreak case. Results are compared with theoretical results, experimental data or existing simulations, which demonstrate the advantages of the proposed algorithm on the conservation of mass, momentum and energy, and the mitigation of unphysical interfacial transport. Without modification of any numerical methods or discretization schemes, the algorithm keeps its simplicity and can work with the existing methods, and it is straightforward to implement.

1. Introduction

Two-phase flow is a frequent phenomenon in the real world, involving the interaction of two fluids with different physical properties. It is of great importance in both industry and as a research area in its own right. The presence of two-phase flow is observed over a variety of scales, e.g. molecular movement [1], lab-scale breaking waves [2], ocean and coastal waves [3], or even at the planetary scale [4]. With the development of computational methods, numerical simulation has become a widely used tool to study two-phase flow [5], as the field of Computational Fluid Dynamics (CFD), has flourished.

The numerical simulation of two-phase flow requires the proper treatment of the behaviours of the separate fluids and their interface. For grid-based CFD methods like the Finite Difference Method (FDM) and Finite Volume Method (FVM), there are two main approaches: Two Fluid Method (TFM) [6] and One Fluid Method (OFM) [7]. In TFM, the Baer–Nunziato model is proposed to describe the two-phase flow [8], with the known problem on conservation that the convective part in the model cannot be written in a conservative way [9]. Many methods have been developed to deal with the non-conservative

[*] Corresponding author.
E-mail address: d.e.reeve@swansea.ac.uk (D.E. Reeve).

https://doi.org/10.1016/j.jcp.2023.111949

terms, like [10–12]. The Baer–Nunziato model often requires 4 or 7 governing equations, indicating a larger demand for computational resources. Navier-Stokes equations (NSE) are also used in TFM, in which both fluids are described explicitly, thus two sets of NSE are needed [13]. The velocity slip and properties transfer between fluids are allowed [14], however, the exchange of mass, momentum and energy across the interface often suffers from large uncertainties, and surface capturing methods is still needed [15]. Additionally, TFM is more applicable for simple cases rather than realistic scenarios [16].

OFM assumes the same fluid occupies in the whole domain, but its physical properties vary temporally and spatially, i.e. the fluids in the whole domain are treated as a mixture. Only one set of NSE is needed to perform the simulation, which significantly saves simulation time. The fluid interface in OFM is described implicitly [17], and an accurate interface definition is essential for the accuracy and stability of OFM. As the interaction between the two fluids dominates the shape of the interface and affects the behaviours of the two fluids, it is essential to accurately capture the fluid interfaces. Interface tracking methods (e.g., Front-tracking method [18] and Marker-And-Cell method [19]) and interface capturing methods (e.g., the Volume of Fluid (VOF) method [20] and the Level – Set (LS) method [21], or the combination of the two, as with the CLSVOF method [22]) have been developed and employed widely. To avoid errors in interface definition, these surface capturing methods need to be coupled appropriately with the NSE, to make sure the variables and surface location are updated in a proper manner, especially in high-order temporal schemes, as the variables are updated more frequently inside one single timestep.

However, conservation of momentum and mass may still be improved across the fluids interface. One origin of errors is the advection term of the incompressible NSE. Large density ratios may lead to a large momentum ratio across the interface, and increase the risk of unphysical interfacial momentum exchange. This error may lead to an unphysical velocity field across the surface, resulting in a deformed surface profile and interfacial spurious currents [23,17], or an inaccurate interface definition.

Another source of errors is the limitation of the surface capturing methods, e.g., the LS method is known to have the problem on volume and mass loss in under-resolved regions [24] while the VOF method often generates inaccurate curvature and lower smoothness at the interfacial region, which may lead to an inaccurate velocity field [25]. Numerous treatments or adjustments have been developed to deal with these foregoing problems. E.g., geometric operations [26–28] are implemented into VOF to improve the accuracy of the interface shape; extra smoothing [29], variable density projection [30], additional diffusion equations [31,32], artificial compression and 'viscosity' have been introduced into the LS method; special boundary conditions are applied on the fluid interface [33]; adaptive mesh refinement has also been used close to the interface to improve accuracy [34]; higher order surface capturing methods have been developed to reduce the numerical error [17,35,36]; interface tracking methods have also been coupled with interface capturing methods [37], as well as particle methods [33].

However, with these modifications, algorithms have become more and more complex and surface capturing methods tend to lose their own simplicity [38]. The balance among accuracy, stability, efficiency and simplicity has become a pressing problem. Meanwhile, the standard forms of these methods or algorithms are still widely used, like the classical LS method [39] or algorithms based on the classical LS method [40,41].

To address these limitations, the present study proposes an alternative approach. Rather than modifying the existing methods or discretization schemes, we are seeking a more appropriate computational procedure that can improve the accuracy and stability with least effort while reducing the need for modifications of the numerical methods. In this paper we propose a momentum-based velocity reconstruction step, coupled with the surface capturing process in a synchronised manner, which enhances the consistency and improves the interfacial smoothness, stability and conservation. As no change is made to the underlying scheme or method, our modification can work with many different schemes and improved surface-capturing methods, while maintaining simplicity and ease of implementation into existing CFD codes.

The paper is organized as follows: the basic theories are introduced in Section 2. The origination of the error is discussed in Section 3. The algorithm modifications are proposed in Section 4. The algorithm is then tested on an idealised case and a laboratory dambreak case to demonstrate its performance in Section 5. Conclusions are presented in Section 6.

2. Numerical methods

The FDM-based open-access CFD code REEF3D [42–44] is used as the underlying tool of the present study. The governing equations and key numerical schemes are introduced in the following [45].

2.1. The incompressible isothermal NSE with constant density and viscosity

The governing equations are the conservation laws of mass and momentum for incompressible flows, written in the nonconservative form as follows:

$$\frac{\partial u_i}{\partial x_i} = 0 \tag{1}$$

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu\left[\frac{\partial}{\partial x_j}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right] + g_i \tag{2}$$

where u_i is the velocity, ρ the density, p the pressure, ν the kinematic viscosity, g_i the gravitational acceleration. Turbulence is ignored in the present study.

A standard two-step projection method [46] is used to solve the incompressible NSE. In the prediction step, the pressure gradient term is ignored to solve for an intermediate velocity field u^{int} as:

$$\frac{u_i^{int}-u_i^n}{\Delta t}=-u_j^n\frac{\partial u_i^n}{\partial x_j}+\nu\left[\frac{\partial}{\partial x_j}\left(\frac{\partial u_i^n}{\partial x_j}+\frac{\partial u_j^n}{\partial x_i}\right)\right]+g_i \tag{3}$$

In the correction step, the fluid velocities are updated by considering the pressure gradient as:

$$\frac{u_i^{n+1}-u_i^{int}}{\Delta t}=-\frac{1}{\rho}\frac{\partial p^{n+1}}{\partial x_i} \tag{4}$$

Applying the incompressibility condition (i.e., Eq. (1)) to Eq. (4) to gives the Pressure Poisson Equation (PPE) as:

$$\frac{\partial}{\partial x_i}\left(\frac{\partial p^{n+1}}{\partial x_i}\right)=\frac{\rho}{\Delta t}\cdot\frac{\partial u_i^{int}}{\partial x_i} \tag{5}$$

Solving the PPE yields the fluid pressure at the current time step, which is then used to update the fluid velocities through Eq. (4).

2.2. Level Set method

The classical Level-Set (LS) method [21] is used to capture the water-air interface. An LS function $\varphi(x,t)$ is defined that represents a signed distance, which satisfies $|\nabla\varphi|=1$. According to the φ values of each computational node, the fluid condition of nodes can be determined as follows:

$$\varphi(x,y,z,t)\begin{cases}>0, & \text{if node (x, y, z) is in phase1}\\ =0, & \text{if node (x, y, z) is on the surface}\\ <0, & \text{if node (x, y, z) is in phase2}\end{cases} \tag{6}$$

The update of the LS function φ is achieved by solving the following advection equation:

$$\frac{\partial\varphi}{\partial t}+u_j\frac{\partial\varphi}{\partial x_j}=0 \tag{7}$$

To ensure that φ remains a signed distance function, after the advection, the φ value at each node is re-initialized as follows [47]:

$$\frac{\partial\varphi}{\partial t}+S(\varphi)\left(\left|\frac{\partial\varphi}{\partial x_j}\right|-1\right)=0 \tag{8}$$

In simulation, a Heaviside function is defined as follows according to the φ value:

$$H(\varphi)=\begin{cases}0, if\,\varphi<-\epsilon\Delta x\\ \frac{1}{2}\left[1+\frac{\varphi}{\epsilon\Delta x}+\frac{1}{\pi}\sin\left(\frac{\pi\varphi}{\epsilon\Delta x}\right)\right], if\ |\varphi|\le\epsilon\ \Delta x\\ 1, if\,\varphi>\epsilon\Delta x\end{cases} \tag{9}$$

The density and viscosity of nodes are then calculated by considering the Heaviside function value, as shown in Eq. (10):

$$\begin{cases}\rho_i=\rho_1 H(\varphi_i)+\rho_2(1-H(\varphi_i))\\ \nu_i=\nu_1 H(\varphi_i)+\nu_2(1-H(\varphi_i))\end{cases} \tag{10}$$

With Eqs. (9) and (10), the water-air interface is represented by an interfacial region with a distance of $\epsilon\Delta x$ (Δx is local grid spacing) towards both sides of the interface, which ensures the smooth variation of fluid properties across the interface.

2.3. Spatial and temporal discretization schemes

A staggered grid system is applied in the simulation. Scalar variables (e.g., density, viscosity and LS value φ) are defined at cell centres, while vector variables (e.g., velocity, momentum) are defined at the centre of cell edges, as shown in Fig. 1.

We have used three schemes to perform the spatial discretization for the advection term in Eqs. (3) and (7), and found the 5th order weighted essentially non-oscillatory (WENO5) produces the best result (based on the cases in the present study). The central difference scheme is applied for the spatial discretization of the viscous force term in Eq. (3), the pressure gradient in Eq. (4), and the pressure Laplacian and the velocity divergence in Eq. (5).

The time propagation of the advection term in Eq. (3) is achieved using the 3rd order Runge-Kutta Total Variation Diminishing (RK3-TVD) scheme, which has 3rd order accuracy and small storage requirements. For the viscous term in Eq. (3), we used the implicit backward time centred space scheme as it allows larger time steps and has good stability properties.

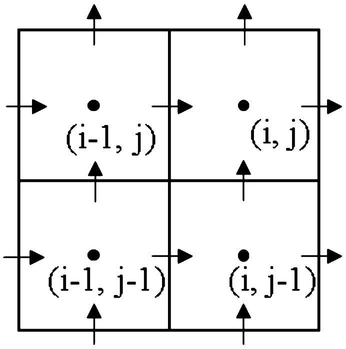

Fig. 1. An 2D staggered grid system (points denote scalar variables and arrows denote vector variables).

3. Investigations on numerical errors near fluid interface

The numerical algorithms presented in Section 2 may lead to spurious interfacial kinematics and dynamics, as presented in Section 5 and previous studies [50,51]. These numerical errors are attributed to the unphysical transport of momentum across the interface, which are caused by two primary factors: non-conservation of momentum in the spatial discretization of the advection term [17] and inaccuracy in the definition of fluid interface [52]. Note that a higher interfacial density ratio would amplify this error. How the two factors erode the numerical accuracy are investigated below.

