Journal of Computational Physics 500 (2024) 112752

Contents lists available at ScienceDirect

Journal of Computational Physics

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

In situ adaptive tabulation of vapor-liquid equilibrium solutions for multi-component high-pressure transcritical flows with phase change

Hongyuan Zhang, Navneeth Srinivasan, Suo Yang *

Department of Mechanical Engineering, University of Minnesota – Twin Cities, 111 Church Street SE, Minneapolis, MN 55455, USA

ARTICLE INFO

Keywords:
Vapor-liquid equilibrium (VLE)
Computational fluid dynamics (CFD)
In situ adaptive tabulation (ISAT)
High-pressure transcritical flows
Phase change

ABSTRACT

Vapor-liquid equilibrium (VLE) is a family of first-principled thermodynamic models for transcritical multiphase flows, which can accurately capture the phase transitions at high-pressure conditions that are difficult to deal with using other models. However, VLE-based computational fluid dynamics (CFD) simulation is computationally very expensive for multi-component systems, which severely limits its applications to real-world systems. In this work, we developed a new ISAT-VLE method based on the *in situ* adaptive tabulation (ISAT) method to improve the computational efficiency of VLE-based CFD simulation with reduced memory usage. We developed several ISAT-VLE solvers for both fully conservative (FC) and double flux (DF) schemes. New methods are proposed to delete redundant records in the ISAT-VLE table and the ISAT-VLE method performance is further improved. To improve the convergence of the VLE solvers, a modified initial guess for equilibrium constant is also introduced. Simulations of high-pressure transcritical two-phase temporal mixing layers and shock-droplet interaction were conducted using the ISAT-VLE CFD solvers. The simulation results show that the new method obtains a speed-up factor approximately from 10 to 60 and the ISAT errors can be controlled within 1%. The shock-droplet interaction results show that the DF scheme can achieve a higher speed-up factor than the FC scheme. The two sets of simulations exhibit the phase separation at high-pressure conditions. It was found that even at supercritical pressures with respect to each component, the droplet surface could still be in a subcritical two-phase state, because the mixture critical pressure is often significantly higher than each component and hence triggers phase separation. In addition, a shock wave could partially or completely convert the droplet surface from a subcritical two-phase state to a single-phase state by raising temperature and pressure.

1. Introduction

The demand for high-performance compact combustors is continually increasing due to the need for high chamber pressure, which is necessary to achieve high thrust/energy density and thermal efficiency. In rocket engines, the characteristic pressure ranges from 40 to 560 bar, while gas-turbine jet engines typically operate at 20 to 60 bar at sea-level takeoff. Diesel engines, on the other hand, operate within the range of 10 to 30 bar [1]. These high working pressures are comparable to or even higher than the

* Corresponding author.
E-mail addresses: zhan6305@umn.edu (H. Zhang), srini237@umn.edu (N. Srinivasan), suo-yang@umn.edu (S. Yang).

https://doi.org/10.1016/j.jcp.2024.112752
Received 15 July 2023; Received in revised form 29 November 2023; Accepted 1 January 2024

Available online 5 January 2024

critical pressure of air (37 bar) and liquid fuel (such as n-dodecane, n-$C_{12}H_{26}$, with a critical pressure of 18 bar). Consequently, transcritical/supercritical effects are introduced into the combustion system, causing a complete change in the underlying physics when compared to subcritical systems. The term **transcritical** in this work refers to an operating pressure higher than the critical pressure of the pure fuel or oxidizer while remaining below the cricondenbar pressures of their possible mixtures [2]. As a result, in transcritical flows, phase separation, or the coexistence of vapor and liquid phases, may occur when the temperature is not excessively high. On the other hand, the term **supercritical** fluid refers to an operating pressure higher than the cricondenbar pressures, where phase separation cannot occur, but strong non-ideal fluid effects can still be observed. Since in real combustors, the cricondenbar point of the fuel/oxidizer mixture can be very high (e.g., several hundred bars), most high-pressure combustion systems are transcritical in practice [2].

Supercritical jet breakup exhibits a distinct behavior from the classical mechanism observed at low pressures where surface tension plays a dominant role. At supercritical conditions, the interface between the pseudo phases (i.e., the liquid-like and gas-like phases) becomes diffused. This phenomenon has been extensively documented in many experimental studies involving both single-component and multi-component systems [3–7]. The absence of surface tension in these diffused interfaces alters the dynamics of jet breakup. Instead of being dominated by surface effects, the breakup process is primarily driven by diffusion [8,9]. In combustors, the physical processes that occur after injection follow a specific sequence: liquid atomization, evaporation, mixing, and chemical reactions. Atomization plays a crucial role in achieving complete evaporation, which can have a significant impact on cold ignition (e.g., high-altitude relight) and combustion efficiency within the combustor. As a result, it is essential to have a thorough understanding of high-pressure injection and mixing processes, particularly when designing modern combustors, including the injector design. However, there are great difficulties when attempting to model the non-reacting and reacting flows at high pressures.