3.1. Non-conservation of momentum in advection term

One source of the non-conservation of momentum is the form of discretisation. Consider the conservative advection term $\frac{\partial(\rho u)\cdot u}{\partial x}$ and its nonconservative form $u\frac{\partial(\rho u)}{\partial x}+\rho u\frac{\partial u}{\partial x}$, and their discretised forms as follows:

$$\left(\frac{\partial\left(\rho u\right)\cdot u}{\partial x}\right)_{i-0.5}=\frac{1}{\Delta x}((\rho u)_i\cdot u_i-(\rho u)_{i-1}\cdot u_{i-1}) \tag{11a}$$

$$\left(u\frac{\partial\left(\rho u\right)}{\partial x}+\rho u\frac{\partial u}{\partial x}\right)_{i-0.5}=\frac{1}{\Delta x}u_{i-0.5}\cdot\left((\rho u)_i-(\rho u)_{i-1}\right)+\frac{1}{\Delta x}(\rho u)_{i-0.5}\cdot(u_i-u_{i-1}) \tag{11b}$$

In a steady state, a unique horizontal flux at each point is ensured by the conservative form rather than nonconservative form. Adding up all flux terms along a given direction, (the x-direction is presented here), the conservative form provides zero net flux, while extra source terms arise with the nonconservative form. Thus, a conservative spatial discretisation is preferred for the conservation of momentum [57].

Although a conservative form is adopted, non-conservation of momentum around the fluid interface may still develop from the advection of velocity which does not consider the density change. Taking the one-dimensional (1D) advection case in Fig. 2 as an example, consider two inviscid fluids with a density ratio of $\rho_1/\rho_2=10^6$ located either side of the interface Γ^n, (and let $\epsilon=1$, see Eq. (9)). In the n^{th} time step, fluid 1 has a constant velocity u_0 while fluid 2 is at rest, and the velocity changes smoothly from u_0 to zero. In the subsequent (i.e. $n+1^{\text{th}}$) time step, the interface moves to Γ^{n+1}. Following the strategy in Section 2, the velocity at point (i+0.5) is updated by velocity advection, which can be performed by different spatial schemes (e.g. WENO5, upwind or central difference).

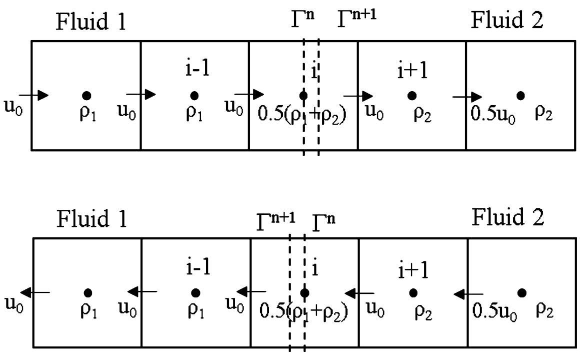

Fig. 2. A 1D case to illustrate the errors of momentum and density.

Table 1
Relative error of updated momentum at point (i+0.5) by velocity advection.

Spatial scheme for Velocity advection	WENO5	Upwind	Central difference
$U_0 > 0$	2.5%	5%	7.5%
$U_0 < 0$	4.3%	10%	7.5%

Regarding the density ratio, the case can be seen as a pure advection case and theoretically the velocity at $n+1^{\mathrm{th}}$ timestep at point (i+0.5) should stay u_0, with an updated momentum of $u_0\rho_{i+0.5}^{n+1}$. Table 1 shows the relative error of the numerical results of the updated momentum at point (i+0.5) obtained by taking the production of the advected velocity and the updated density (obtained from Γ^{n+1}) (Let the CFL number be 0.1). It is noted although the numerical results of WENO5 show the smallest discrepancies (due to its higher-order accuracy), errors of velocity advection close to the fluids interface (i.e. point (i+0.5)) cause non-negligible momentum deviation and extra pressure. Thus, a velocity advection scheme that preserves the momentum is crucially necessary.

3.2. Temporal mismatch of fluid variables in time propagation

Time propagation schemes often consist of multi substages, which brings the consideration of temporal synchronization in the coupling of surface-capturing and NSE-solving steps. The present work adopts the RK3-TVD scheme [53] for the advection terms, which consists of three sub-stages as below:

$$\begin{cases} f^{(1)} = f^n + \Delta t L(f^n) \\ f^{(2)} = \frac{3}{4} f^n + \frac{1}{4} f^{(1)} + \frac{1}{4}\Delta t L(f^{(1)}) \\ f^{n+1} = \frac{1}{3} f^n + \frac{2}{3} f^{(2)} + \frac{2}{3}\Delta t L(f^{(2)}) \end{cases} \tag{12}$$

where f^n, $f^{(1)}$, $f^{(2)}$ and f^{n+1} are the fluid quantities at time t, $t+\Delta t$, $t+0.5\Delta t$, and $t+\Delta t$, respectively, and $L(f^n)$ is the time derivative of fluid quantity at the corresponding time instant. Fig. 3 illustrates the updates of the LS and velocity values in a time step of the RK3-TVD scheme, which is widely used in CFD simulations [32,39] and some variants are also presented by [54,55].

Note that the weak coupling between LS and NSE at the 2nd and 3rd substages may lead to an inaccurate interface definition, which can result in non-conservation of momentum. The first concern is that the update of fluid quantities, (obtaining ρ^{n+1} and ν^{n+1} according to φ^{n+1}), is completed before solving the NSE, like [42], and no coupling is found between $\varphi^{(1)}$ and u^{stage1} (also $\varphi^{(2)}$ and u^{stage2}). The second concern is the selection of $u^{stage\ 1}$ and $u^{stage\ 2}$, as using u^n for $u^{stage\ 1}$ and $u^{stage\ 2}$ may generate a temporal mismatch. Temporal extrapolation based on u^n and u^{n-1} [17,57] fulfils the requirement of time matching, but the coupling between LS and NSE at these two substages is still not considered, as the temporal extrapolation solely relies on the velocity field. Alternatively, $u^{stage\ 1}$ and $u^{stage\ 2}$ can be obtained by solving the corresponding NSE at these two substages ($t+\Delta t$, $t+0.5\Delta t$) [45], which requires the corresponding density and viscosity field (thus $\varphi^{(1)}$ and $\varphi^{(2)}$). Thus, a proper coupling between LS and NSE is needed. Additionally, it is noted the above analysis is based on using the same time propagation scheme for both LS and NSE. Applying different time propagation schemes to them makes the synchronization more difficult [56].

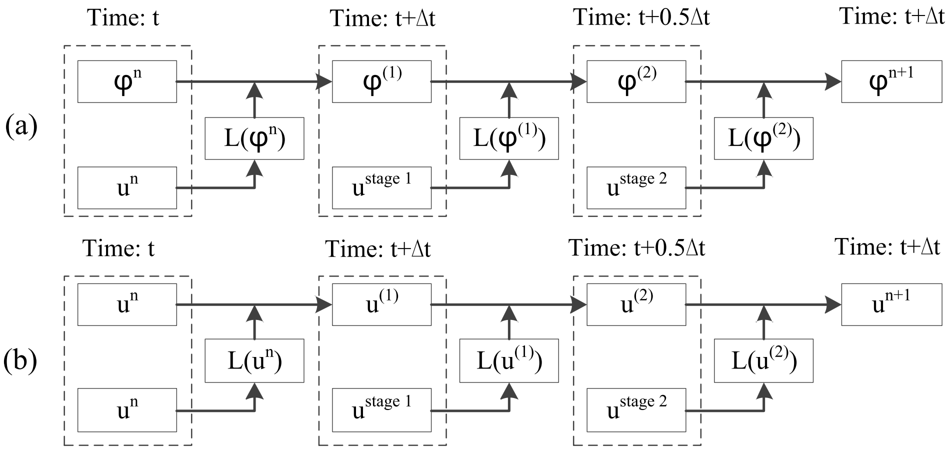

Fig. 3. Updates of the LS (a) and velocity (b) values in a time step of the RK3-TVD scheme.

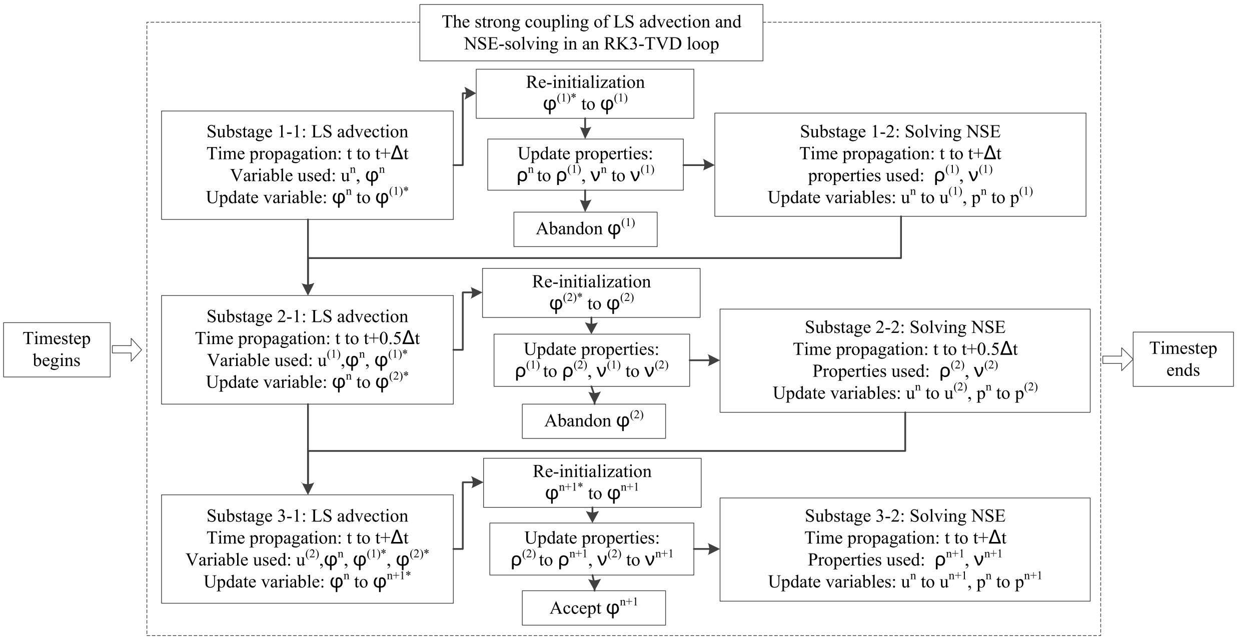
The strong coupling of LS advection and NSE-solving in an RK3-TVD loop
Timestep begins
Substage 1-1: LS advection
Time propagation: t to t+Δt
Variable used: u^n, φ^n
Update variable: φ^n to φ^(1)*
Re-initialization
φ^(1)* to φ^(1)
Update properties:
ρ^n to ρ^(1), ν^n to ν^(1)
Abandon φ^(1)
Substage 1-2: Solving NSE
Time propagation: t to t+Δt
properties used: ρ^(1), ν^(1)
Update variables: u^n to u^(1), p^n to p^(1)
Substage 2-1: LS advection
Time propagation: t to t+0.5Δt
Variable used: u^(1),φ^n, φ^(1)*
Update variable: φ^n to φ^(2)*
Re-initialization
φ^(2)* to φ^(2)
Update properties:
ρ^(1) to ρ^(2), ν^(1) to ν^(2)
Abandon φ^(2)
Substage 2-2: Solving NSE
Time propagation: t to t+0.5Δt
Properties used: ρ^(2), ν^(2)
Update variables: u^n to u^(2), p^n to p^(2)
Substage 3-1: LS advection
Time propagation: t to t+Δt
Variable used: u^(2),φ^n, φ^(1)*, φ^(2)*
Update variable: φ^n to φ^n+1*
Re-initialization
φ^n+1* to φ^n+1
Update properties:
ρ^(2) to ρ^n+1, ν^(2) to ν^n+1
Accept φ^n+1
Substage 3-2: Solving NSE
Time propagation: t to t+Δt
Properties used: ρ^n+1, ν^n+1
Update variables: u^n to u^n+1, p^n to p^n+1
Timestep ends

Fig. 4. Revised calculation sequence for RK3-TVD schemes of LS and NSE.

3.3. *Influences of interfacial thickness*

To capture more flow details and achieve a higher accuracy, a thinner fluid interface is often needed, which requires a smaller grid spacing Δx (grid refining) or a smaller interfacial thickness parameter ϵ (interface sharpening). However, these adjustments may increase the risk of non-conservation of momentum [23]. In the illustrative example of Fig. 2, points $(i-1)$ and $(i+1)$ are both in the interfacial region and point (i) is on the interface. According to Eqs. (9) and (10), the density at the three points can be calculated as:

$$\begin{cases} \rho_{i-1} = \frac{1}{2}\left[1 + \frac{1}{\epsilon} + \frac{1}{\pi}\sin\left(\frac{\pi}{\epsilon}\right)\right](\rho_1 - \rho_2) + \rho_2 \\ \rho_i = \frac{1}{2}(\rho_1 + \rho_2) \\ \rho_{i+1} = \frac{1}{2}\left[1 - \frac{1}{\epsilon} - \frac{1}{\pi}\sin\left(\frac{\pi}{\epsilon}\right)\right](\rho_1 - \rho_2) + \rho_2 \end{cases} \tag{13}$$

Note that a lower Δx or ϵ increases the local density gradient across the fluid interface. As explained in [17], large density gradient close to the interface may aggravate the spurious momentum transfer and lead to the non-conservation of momentum.

Additionally, the error in the advected velocity (e.g., point (i+0.5) in Fig. 2) leads to an unphysical velocity gradient in the denser fluid, which may also increase the instability at a lower Δx or ϵ. According to Fig. 2, reducing Δx could increase this unphysical velocity gradient and results in a larger error on the local pressure field by solving the PPE (Eq. (5)). This error may be further amplified due to the increased local pressure gradient at a lower Δx through Eq. (4). Meanwhile, reducing ϵ increases the updated density at point (i+0.5) (i.e., $\rho_{i+0.5}^{n+1}$), which leads to a higher error on the updated momentum and adversely affects the momentum conservation. In summary, as will be proved in Section 5, the interfacial instability develops faster with grid refining and interface sharpening with the strategy introduced in Section 2.

4. Enhanced momentum conservation (EMC) treatment

4.1. *Strong time-coupling in high-order temporal scheme*

For the problem of variable synchronization (Section 3.2), we suggest a stronger coupling order in Fig. 4. For each substage of RK3-TVD scheme, LS function is advected firstly according to Eq. (7), then the re-initialized LS values ($\varphi^{(1)}$, $\varphi^{(2)}$, φ^{n+1}) are used immediately to define ν and ρ for solving the corresponding NSE, and the updated velocity ($u^{(1)}$, $u^{(2)}$, u^{n+1}) is used immediately for the calculation of time derivatives of φ ($L(\varphi^{(1)}), L(\varphi^{(2)}), L(\varphi^{n+1})$).

Compared with the original scheme in Fig. 3(a) which only requires one re-initialization in each timestep, the proposed manner performs three times of re-initialization, which may amplify the well-known volume-loss problem of the classical LS method [24], and the re-initialized LS value ($\varphi^{(1)}$, $\varphi^{(2)}$) may also break the consistency of the RK3-TVD scheme of the 2nd and 3rd substages. Thus, the un-reinitialized LS values ($\varphi^{(1)*}$, $\varphi^{(2)*}$) are accepted for the advection of the next substage, instead of the re-initialized LS values ($\varphi^{(1)}$, $\varphi^{(2)}$) which are solely used for updating fluid properties. Only the re-initialized LS value in the last substage (φ^{n+1}) is accepted for the advection of the next timestep, to avoid the potential error from multi re-initializations in one timestep.

4.2. Momentum-based velocity reconstruction

Regarding the issue of velocity advection (Section 3.1), we propose a velocity-reconstruction technique. Based on the conservative form of NSE (Eqs. (14) and (15)), we perform the conservative momentum advection, as shown in Eq. (16):

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \cdot u) = 0 \tag{14}$$

$$\frac{\partial (\rho \cdot u)}{\partial t} + \nabla \cdot (\rho u \otimes u) = -\nabla p + \nabla \cdot \tau + \rho g \tag{15}$$

$$\frac{\partial M}{\partial t} + \nabla \cdot (M \otimes u) = 0 \tag{16}$$

in which M is the momentum calculated from $\rho \cdot u$. The advections of velocity and density are also tracked simultaneously for further use, as shown in Eqs. (17) and (18). Eq. (19) shows the explicit form of advections of M, u and ρ.

$$\frac{\partial u}{\partial t} + \nabla \cdot (u \otimes u) = 0 \tag{17}$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \otimes u) = 0 \tag{18}$$

$$u_i^* = u_i^n - \Delta t \frac{\partial u_j^n u_i^n}{\partial x_j} \tag{19a}$$

$$M_i^* = M_i^n - \Delta t \frac{\partial u_j^n M_i^n}{\partial x_j} \tag{19b}$$

$$\rho_i^* = \rho_i^n - \Delta t \frac{\partial u_j^n \rho_i^n}{\partial x_j} \tag{19c}$$

The advected velocity field $u^{*,r}$ is then restored based on the quotient of M^* and ρ^*, as shown in Eq. (20):

$$u_i^{*,r} = \frac{M_i^*}{\rho_i^*} \tag{20}$$

For the spatial matching of the density and momentum of Eq. (20) in a staggered grid system, the density advection (Eq. (19c)) is performed in a vectorised manner based on the interpolated face-centred density.

The velocity reconstruction process shares the same framework as [52,58,59] in which the velocity is restored in the same way, but differences appear on the convective density and momentum. The density flux often relies on the surface-capturing method, e.g. the density flux in [52] comes from the VOF flux, and [52,58–60] trace the motion of the interface from the LS values and reconstruct the density flux by geometric arguments. Compared with the straightforward Eq. (19c), the geometric argument may have extra limitations, like [60] requires a sharp interface ($\varphi = 0$), as the convective density is hard to define if the density changes continuously across the interface. Additionally, more effort may be needed to reconstruct a complex interface shape, leading to an increasingly complex geometric argument, which makes it challenging to perform in 3D conditions [61]. It is noted the intermediate density (ρ_i^*) from Eq. (19c) should be matched with the intermediate momentum (M_i^*), rather than the density defined by LS method at the end of the timestep (ρ_i^{n+1}).

The momentum flux is taken as the production of advected velocity and density flux in [52,60,61]. However, we still perform the momentum advection based on the advection equation (Eq. (19b)), to ensure the consideration of velocity gradient (recall that $\frac{\partial M}{\partial x} = u\frac{\partial \rho}{\partial x} + \rho\frac{\partial u}{\partial x}$). As the same advection equation, velocity field and numerical operation are applied on the update of the intermediate density and momentum, better spatial and temporal matching between them is ensured.

A simple 1D advection test is designed to demonstrate the conservation of momentum during interface moving. As shown in Fig. 5, an interface divides a 1D domain with 400 gridcells, (grid spacing Δx=0.005 m), and the denser fluid 1 is on the left and the less dense fluid 2 on the right, (the density ratio being 10^6). Fluid 1 is initialized with the velocity of ±1 m/s and the CFL number is 0.1. Open boundary conditions are applied on both sides, and gravity and viscosity are ignored. No interface deformation is supposed to appear in this case.

This case is simulated by both the proposed velocity reconstruction technique and the original velocity advection method. Compared with the analytical solution, the momentum discrepancies of both methods after 1 s (1000 loops) are shown in Fig. 6, in which it is observed the momentum conservation is improved by the velocity reconstruction technique, as the

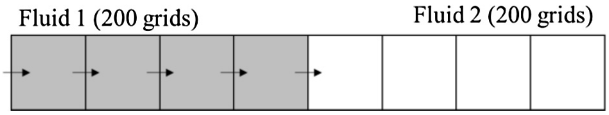

Fig. 5. Grid settings for the 1D advection case.

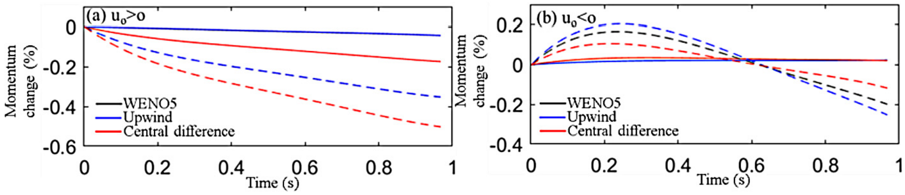

Fig. 6. Comparison of momentum error after 1 s (momentum change is normalized by the initial momentum of the fluid 1 of one grid, dashed line – velocity advection; solid line – velocity reconstruction) (a): positive initial velocity; (b): negative initial velocity.

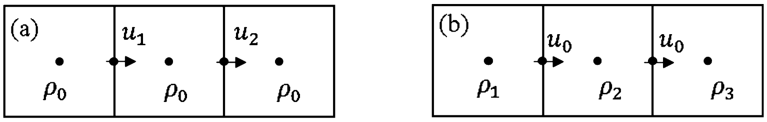

Fig. 7. Two idealized cases for the demonstration of applying the same spatial scheme.

momentum change is much more stable and is no more than 0.04% of the initial momentum of one cell of fluid 1 with WENO5 and upwind schemes.

After the velocity reconstruction, the viscous and gravitational terms in Eq. (3) are integrated into the restored velocity field, yielding the intermediate velocity field u^{int}. Eq. (4) and (5) are then solved for pressure update and velocity correction. In contrast to the method proposed by [17] in which the intermediate density field ρ_i^* is used in Eq. (4) and (5) or similarly [61] who uses $\rho_i^{n+0.5}$, ρ_i^* is immediately discarded after the velocity restoration in our strategy. As the LS value at the next iteration is already known, ρ^{n+1} is used in Eq. (4) and (5), which is defined by the corresponding φ^{n+1}.

Potential errors may rise in the velocity restoration, and we applied 2 treatments: 1) applying the same spatial schemes for variable advections (momentum M, velocity u, density ρ and LS value φ); 2) applying a 'limiter' treatment around the boundary between the interfacial region and less dense fluid. The details are described in the following sections.

4.3. Control the interaction between truncation errors

The truncation errors of M and ρ may interact with each other in Eq. (20) and reduce the accuracy of the restored velocity. To tackle this undesirable effect, we suggest that the advection of u, M and ρ follow the same spatial scheme. Consider two simple 1D convection cases as shown in Fig. 7, let the truncation error be proportional to the accurate advection term, the updated values of u, M and ρ, denoted by asterisks, are shown in Eq. (21). The coefficients a_1, a_2 and a_3 denote the proportionate truncation errors of u, M and ρ.

$$\begin{cases} u_x^* = u_x^n - (1+a_1)\Delta t \frac{\partial u_x^n u_x^n}{\partial x} \\ M_x^* = M_x^n - (1+a_2)\Delta t \frac{\partial u_x^n M_x^n}{\partial x} \\ \rho_x^* = \rho_x^n - (1+a_3)\Delta t \frac{\partial u_x^n \rho_x^n}{\partial x} \end{cases} \tag{21}$$

In the case of an accelerating fluid of constant density, Fig. 7(a), density gradient is not accepted after the advection process. Hence, the restored velocity $u_i^{*,r}$ (Eq. (20)) should be the same as the directly-advected velocity u_i^* (Eq. (19a)), as shown in Eq. (22), requiring the equivalence of a_1 and a_2. The constant density leads to the same term of $\frac{\partial u_x^n u_x^n}{\partial x}$ on both sides of Eq. (22), thus applying the same advection scheme on u and M meets the requirement.

$$u_i^{*,r} = u_x^* \quad \Rightarrow \quad u_x^n - (1+a_2)\,\Delta t \frac{\partial u_x^n u_x^n}{\partial x} = u_x^n - (1+a_1)\Delta t \frac{\partial u_x^n u_x^n}{\partial x} \tag{22}$$

Now consider the case of uniform flow of a fluid with density gradients, Fig. 7(b). All the properties are transported by u_0, and the restored velocity $u_i^{*,r}$ should stay at u_0, as shown in Eq. (23), requiring the equivalence of a_2 and a_3. The constant velocity leads to the same term of $\frac{\partial \rho_x^n}{\partial x}$ on both sides of Eq. (23), thus applying the same advection scheme on M and ρ meets the requirement.