The first difficulty in modeling transcritical/supercritical flows is that at high pressure, the gaseous/gas-like fluid does not satisfy the ideal gas law, and this makes transcritical/supercritical fluid behavior peculiar because of the large variation of thermodynamic properties (such as density and specific heat) near the critical point. Accurately describing these properties requires the use of a real-fluid equation of state (EOS). Among the commonly employed options, cubic EOS, such as the Peng–Robinson (PR) EOS [10] and Soave–Redlich–Kwong (SRK) EOS [11], are popular choices for transcritical/supercritical flow simulations due to their computational efficiency and acceptable accuracy. Additionally, volume translation methods can be utilized to improve density prediction [12]. While more sophisticated EOS are available, such as the Benedict–Webb–Rubin equation [13] and the statistical associating fluid theory (SAFT) [14] formulated in the Helmholtz free energy, they often come with higher computational costs and increased complexity. Implementing phase change models based on these more sophisticated EOS can be challenging due to their intricacy. Therefore, for this work, the PR EOS has been selected.

The non-ideal/real-fluid effects observed in transcritical/supercritical flows have been known to cause spurious pressure oscillations in numerical simulations that employ fully conservative (FC) schemes [15–19]. To address this issue, several approaches have been proposed that trade off full conservation for improved stability and accuracy, leading to the development of quasi-conservative schemes. One such scheme, introduced by Schmitt et al. [15], incorporates an artificial viscosity term derived from the pressure equation. By adding this term, the scheme effectively dampens pressure oscillations. Another approach, proposed by Johnson and Ham [20], introduces an additional transport equation for a function of the specific heat ratio. This method is suitable for well-defined interfacial flows with inert species. Lacaze et al. [21] put forward an enthalpy-based quasi-conservative scheme, which mitigates the loss of conservation. Abgrall and Karni developed the double flux (DF) method [22], which has been extended for reacting flows [23], as well as for use with discontinuous Galerkin (DG) schemes [24,25] and transcritical flows [26,27]. The DF method has demonstrated favorable behavior in handling the non-ideal effects associated with transcritical flows. In the present work, the DF method is incorporated into the computational fluid dynamics (CFD) solver and its performance and behavior are evaluated to better understand its ability to handle the challenges posed by transcritical/supercritical flows with phase change.

The second difficulty in modeling transcritical flows arises from phase change phenomena. While the operating conditions may be nominally supercritical with respect to a single component, the local mixture can remain in the subcritical two-phase regime. This occurs because the critical pressure of the mixture can be significantly higher than the critical pressure of each individual component forming the mixture [27–29]. Hence, most multi-component transcritical systems exhibit a hybrid behavior, with a subcritical two-phase interface (involving phase change) and a supercritical diffused mixing layer (without phase change) coexisting at different locations. This hybrid behavior will also be discussed in Sec. 3 of the present paper. Although there has been considerable interest in modeling transcritical/supercritical injection and mixing over the past three decades, most studies have struggled to capture this hybrid behavior effectively. Some researchers have used Lagrangian particles to track droplets [30–32], but the breakup and evaporation processes of the droplets were empirically modeled using sensitive tuning parameters. The tuning of these parameters heavily relies on the availability of experimental data. Another approach is the dense-fluid method [1,33,34], which accounts for the real-fluid effects but neglects phase boundaries and phase change since it assumes that the entire system is within the supercritical single-phase regime. However, as discussed in the works of [35,36], a phase change model has to be used to predict the properties of the mixture in such systems. To overcome the limitations of these methods, a two-phase diffuse-interface method based on vapor-liquid equilibrium (VLE) theory has been introduced in the recent decade. In this method, the fully compressible Navier–Stokes equations are solved to obtain local mixture properties such as density, internal energy, and component mass fraction. These properties are then used as inputs to compute other thermodynamic and transport properties using the VLE theory, which is based on a real-fluid EOS. The VLE theory is derived from the principle of minimizing the Gibbs free energy and accurately describes multiphase flow physics, including high-pressure phase change, preferential evaporation, and gas dissolution in the liquid phase. The VLE-based two-phase diffuse-interface method has been successfully applied to simulations involving ECN Spray A [37,38], flash boiling [39], supercritical

CO_2 systems [29], cavitation [40], mixing layers [41], and reacting flows [2,42]. It offers a promising approach for capturing the complex behavior of transcritical flows with phase change and has shown good agreement with experimental observations.

However, the computational cost of vapor-liquid equilibrium (VLE) calculations can be prohibitively high [29,38,43], particularly when dealing with a large number of components or a large computational domain. To address this issue, researchers have developed tabulation approaches that aim to accelerate thermodynamic model computation. One such approach is the tabulated thermodynamic model (based on the NIST REFPROP database and VLE with the PR EOS) proposed by Koukouvinis et al. [44]. This method stores the thermodynamic properties in a structured table in $log_{10}(P)$-T form, enabling efficient lookup during the simulation. Additionally, an efficient tabulation approach for binary mixtures, which supports reverse look-up using the inverse-distance weighting method, has been developed for VLE calculations of binary mixtures [43,45,46]. This approach has been successfully employed in investigating transcritical droplet injection. Furthermore, the tabulation method has been extended to ternary mixtures to study droplet evaporation [47]. We also developed a tabulation method based on Yi et al. [43] to investigate phase separation in supercritical CO_2 systems [29]. These tabulation methods offer computationally efficient and robust VLE solutions for CFD simulations. However, the memory requirement of tabulation methods grows exponentially with the number of components, leading to unaffordable memory demands for larger systems (i.e., the curse of dimensionality: table size $\sim M^N$, where M is the number of grids of each component in the table and N is the number of components). For example, using a table with the same resolution as our previous 3-component work [29] (i.e., a table with variable ranges: T of 470-600 K, P of 9-25 MPa, x_i of 0.6-0.9, and table grid sizes: $\Delta T = 1.3$ K, $\Delta P = 0.4$ MPa, $\Delta x_i = 0.015$) for a 4-component system would result in a table size of approximately 4 terabytes (4 TBs). Storing such large tables in each CPU core (without shared memory) or each node (with shared memory) becomes infeasible for existing computing facilities. Consequently, these traditional tabulation methods are typically limited to binary or at most ternary systems and are unsuitable for multiphase combustion and other multi-component problems. Recently, artificial neural networks (ANNs) have emerged as an alternative to tabulation methods and have been proposed as nonlinear regression models for thermodynamic properties [48,49]. ANNs offer the advantage of easier extension to a large number of components, mitigating the curse of dimensionality associated with tabulation methods. However, controlling the error and ensuring accurate performance of ANNs can be challenging, as their accuracy heavily relies on the training process.

In this work, we developed a novel method for accelerating VLE-based CFD simulations by employing *In Situ* Adaptive Tabulation (ISAT) [50]. Originally designed to expedite combustion chemistry calculations, ISAT has proven to be adaptable to various problem domains, including chemical engineering [51–53], surface reactions [54], thin film growth [55], solid mechanics [56], control systems [57], and turbulent combustion models based on manifolds [58]. By constructing the tabulation on-the-fly, ISAT minimizes memory requirements and effectively addresses the curse of dimensionality by storing only essential data. Moreover, it achieves computational efficiency comparable to traditional tabulation methods. Unlike other approaches, our proposed ISAT-VLE method does not necessitate pre-processing steps such as table generation or neural network training. Computation is performed in real-time, eliminating the need for extensive setup procedures. This method is integrated with a CFD solver based on the central-upwind scheme [59,60], enabling the simulation of multiphase transcritical flows. To evaluate the performance and error control of our ISAT-VLE method, we conducted simulations of transcritical temporal mixing layer and shock-droplet interaction. Phase separation is captured in both simulations. In the shock-droplet interaction simulations, we compared the performance and error control of ISAT-VLE of the DF scheme with those of the FC scheme and evaluated the DF's spurious oscillation removal.

This paper is organized as follows: Section 2 provides details of the numerical methods used in this work, including EOS, VLE solvers, numerical scheme, and ISAT-VLE methods implementation. This is followed by Sec. 3, which evaluates the behavior of the ISAT-VLE method using simulations of temporal mixing layer and shock-droplet interaction. The DF scheme is compared with FC scheme using the simulation of shock-droplet interaction. The key physics of the simulation results are also discussed to demonstrate the unique capability of the ISAT-VLE method. Section 4 summarizes the main conclusions of this paper.

2. Numerical modeling

We have developed an ISAT-VLE transcritical flow solver, and in this section, we discuss the methodology employed in the solver. In Sec. 2.1, we present the calculation methods for thermodynamic and transport properties, including the Equation of State, Vapor-Liquid Equilibrium (VLE) solvers, and the transport property model. Next, in Sec. 2.2, we describe the Computational Fluid Dynamics (CFD) framework, governing equations, numerical scheme, and the utilization of the double flux method in our simulations. Then, in Sec. 2.3, we explain the implementation of the ISAT method to accelerate the VLE calculation, as well as our novel record deletion methods. Finally, in Sec. 2.4, we provide an analytic framework to support the VLE-ISAT calculation.