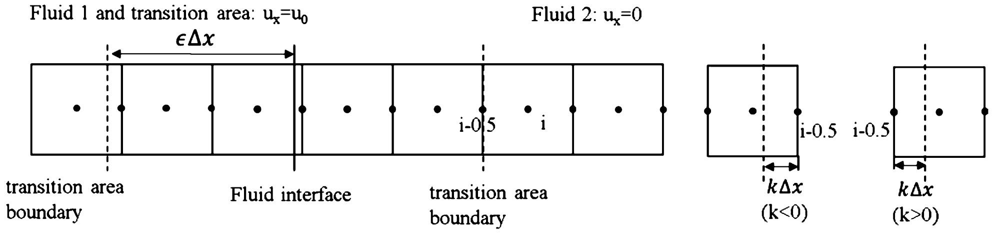

Fig. 8. The initial settings of the advection case.

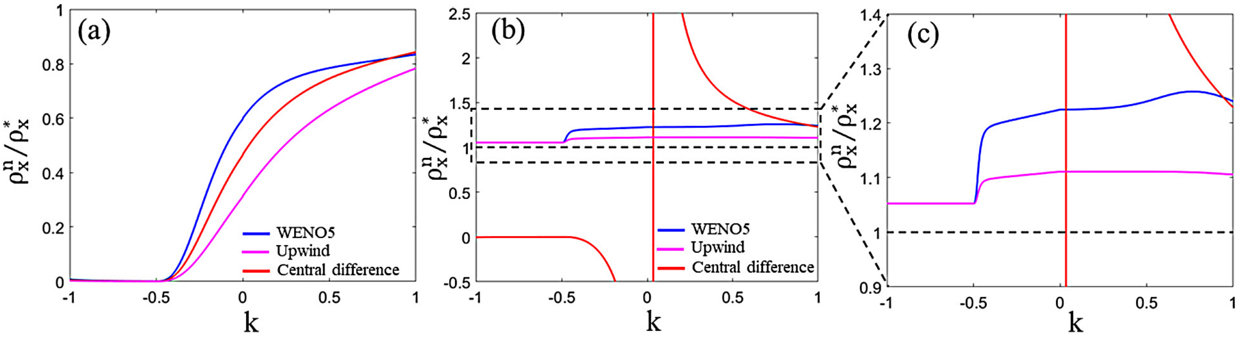

Fig. 9. ρ^n/ρ^* at different initial k; (a) $u_0>0$; (b) and (c) $u_0<0$.

$$u_i^{*,r}=u_0 \quad\Rightarrow\quad \frac{\rho_x^n u_0-(1+a_2)\,u_0\cdot u_0\Delta t\frac{\partial\rho_x^n}{\partial x}}{\rho_x^n-(1+a_3)u_0\Delta t\frac{\partial\rho_x^n}{\partial x}}=u_0 \quad\Rightarrow\quad \frac{\rho_x^n-(1+a_2)\,u_0\Delta t\frac{\partial\rho_x^n}{\partial x}}{\rho_x^n-(1+a_3)u_0\Delta t\frac{\partial\rho_x^n}{\partial x}}=1 \tag{23}$$

Starting from these two simple cases, there are two basic requirements: $a_1=a_2$ (same advection scheme on u and M) and $a_2=a_3$ (same advection scheme on ρ and M). Ideally, we would seek an advection scheme for which $a_1=a_2=a_3$. In general, this is not possible as this would impose an unphysical restraint on the gradients of the advected quantities. Nevertheless, applying the same scheme on the advection of u, M and ρ may still help to reduce the interaction between the truncation errors and its necessity is discussed in Section 5.1.

4.4. The 'limiter' treatment around the interfacial region

While density changes smoothly in the interfacial region, unbounded density may develop due to the numerical overshoot or undershoot in the density advection, which strongly influences the velocity restoration. This topic has been discussed by [52,61].

The following advection case in Fig. 8 gives a brief illustration. Fluid 1 with high density ($\rho_1/\rho_2=10^6$) is initialized by a velocity u_0 and the light fluid 2 is at rest. Viscosity and gravity are ignored. Taking $\epsilon=2.1$ and CFL=0.1, the boundary of the interfacial region is set to be $k\Delta x$ away from point (i-0.5). Based on Eq. (19) and Eq. (20), the restored velocity at point (i-0.5) is shown in Eq. (24). Compared with Eq. (25), the ratio between the original and advected density (ρ^n/ρ^*) is the key contribution to the velocity reconstruction. Should ρ^* become small, zero or even negative, this is likely to cause instability without further intervention.

$$u^{*,r}\approx\frac{u^n\rho_x^n-\Delta t\left(u^n\frac{\partial(u^n\rho_x^n)}{\partial x}+\rho_x^n\frac{\partial(u^nu^n)}{\partial x}\right)}{\rho_x^n-\Delta t\frac{\partial u_x^n\rho_x^n}{\partial x}}=u^n-\Delta t(\frac{\partial u^n\cdot u^n}{\partial x})\cdot\frac{\rho_x^n}{\rho_x^*} \tag{24}$$

$$u^*=u^n-\Delta t(\frac{\partial u^n\cdot u^n}{\partial x}) \tag{25}$$

ρ^n/ρ^* is lower than unity at a positive u_0, reducing the influence of the velocity gradient in the interfacial region (see Eq. (24)), and reduces the unphysical effect of the velocity of fluid 2 on denser fluid 1. In contrast, a negative u_0 may strengthen the effect of fluid 1 on the less dense fluid 2. Fig. 9 shows the results of ρ^n/ρ^* at point (i-0.5) calculated by 3 commonly used schemes (WENO5, central difference and upwind). As shown in Fig. 9(b), with large density gradient and low local density, the density advection of Eq. (19c) may lead to an unbounded density at a negative u_0, as ρ^* might be lower than the less dense fluid (or even negative, as observed in [57]), resulting in an unphysical value of ρ^n/ρ^*, which is especially noticeable with the central difference scheme. This numerical overshoot or undershoot is also observed in the advection of the volume fraction in VOF method which relies on the same advection equation as the volume fraction may be lower than 0 or higher than unity [62,63].

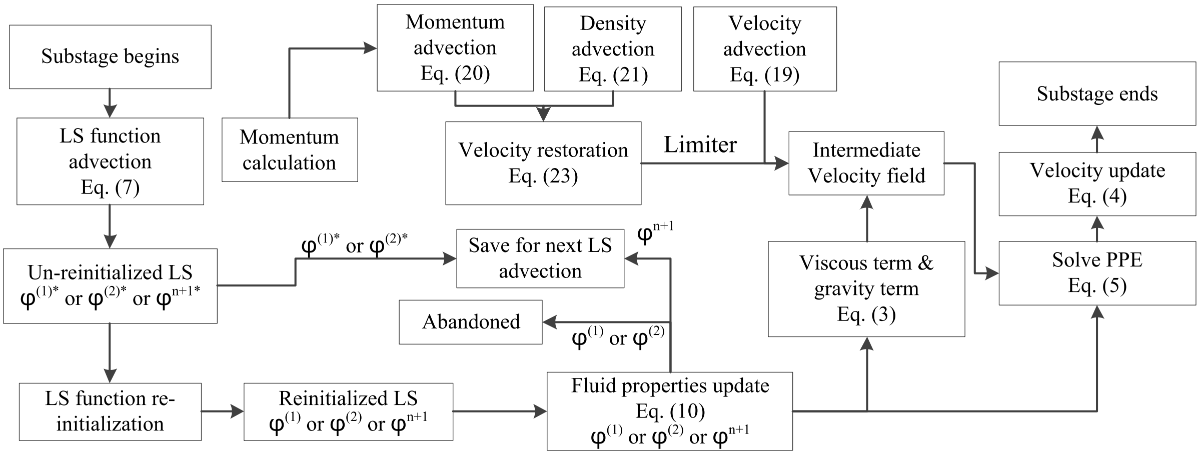
Substage begins
LS function advection Eq. (7)
Momentum calculation
Momentum advection Eq. (20)
Density advection Eq. (21)
Velocity advection Eq. (19)
Velocity restoration Eq. (23)
Limiter
Intermediate Velocity field
Substage ends
Velocity update Eq. (4)
Solve PPE Eq. (5)
Un-reinitialized LS φ^(1)* or φ^(2)* or φ^n+1*
φ^(1)* or φ^(2)*
Save for next LS advection
φ^n+1
Abandoned
φ^(1) or φ^(2)
Viscous term & gravity term Eq. (3)
LS function re-initialization
Reinitialized LS φ^(1) or φ^(2) or φ^n+1
Fluid properties update Eq. (10) φ^(1) or φ^(2) or φ^n+1

Fig. 10. The computational procedure of the proposed algorithm in one substage of RK3-TVD scheme.

As mentioned in [61], the density flux must meet the critical requirement of boundness. A limiter treatment is then applied on the nodes adjacent to the boundary with density close to the lighter fluid. Thus, the velocity restoration follows Eq. (26) instead of Eq. (20), as shown below:

$$\begin{cases} u_i^{*,r} = \frac{M_i^*}{\rho_i^*}, \rho_i^* \geq \rho_{threshold} \\ u_i^{*,r} = \frac{M_i^*}{\rho_i^{*,f}} * \frac{\rho_i^{*,f}}{\rho_{threshold}} + u_i^* * \frac{\rho_{threshold} - \rho_i^{*,f}}{\rho_{threshold}}, \rho_{air} < \rho_i^* < \rho_{threshold} and \rho_i^* < \rho_i^n \\ u_i^{*,r} = u_i^*, other \end{cases} \tag{26}$$

in which $\rho_i^{*,f}$ is a filtered density value following Eq. (27):

$$\begin{cases} \rho_i^{*,f} = \rho_{air}, \rho_i^* < \rho_{air} \\ \rho_i^{*,f} = \rho_i^*, \rho_i^* > \rho_{air} \end{cases} \tag{27}$$

For numerical stability, the 'threshold density' in Eq. (26) is set to be $0.05\rho_1 + \rho_2$, and a slightly larger threshold density is suggested when applying the central difference scheme due to the stronger numerical overshoot and undershoot.

4.5. The proposed algorithm

Combining the strong time-coupling (Section 4.1), velocity reconstruction (Section 4.2) and the limiter treatment (Section 4.4), the flow chart of the proposed algorithm which enhances the momentum conservation is shown in Fig. 10. As the proposed algorithm focusses on momentum conservation, we have termed it the 'Enhanced Momentum Conservation' (EMC) treatment. For simplicity, the FDM code with the implementation of EMC treatment is termed 'FDM-EMC'.

It is noted the proposed algorithm does not include any modification of existing schemes or methods. As explained in the Introduction, the aim of this study is to establish a proper computational procedure that improves the accuracy and stability with the least requirement for modification of numerical methods. The strong time coupling (Section 4.1) is a variation of the well-established RK3-TVD scheme, and the velocity reconstruction (Section 4.2) utilizes the existing spatial and temporal schemes to perform the advection of momentum, density and velocity instead of only velocity. The effectiveness of these two techniques is tested separately in Section 5.1.

Although not aiming to modify the surface-capturing method, the mass-loss problem of the classical LS method is still partly countered by the EMC treatment, which is shown in Section 5. As all the schemes and methods used (e.g., RK3-TVD, WENO5, classical LS and its re-initialization, projection method) remain the same, the EMC treatment retains the simplicity of these methods, and it can be implemented in a relatively straightforward manner. The EMC approach should also work with many existing variants, like the conservative LS method [48] or VOF method.

5. Numerical results

Two idealised numerical cases and one laboratory dambreak case are presented to investigate the performance of EMC treatment. In Section 5.1, a 2D pure advection case with large density ratio is chosen to detect the interfacial instability and study the conservation of mass, momentum and energy. In Section 5.2, an inviscid standing wave case is used to investigate the conservation in the absence of intensive interface deformation. In Section 5.3, an experimental dambreak case with a dry bed is chosen to demonstrate the advantage of EMC on realistic cases.

Table 2
Grid settings in case 1.

Grid settings	Grid resolution	Grid size	ϵ of LS	Density ratio R_ρ
G1	D=30 Δx	0.01 m	2.1	$R_\rho = 10^6$
G2	D=60 Δx	0.005 m	2.1	$R_\rho = 10^6$
G3	D=60 Δx	0.005 m	1.05	$R_\rho = 10^6$
G4	D=120 Δx	0.0025 m	2.1	$R_\rho = 10^6$
G5	D=240 Δx	0.00125 m	2.1	$R_\rho = 10^6$

Table 3
Code settings for the sensitivity studies 1 to 3.

Code settings	Modification applied	Advection scheme for ρ	Advection scheme for M and u	Density ratio R_ρ
S0*	N/A	N/A	WENO5	$R_\rho = 10^6$
S1	EMC	WENO5	WENO5	$R_\rho = 10^6$
S2	VR* only	WENO5	WENO5	$R_\rho = 10^6$
S3	TC* only	WENO5	WENO5	$R_\rho = 10^6$
S4	EMC	WENO5-HJ	WENO5	$R_\rho = 10^3$
S5	EMC	WENO5	WENO5-HJ	$R_\rho = 10^3$

*VR – Velocity Reconstruction; TC – Temporal Coupling; S0 is the original FDM code (REEF3D) without modification.

Table 4
Code settings for sensitivity study 4.

Code settings	Modification applied	Convection scheme for ρ	Convection scheme for M and u	Density ratio R_ρ
S0C1	N/A	N/A	Central difference	$R_\rho = 10^3$
S1C1	EMC	Central difference	Central difference	$R_\rho = 10^3$
S0C2	N/A	N/A	Upwind	$R_\rho = 10^6$
S1C2	EMC	Upwind	Upwind	$R_\rho = 10^6$
S0C3	N/A	N/A	QUICK	$R_\rho = 10^3$
S1C3	EMC	QUICK	QUICK	$R_\rho = 10^3$
S0C4	N/A	N/A	WENO5-HJ	$R_\rho = 10^6$
S1C4	EMC	WENO5-HJ	WENO5-HJ	$R_\rho = 10^6$

5.1. *Advection of a dense inviscid droplet*

The advection of a dense droplet has been investigated by various researchers [17,57,64], as a large density ratio makes it easier to detect unphysical momentum transport and spurious current across the interface. In this case, a droplet with diameter D=0.3 m is initialized with a horizontal speed u_0 (D/u_0=0.3 s) while the surrounding air is at rest. The density ratio (R_ρ) between the droplet and air is 10^6 (10^3 if the simulation crashes too early). Viscosity, surface tension and gravity are ignored and open boundary conditions are applied at each side of the domain. The CFL number is set to be 0.1. Due to the extreme density ratio, it is supposed that the influence of the air on the droplet is negligible, and the droplet travels with a constant speed (u_0) without shape change.

As shown in Tables 2 to 4, 5 uniform grid systems (G1 to G5) and different code settings are used for these sensitivity studies: 1) grid convergence; 2) the individual contributions of the strong time-coupling (Section 4.1) and the velocity reconstruction (Section 4.2); 3) the necessity of applying the same spatial scheme on density and momentum (Section 4.3); 4) the model performance under different advection schemes.

Fig. 11 shows the results of the grid convergence study of code settings S0 and S1 under grids G1 to G5. Due to the non-conservation of momentum in the velocity advection (Section 3.1) and the temporal mismatch (Section 3.2), no obvious grid convergence is observed in the results of S0, which also loses the smooth disk shape of the droplet. The stability and interface smoothness are downgraded under a thinner interface thickness (G3) or a finer grid (G4, G5), which agrees with the analysis in Section 3.3. However, lower Δx or ϵ are allowed in the simulation with EMC treatment, as S1 achieves grid convergence and retains the smooth disk shape of the droplet, while its location also agrees with the reference one. It is noted the interface deformation is visible at a resolution close to G4 in [17], while S1 allows a resolution of 1 time finer (G5).

Fig. 12 shows the evolution of the volume, horizontal momentum and kinematic energy of the droplet, presenting a quantitative comparison between the results of S0 and S1. Suffering from the instability explained in Section 3 and the well-known mass loss problem of LS method [49], the time series of mass, volume and momentum show obvious oscillations and a decreasing trend in the results for S0 in Fig. 12. Although there is no modification to the classical LS method, the implementation of EMC shows a significant improvement in its conservation properties, especially under the finer grids (G4, G5), as the loss of all 3 variables is significantly reduced, (compare the solid and dashed line with the same colour in Fig. 12(a) to (c)). As an approximate means of filtering out the influence of mass loss on energy loss and study the energy change from the interfacial transport, we compare the ratio between energy and volume in Fig. 12(d), from which

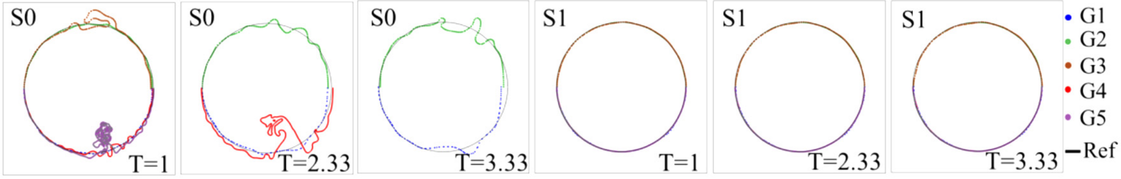

Fig. 11. Grid convergence study for S0 and S1 (time t is normalized by T=t*u0/D). Solutions are symmetric about the 'equator'. Results from each grid are shown for a hemisphere. See text for further explanation.

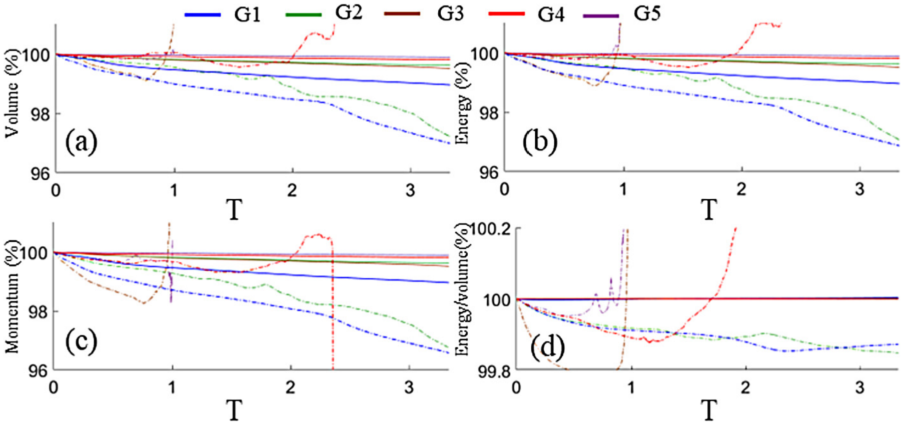

Fig. 12. Comparison of the results of G1 to G5 on the evolution of (a) the total volume; (b) the total energy; (c) the total momentum of the droplet towards the x direction; (d) the energy per unit volume. (Dashed line – S0; Solid line – S1).

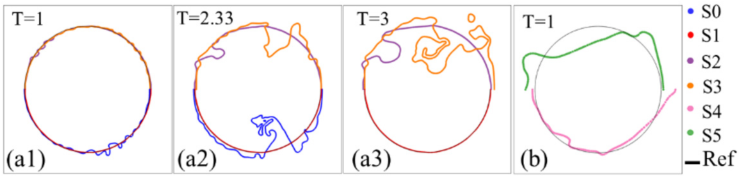

Fig. 13. (a) comparison of the contributions of the modifications (b) results of EMC treatment when applying different spatial scheme for ρ, M and u.

it is concluded the energy loss is mainly from the volume loss, as the ratio is always close to 100% with both S0 and S1. However, the superiority of S1 is still confirmed by its more stable and converged time series and the negligible energy change per unit volume.

To identify the individual contributions of the strong time-coupling (Section 4.1) and velocity reconstruction (Section 4.2), the results of S0 (original FDM code) S1 (EMC fully implemented), S2 (only strong time-coupling implemented) and S3 (only velocity reconstruction implemented) under grid G4 are compared in Fig. 13(a) and Fig. 14. As shown in Fig. 13(a), the smooth disk shape only retains in the result of S1 while slight interface twisting is already observed at an early stage (T=1) in S0 (most obvious), S2 and S3. At T=2.33, the disk shape is distorted in S3 and S0 (slightly more intense) while S2 roughly keeps the disk shape despite the deformation at the rear side of the droplet. Runs S0 and S3 crash at T=2.50 and T=3.03, respectively. At T=3, significant interface shape deformation is eventually observed in S2. Fig. 14 compares the time series of volume, momentum, energy and energy per unit volume of the four code settings, in which S0 is the most unstable. All S1, S2 and S3 show a decreasing trend on volume, momentum and energy, while S1 shows the best performance (reduction is less than 0.25% at T=3.3) and S2 outperforms S3. It is noted the reason of the rising trend in Fig. 14(d) is that the energy loss is slower than the volume loss. Compared with S0, both S2 and S3 show improvement on interface smoothness, numerical stability and property conservation, and the performance of S3 is significantly better, as the velocity reconstruction directly focuses on the momentum conservation and the aim of strong time coupling is the synchronization of variables. However, satisfactory disk shape is only observed in S1 in which the time coupling and velocity reconstruction are both implemented. Thus, it is concluded that the main contribution of EMC treatment is the velocity reconstruction, but the strong time-coupling is also essential.

To prove the necessity to use the same advection scheme for both density and momentum, the results of S4 and S5 are compared in Fig. 13(b). Although both WENO5 and the WENO5 for Hamilton-Jacobi equation (WENO5-HJ, which is more suitable for Hamilton-Jacobi equations [65] and frequently used for LS advection, but less conservative than normal WENO5) can achieve 5th order accuracy, S4 and S5 crash immediately under a density ratio of 10^6. As shown in Fig. 13(b), in the

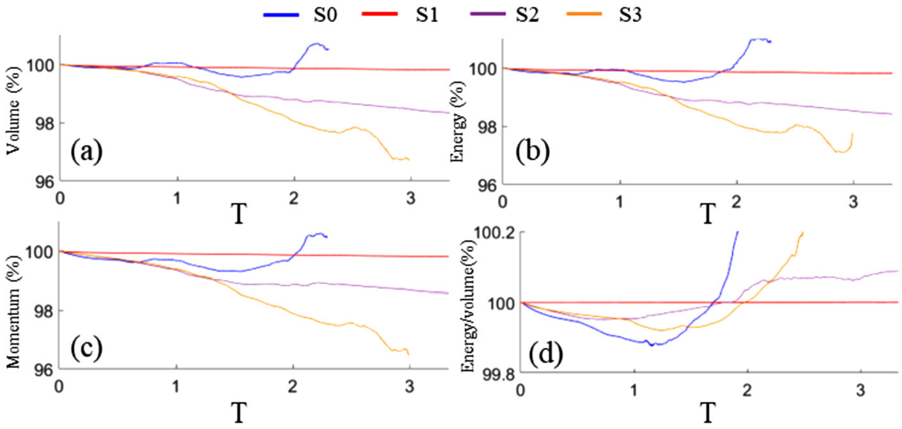

Fig. 14. Comparison of the results of S0 to S3 on the evolution of (a) the total volume; (b) the total energy; (c) the total momentum of the droplet towards the x direction; (d) the energy per unit volume.

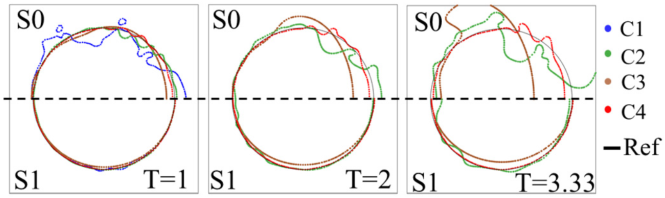

Fig. 15. Comparison between S0 and S1 under different spatial schemes (S0C1 and S1C1 both crashes before T=2).

simulation with density ratio of 10^3, both of them failed to keep the disk shape of the droplet even at T=1, indicating the virtue of using the same spatial scheme for the advection of ρ, M and u. The same conclusion is obtained from different codes, for example [64], in which the distorted interface and larger velocity errors are detected when dissimilar convection schemes for mass and momentum are applied.

To test the performance of the EMC treatment under different spatial schemes, simulations under grid G2 are performed with different convection schemes shown in Table 4, and the interface shapes are compared in Fig. 15. With lower order schemes, S0 quickly loses the disk shape, and instability can still be observed with a high order scheme like WENO5-HJ. In comparison, with all schemes tested, the EMC treatment significantly improves the stability and smoothness of the interface although the performance is much better with higher order schemes.

5.2. Standing wave case

A standing wave case is designed to test the effectiveness of EMC in a case with more gentle surface variation. As shown in Fig. 16(a), the computational area is 5.5 m long with still water level 0.5 m. A sinusoidal wave with wave height 0.01 m and 2 m wavelength (period 1.182 s) is generated by a relaxation area of 2 m length. The wave propagates towards right and hits a perfectly reflecting boundary, the superposition of the incident and reflected waves forms a standing wave in an area of 3.5 m length. The reflected wave propagates out of the active area and is absorbed by the relaxation area [66,67]. The densities of water and air are set to 1000 kg/m^3 and 1 kg/m^3 respectively, and g is set to 9.81 m/s^2 and the fluids are considered inviscid. A wall boundary condition is applied to the top and bottom boundaries and a reflection boundary condition is applied at the right wall. WENO5 and RK3-TVD schemes are used for the advection of variables and the CFL number is 0.1. The interfacial thickness parameter of LS (ϵ) is set to 2.1. Three uniform grid systems are applied, namely a coarse grid (Δx=0.02 m), a medium grid (Δx=0.01 m) and a fine grid (Δx=0.005 m). The simulation is performed for 120 s with both the original FDM-based code and FDM-EMC. The simulated wave profiles of the standing wave between 57.3 s and 57.9 s are shown in Fig. 16(b).

Fig. 17 and 18 compares the vertical momentum of water and air in the last 20 seconds. FDM-EMC shows a good grid convergence, and the momentum of water oscillates around zero with a stable amplitude, either water or air (Fig. 17(a) and 18(a)). However, FDM leads to a downgraded grid convergence, and a less stable momentum oscillation, especially with coarser grids, indicating the unphysical momentum change across the water-air interface (Fig. 17(b) and 18(b)), which is well suppressed by FDM-EMC.

The time series of the total energy of water in the last 20 seconds are compared in Fig. 19 and the average volume of water of the last 10 wave periods is shown in Table 5. Although there is no intensive water-air interaction and visible interface deformation under this low wave height (0.02 m in 0.5 m water depth), FDM-EMC still provides a more stable

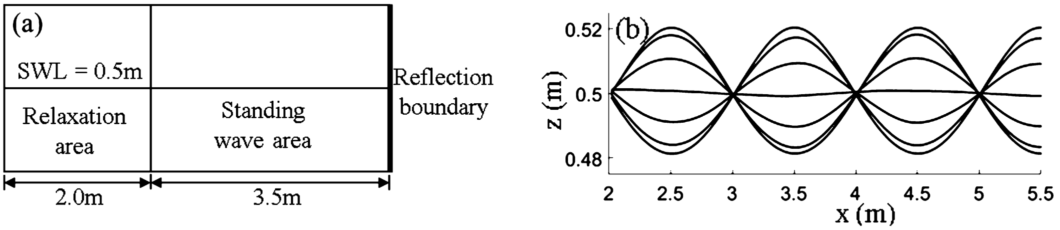

Fig. 16. (a) the settings of the standing wave case (not in scale); (b) the profiles of the simulated incident wave and standing wave.

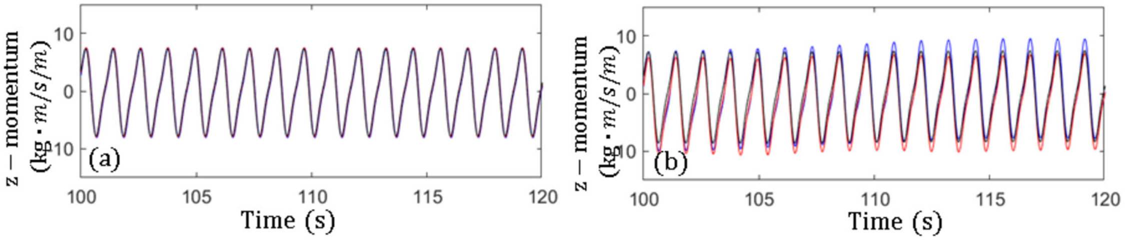

Fig. 17. The vertical momentum of the water in the active area (a) FDM - EMC (b) original FDM code. (blue – coarse grid, red – medium grid, black – fine grid).

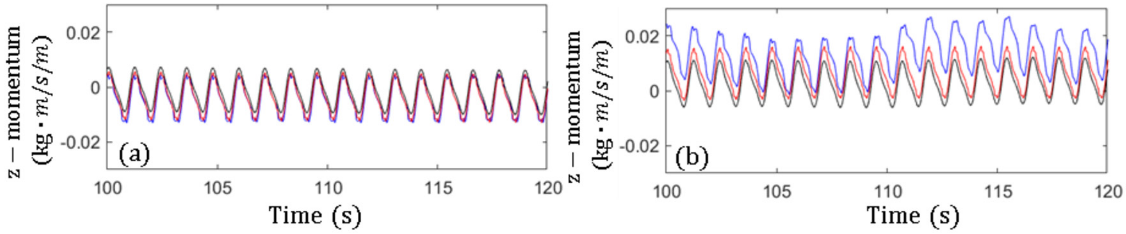

Fig. 18. The vertical momentum of the air in the active area (a) FDM - EMC (b) original FDM code. (blue – coarse grid, red – medium grid, black – fine grid).

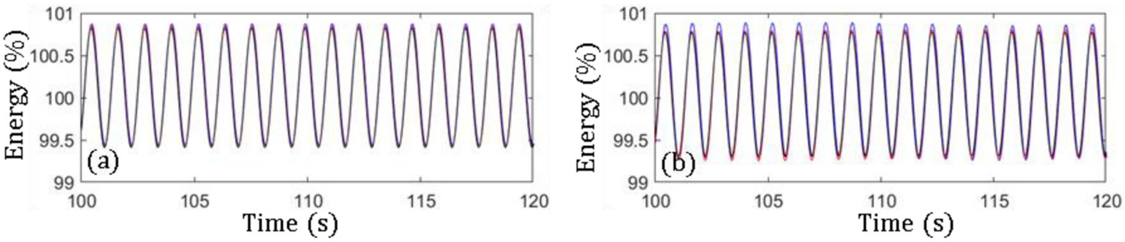

Fig. 19. The total energy of the water (a) FDM with EMC (b) original FDM code. (blue – coarse grid, red – medium grid, black – fine grid; energy is normalized by the initial potential energy of still water).

Table 5
The comparison of volume loss (unit: %).

Grid size	0.02	0.01	0.005
FDM-EMC	0.184	0.086	0.035
FDM	0.326	0.15	0.089

oscillation and a better convergence for the total energy. Both codes produce a minimum volume change (no more than 0.33%) in this case, but FDM-EMC still slightly outperforms the original FDM because of the reduced volume loss.

5.3. Dambreak case with a dry bed

The third benchmark case is a lab-scale dambreak case with a dry bed with settings and parameters shown in Fig. 20 and more information can be found in [68]. Three uniform grid systems are applied to the simulation, namely coarse

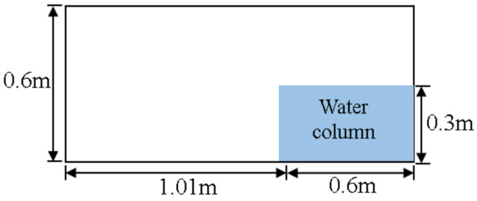

Fig. 20. Settings of the dambreak case with a dry bed.

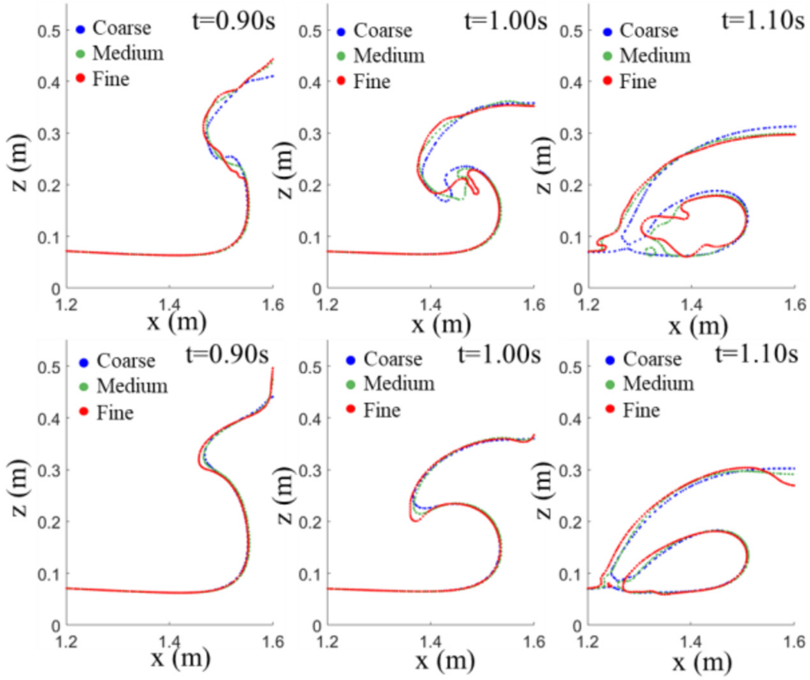

Fig. 21. Grid convergence study (First row: FDM; Second row: FDM-EMC).

($\Delta x = 0.01m$), medium ($\Delta x = 0.0075m$) and fine grid ($\Delta x = 0.005m$), and the interfacial region parameter ϵ is set to be 2.1 [45]. A wall boundary condition is applied on the left, right and bottom sides while an open boundary condition is applied to the top boundary.

The grid convergence studies of the FDM and FDM-EMC (S0 and S1 in Table 3) are compared in Fig. 21. It is seen FDM-EMC better keeps the smoothness of the water-air interface and produces a sharper overturning tongue than FDM. A better grid convergence is also found in the results of FDM-EMC as the water-air interfaces of different grid resolutions overlap together while larger differences are found in the results of FDM.

Fig. 22 compares the surface profile simulated by FDM-EMC and several existing simulations [70,69,71] with the experimental snapshots [68]. As shown in the top row of Fig. 22, the simulation results all agree well with the experiment before the collision between wavefront and left wall. The intense water-air interaction leads to the twisted water-air interface since t=573.3 ms in VOF simulation [69] while other codes still agree well with the experiment, as seen in the middle row of Fig. 22. For the fully developed breaker (t=1023.3 ms), FDM-EMC generates a much sharper tongue and lower splash than VOF [71] and SPH [70]. At 1166.6 ms, FDM-EMC and compressive interface tracking method [71] generates a highly similar entrapped air pocket, while the secondary jet from FDM-EMC is closer to the experiment.

The pressure distributions, water-air interface shapes and velocity vector maps reproduced by FDM-EMC and FDM are compared in Fig. 23. A twisted air-water interface is already observed in FDM from t=0.30 s as negative pressure areas and vortices develop at the front border which propagate in a retrograde manner along the interface. Numerical instability is visible on the interface shape at t=1.0 s which is characterised by unphysical negative pressure and water 'branching' around the breaking tongue. The instability at t=1.16 s causes the simulation to terminate, as the local vortex and negative pressure (as low as -2000 Pa) dominates the flow field around the secondary jet. However, FDM-EMC effectively suppresses the negative pressure and vortex around the water-air interface, as the water movement is less influenced by the air due to its improved momentum conservation.

The collision between the overturning breaker and underlying water body proves the necessity of the limiter, as shown in Fig. 24. At t=1.06 s, high gradients of density and velocity develop between the two separate waterbodies and the

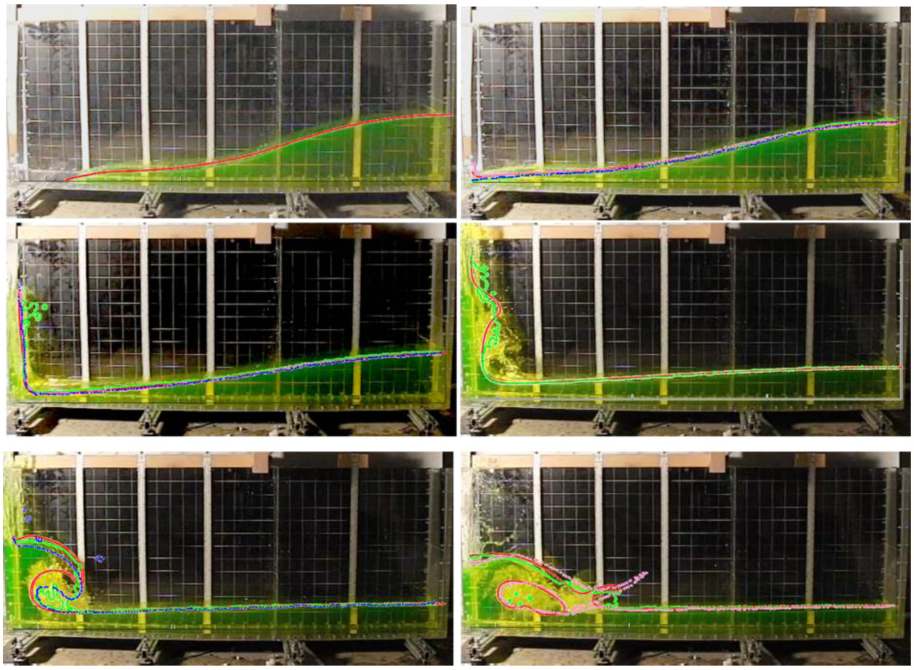

Fig. 22. The comparison of the simulation results of FDM-EMC and several different methods and the experiment snapshot from [68] (time = 373.3, 449.9, 573.3, 862.3, 1023.3, 1166.6 ms; Red – FDM-EMC; Blue – a stable SPH method in [70]; Green – VOF simulation of [69]; Pink – a compressive interface capturing method in [71]).

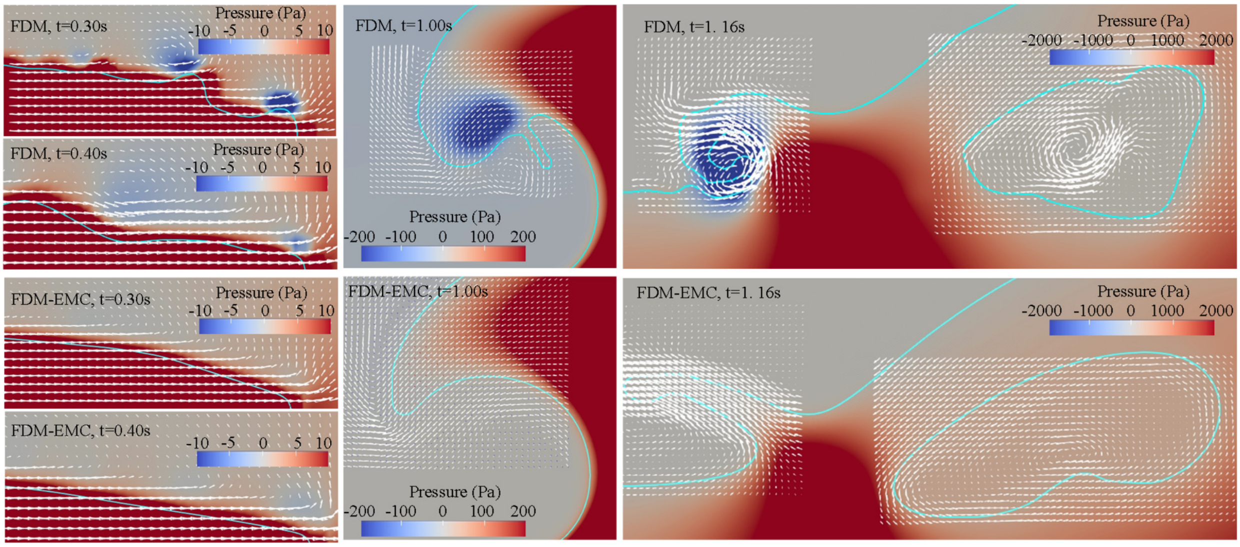

Fig. 23. Comparison of the pressure distribution, interface shape and velocity vector around the leading front (white line is the water-air interface).

surrounding air. According to Eq. (24), the unphysical density ratio leads to an incorrect velocity field and pressure field, culminating in the termination of the simulation without the limiter before the water-air interfaces merge.

5.4. CPU performance

As described in Section 4, the EMC treatment uses the strong time-coupling and momentum-based velocity reconstruction to enhance the conservation. Compared with the original code, additional advection of momentum and density are performed, leading to the reduction of numerical efficiency. The CPU time per loop of FDM and FDM-EMC are compared in Table 6. It is seen the CPU time rises about 50% at a lower grid resolution and 100% at a higher resolution. Future work would be conducted to improve the computational efficiency.

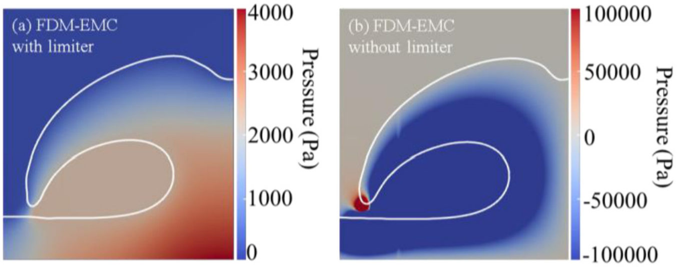

Fig. 24. The interface shapes and pressure contours of FDM-EMC with and without limiter at t=1.06s.

Table 6
The comparison of CPU time of the code with and without EMC treatment (All simulations are based on 8 threads).

	Droplet G5	Droplet G4	Droplet G2	Droplet G1	Dambreak coarse grid	Dambreak medium grid	Dambreak fine grid
Cell amount	640000	160000	40000	10000	9660	17200	38640
Time per loop - S0	1.903	0.503	0.155	0.053	0.027	0.050	0.110
Time per loop - S1	3.784	0.862	0.212	0.053	0.046	0.091	0.176

6. Conclusion

Simulation of two-phase flow suffers from spurious interfacial currents and instability. The associated lack of conservation of momentum arises chiefly from inaccuracies in velocity advection and synchronization of the variables. Grid refinement and interface sharpening may also amplify the interfacial instability. To deal with these problems, the EMC treatment is proposed, which consists of the strong time-coupling between LS and NSE, the momentum-based velocity reconstruction and the limiter treatment around the interfacial region. The EMC treatment has been implemented into an existing FDM code and significantly improves its performance. The simulation of the convection of a dense inviscid droplet demonstrates the improvement on the accuracy, stability and the conservation of mass, momentum and energy, and the capability of dealing with two phase flows with an extreme density ratio. Through the comparison study, it is concluded that the momentum-based velocity reconstruction contributes most to the EMC treatment, but strong time-coupling is also essential. A standing wave case confirms that the EMC treatment can provide improvements even under milder conditions without large interface deformation. In the dambreak case, the shape of the breaking wave is well reproduced, and the interfacial smoothness is also improved considerably. The distribution of pressure and velocity around the water-air interface has been analysed in detail, confirming the advantages of the EMC approach.

The EMC treatment requires neither modifications to the underlying numerical scheme or method, nor special treatments or additional boundary conditions applied to the interface. It therefore provides an additional way to improve the conservation of mass, momentum and energy as well as the smoothness of the fluid interface. Furthermore, the EMC treatment can work with existing algorithms and surface tracking methods, and it is straightforward to implement because of its simplicity.

CRediT authorship contribution statement

Xin Wang: Formal analysis, Investigation, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Min Luo:** Conceptualization, Formal analysis, Funding acquisition, Writing – original draft, Writing – review & editing. **Harshinie Karunarathna:** Investigation, Validation, Writing – original draft, Writing – review & editing. **Dominic E. Reeve:** Conceptualization, Formal analysis, Project administration, Resources, Validation, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

The authors would like to acknowledge the support Xin Wang received through a Swansea College of Engineering PhD Scholarship and useful discussions with Dr Jose Horrillo-Caraballo. Dr Min Luo also acknowledges the support of the Science Foundation of Donghai Laboratory (No. DH-2022KF0311).

References

[1] J. Xu, S. Zhan, W. Wang, Y. Su, H. Wang, Molecular dynamics simulations of two-phase flow of n-alkanes with water in quartz nanopores, Chem. Eng. J. 430 (2022) 132800.

[2] P.N. Sun, M. Luo, D. Le Touzé, A.M. Zhang, The suction effect during freak wave slamming on a fixed platform deck: smoothed particle hydrodynamics simulation and experimental study, Phys. Fluids 31 (11) (2019) 117108.

[3] M. Luo, A. Khayyer, P. Lin, Particle methods in ocean and coastal engineering, Appl. Ocean Res. 114 (2021) 102734.

[4] O. Souček, K. Kalousová, O. Čadek, Water transport in planetary ice shells by two-phase flow–a parametric study, Geophys. Astrophys. Fluid Dyn. 108 (6) (2014) 639–666.

[5] Z. Wang, Q. Zou, D. Reeve, Simulation of spilling breaking waves using a two phase flow CFD model, Comput. Fluids 38 (10) (2009) 1995–2005.

[6] M. Ishii, Thermo-fluid dynamic theory of two-phase flow, NASA Sti/recon Tech. Rep. A 75 (1975) 29657.

[7] M. Manninen, V. Taivassalo, S. Kallio, On the mixture model for multiphase flow, 1996, Finland: N. p., Web.

[8] M. Baer, J. Nunziato, A two-phase mixture theory for the deflagration-to-detonation transition (DDT) in reactive granular materials, Int. J. Multiph. Flow 12 (1986) 861–889.

[9] H. Lochon, F. Daude, P. Galon, J.M. Hérard, HLLC-type Riemann solver with approximated two-phase contact for the computation of the Baer–Nunziato two-fluid model, J. Comput. Phys. 326 (2016) 733–762.

[10] R. Saurel, R. Abgrall, A multiphase Godunov method for compressible multifluid and multiphase flows, J. Comput. Phys. 150 (1999) 425–467.

[11] D.W. Schwendeman, C.W. Wahle, A.K. Kapila, The Riemann problem and a high-resolution Godunov method for a model of compressible two-phase flow, J. Comput. Phys. 212 (2006) 490–526.

[12] V. Deledicque, M.V. Papalexandris, An exact Riemann solver for compressible two-phase flow models containing non-conservative products, J. Comput. Phys. 222 (2007) 217–245.

[13] J.M. Ghidaglia, A. Kumbaro, G. Le Coq, On the numerical solution to two fluid models via a cell centered finite volume method, Eur. J. Mech. B, Fluids 20 (6) (2001) 841–867.

[14] S. Dykas, W. Wróblewski, Single-and two-fluid models for steam condensing flow modeling, Int. J. Multiph. Flow 37 (9) (2011) 1245–1253.

[15] G. Cerne, S. Petelin, I. Tiselj, Coupling of the interface tracking and the two-fluid models for the simulation of incompressible two-phase flow, J. Comput. Phys. 171 (2) (2001) 776–804.

[16] A. Prosperetti, G. Tryggvason, Computational Methods for Multiphase Flow, Cambridge University Press, 2009.

[17] F. Desmons, M. Coquerelle, A generalized high-order momentum preserving (HOMP) method in the one-fluid model for incompressible two-phase flows with large density ratio, J. Comput. Phys. 437 (2021) 110322.

[18] G. Tryggvason, R. Scardovelli, S. Zaleski, Direct Numerical Simulations of Gas–Liquid Multiphase Flows, Cambridge University Press, 2011.

[19] S. McKee, M.F. Tomé, V.G. Ferreira, J.A. Cuminato, A. Castelo, F.S. Sousa, N. Mangiavacchi, The MAC method, Comput. Fluids 37 (8) (2008) 907–930.

[20] C.W. Hirt, B.D. Nichols, Volume of fluid (VOF) method for the dynamics of free boundaries, J. Comput. Phys. 39 (1) (1981) 201–225.

[21] S. Osher, J.A. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations, J. Comput. Phys. 79 (1) (1988) 12–49.

[22] M. Sussman, E.G. Puckett, A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows, J. Comput. Phys. 162 (2) (2000) 301–337.

[23] Z. Li, C. Liu, R. Gao, C. Hu, A consistent mass-momentum advection method for the simulation of large-density-ratio two-phase flows, Int. J. Multiph. Flow 156 (2022) 104192.

[24] G. Keshavarzi, G.H. Yeoh, T. Barber, Comparison of the VOF and CLSVOF methods in interface capturing of a rising bubble, J. Comput. Multiph. Flows 5 (1) (2013) 43–55.

[25] D.L. Sun, W.Q. Tao, A coupled volume-of-fluid and level set (VOSET) method for computing incompressible two-phase flows, Int. J. Heat Mass Transf. 53 (4) (2010) 645–655.

[26] J. Lopez, J. Hernandez, P. Gomez, F. Faura, An improved PLIC-VOF method for tracking thin fluid structures in incompressible two-phase flows, J. Comput. Phys. 208 (1) (2005) 51–74.

[27] J. López, J. Hernández, Analytical and geometrical tools for 3D volume of fluid methods in general grids, J. Comput. Phys. 227 (12) (2008) 5939–5948.

[28] A. Cervone, S. Manservisi, R. Scardovelli, S. Zaleski, A geometrical predictor–corrector advection scheme and its application to the volume fraction function, J. Comput. Phys. 228 (2) (2009) 406–419.

[29] M.R. Hashemi, P.B. Ryzhakov, R. Rossi, An enriched finite element/level-set method for simulating two-phase incompressible fluid flows with surface tension, Comput. Methods Appl. Mech. Eng. 370 (2020) 113277.

[30] M. Sussman, E. Fatemi, P. Smereka, S. Osher, An improved level set method for incompressible two-phase flows, Comput. Fluids 27 (5–6) (1998) 663–680.

[31] A.K. Tornberg, B. Engquist, A finite element based level-set method for multiphase flow applications, Comput. Vis. Sci. 3 (1) (2000) 93–101.

[32] M.R. Hashemi, P.B. Ryzhakov, R. Rossi, Three dimensional modeling of liquid droplet spreading on solid surface: an enriched finite element/level-set approach, J. Comput. Phys. 442 (2021) 110480.

[33] D. Enright, D. Nguyen, F. Gibou, R. Fedkiw, Using the particle level set method and a second order accurate pressure boundary condition for free surface flows, in: Fluids Engineering Division Summer Meeting, vol. 36975, January 2003, pp. 337–342.

[34] C. Min, F. Gibou, A second order accurate level set method on non-graded adaptive cartesian grids, J. Comput. Phys. 225 (1) (2007) 300–321.

[35] A.M. Moghadam, M. Shafieefar, R. Panahi, Development of a high-order level set method: compact conservative level set (CCLS), Comput. Fluids 129 (2016) 79–90.

[36] C. Liu, D. Ray, C. Thiele, L. Lin, B. Riviere, A pressure-correction and bound-preserving discretization of the phase-field method for variable density two-phase flows, J. Comput. Phys. 449 (2022) 110769.

[37] E. Aulisa, S. Manservisi, R. Scardovelli, S. Zaleski, Interface reconstruction with least-squares fit and split advection in three-dimensional cartesian geometry, J. Comput. Phys. 225 (2007) 2301–2319.

[38] S. Mirjalili, S.S. Jain, M. Dodd, Interface-Capturing Methods for Two-Phase Flows: An Overview and Recent Developments, Center for Turbulence Research Annual Research Briefs, 2017, pp. 117–135, 2017, 13.

[39] A. Christou, T. Stoesser, Z. Xie, A large-eddy-simulation-based numerical wave tank for three-dimensional wave-structure interaction, Comput. Fluids 231 (2021) 105179.

[40] V. Bogdanov, F.S. Schranner, J.M. Winter, S. Adami, N.A. Adams, A level-set-based sharp-interface method for moving contact lines, J. Comput. Phys. 467 (2022) 111445.

[41] D. Kuzmin, J.P. Bäcker, An unfitted finite element method using level set functions for extrapolation into deformable diffuse interfaces, J. Comput. Phys. 461 (2022) 111218.

[42] H. Bihs, A. Kamath, A combined level set/ghost cell immersed boundary representation for floating body simulations, Int. J. Numer. Methods Fluids 83 (12) (2017) 905–916.

[43] H. Bihs, M.A. Chella, A. Kamath, Ø.A. Arntsen, Numerical investigation of focused waves and their interaction with a vertical cylinder using REEF3D, J. Offshore Mech. Arct. Eng. 139 (4) (2017).

[44] M.A. Chella, H. Bihs, D. Myrhaug, M. Muskulus, Breaking solitary waves and breaking wave forces on a vertically mounted slender cylinder over an impermeable sloping seabed, J. Ocean Eng. Mar. Energy 3 (1) (2017) 1–19.

[45] H. Bihs, A. Kamath, M.A. Chella, A. Aggarwal, Ø.A. Arntsen, A new level set numerical wave tank with improved density interpolation for complex wave hydrodynamics, Comput. Fluids 140 (2016) 191–208.

[46] A.J. Chorin, Numerical solution of the Navier-Stokes equations, Math. Comput. 22 (104) (1968) 745–762.

[47] D. Peng, B. Merriman, S. Osher, H. Zhao, M. Kang, A PDE-based fast local level set method, J. Comput. Phys. 155 (2) (1999) 410–438.

[48] E. Olsson, G. Kreiss, A conservative level set method for two phase flow, J. Comput. Phys. 210 (1) (2005) 225–246.

[49] E. Olsson, G. Kreiss, S. Zahedi, A conservative level set method for two phase flow II, J. Comput. Phys. 225 (1) (2007) 785–807.

[50] D.J. Harvie, M.R. Davidson, M. Rudman, An analysis of parasitic current generation in volume of fluid simulations, Appl. Math. Model. 30 (10) (2006) 1056–1066.

[51] K. Connington, T. Lee, A review of spurious currents in the lattice Boltzmann method for multiphase flows, J. Mech. Sci. 26 (12) (2012) 3857–3863.

[52] D. Zuzio, A. Orazzo, J.L. Estivalèzes, I. Lagrange, A new efficient momentum preserving level-set/vof method for high density and momentum ratio incompressible two-phase flows, J. Comput. Phys. 410 (2020) 109342.

[53] S. Gottlieb, C.W. Shu, Total variation diminishing Runge-Kutta schemes, Math. Comput. 67 (221) (1998) 73–85.

[54] Z. Zheng, L. Petzold, Runge–Kutta–Chebyshev projection method, J. Comput. Phys. 219 (2) (2006) 976–991.

[55] Y.J. Kim, B. Bouscasse, S. Seng, D. Le Touzé, Efficiency of diagonally implicit Runge-Kutta time integration schemes in incompressible two-phase flow simulations, Comput. Phys. Commun. (2022) 108415.

[56] M. Herrmann, A balanced force refined level set grid method for two-phase flows on unstructured flow solver grids, J. Comput. Phys. 227 (4) (2008) 2674–2706.

[57] N. Nangia, B.E. Griffith, N.A. Patankar, A.P.S. Bhalla, A robust incompressible Navier-Stokes solver for large density ratio multiphase flows, J. Comput. Phys. 390 (2019) 548–594.

[58] M. Raessi, A Level Set Based Method for Calculating Flux Densities in Two-Phase Flows, Annual Research Briefs, Center for Turbulence Research, Stanford, 2008.

[59] M. Raessi, H. Pitsch, Modeling Interfacial Flows Characterized by Large Density Ratios with the Level Set Method, Center for Turbulence Research Annual Research Briefs, 2009, pp. 159–169.

[60] M. Raessi, H. Pitsch, Consistent mass and momentum transport for simulating incompressible interfacial flows with large density ratios using the level set method[J], Comput. Fluids 63 (2012) 70–81.

[61] O. Desjardins, V. Moureau, Methods for Multiphase Flows with Large Density Ratio. Center for Turbulent Research, Center for Turbulent Research, Summer Programm, 2010 2010, pp. 313–322.

[62] G.H. Miller, E.G. Puckett, A high-order Godunov method for multiple condensed phases, J. Comput. Phys. 128 (1) (1996) 134–164.

[63] L. Margolin, J.M. Reisner, P.K. Smolarkiewicz, Applications of the volume-of-fluid method to the advection–condensation problem, Mon. Weather Rev. 125 (1997) 2265–2273.

[64] J.K. Patel, G. Natarajan, A novel consistent and well-balanced algorithm for simulations of multiphase flows on unstructured grids, J. Comput. Phys. 350 (2017) 207–236.

[65] G.S. Jiang, D. Peng, Weighted ENO schemes for Hamilton–Jacobi equations, SIAM J. Sci. Comput. 21 (6) (2000) 2126–2143.

[66] N.G. Jacobsen, D.R. Fuhrman, J. Fredsøe, A wave generation toolbox for the open-source CFD library: OpenFoam®, Int. J. Numer. Methods Fluids 70 (9) (2012) 1073–1088.

[67] A.M. Miquel, A. Kamath, M. Alagan Chella, R. Archetti, H. Bihs, Analysis of different methods for wave generation and absorption in a CFD-based numerical wave tank, J. Mar. Sci. 6 (2) (2018) 73.

[68] L. Lobovský, E. Botia-Vera, F. Castellana, J. Mas-Soler, A. Souto-Iglesias, Experimental investigation of dynamic pressure loads during dam break, J. Fluids Struct. 48 (2014) 407–434.

[69] W.C. Moon, H.T. Puay, T.L. Lau, Robust and efficient 3-D numerical model for the hydrodynamic simulation of tsunami wave on land, Adv. Water Resour. 146 (2020) 103762.

[70] F. He, H. Zhang, C. Huang, M. Liu, A stable SPH model with large CFL numbers for multi-phase flows with large density ratios, J. Comput. Phys. 453 (2022) 110944.

[71] C.T. Ha, D.H. Kim, W.G. Park, C.M. Jung, A compressive interface-capturing scheme for computation of compressible multi-fluid flows, Comput. Fluids 152 (2017) 164–181.