2.1. Models of thermodynamic and transport properties

2.1.1. Peng-Robinson (PR) equation of state (EOS)

Transcritical flow simulations require a real-fluid equation of state (EOS). Considering the computational efficiency, acceptable accuracy, and simple implementation, the Peng-Robinson (PR) EOS [10] is used in this work. The PR EOS is formulated as:

$$P = \frac{RT}{V-b} - \frac{a}{V(V+b)+b(V-b)}, \tag{1}$$

where R is the ideal gas constant, V is molar volume. The energy parameter a and co-volume parameter b for pure component i are formulated as:

$$a_i = 0.45724 \frac{R^2 T_{c_i}^2}{P_{c_i}} \hat{a}, \qquad b_i = 0.07780 \frac{R T_{c_i}}{P_{c_i}}, \tag{2}$$

$$\hat{a} = \left[1 + \kappa \left(1 - \sqrt{\frac{T}{T_{c_i}}} \right) \right]^2, \qquad \kappa = 0.37464 + 1.54226 \omega_i - 0.26992 \omega_i^2, \tag{3}$$

where T_{c_i} is the critical temperature of component i, P_{c_i} is critical pressure of component i, and ω_i is the acentric factor of component i. These specie properties are obtained from the NIST REFPROP database. When the PR EOS is used for a multi-component mixture, mixing rules are used to calculate the energy parameter a and co-volume parameter b [61]:

$$a = \sum_i \sum_j \chi_i \chi_j \sqrt{a_i a_j} (1 - k_{ij}), \qquad b = \sum_i \chi_i b_i, \tag{4}$$

where χ_i is the mole fraction of component i, and k_{ij} is the binary interaction parameter. The binary interaction parameters utilized in this study were obtained from researches [62,63], where k_{ij} values are determined by minimizing the discrepancy between experimental and simulation outcomes.

2.1.2. Vapor-liquid equilibrium (VLE)

VLE can determine multi-component mixtures' phase boundaries to capture phase change in transcritical flows. In the VLE theory, the multiphase thermodynamic equilibrium is locally satisfied, which requires the equality of pressure P (i.e., mechanical equilibrium), the equality of temperature T (i.e., thermal equilibrium), and the equality of fugacity f of the two phases for each component i (i.e., chemical equilibrium):

$$P_l = P_v = P, \quad T_l = T_v = T, \quad f_{i,l} = f_{i,v}, \quad i = 1, ..., N, \tag{5}$$

where the subscript "l" and "v" refer to liquid and vapor, respectively; f_i is the fugacity of component i; and N is the number of components.

There are many different types of VLE solvers, such as isothermal-isobaric (TP) flash [64] and isoenergetic-isochoric (UV) flash [65]. All of them need to solve a two-phase system constrained by Eq. (5), together with the mass conservation of each component. To determine the mixture phase properties, in addition to the mole fraction of each component (i.e., the composition of the mixture), two more mixture properties are required. These two properties determine the type of VLE solvers. For example, TP flash uses the mixture temperature (T) and pressure (P) as inputs to solve the system. The VLE solution provides mixture phase properties, including the vapor phase molar fraction β and molar fraction of each component in each phase: x_i in liquid phase and y_i in gas/vapor phase.

The single-phase thermodynamic properties can be obtained using EOS (including the departure functions derived from EOS) and NASA Polynomials. Additional blend rules are required to bridge the gap between single-phase and two-phase systems. We follow a commonly used method [27,37] which blends extensive volume and internal energy, and hence mixture density and mixture specific internal energy are formulated as:

$$\frac{M}{\rho} = \beta \frac{M_v}{\rho_v} + (1 - \beta) \frac{M_l}{\rho_l}, \tag{6}$$

$$M e = \beta M_v e_v + (1 - \beta) M_l e_l, \tag{7}$$

where M is the mixture molar mass, M_v and M_l are vapor phase and liquid phase molar mass, respectively, and e is specific internal energy. Other mixture thermodynamic properties (e.g., mixture enthalpy and mixture specific heat) can be derived based on these properties. In this research, TP flash, UV flash, and isobaric-isochoric (PV) flash are applied. Specifically, TP flash solves VLE problems with a given temperature (T) and pressure (P), and is used for the initialization step of the CFD simulation; PV flash solves problems with given pressure (P) and specific volume (V), and is used in the double flux (DF) scheme (in Sec. 2.2); UV flash solves problems with given internal energy (U) and specific volume (V), and is used in the fully conservative (FC) scheme.

Isothermal and isobaric (TP) flash: VLE is governed by fugacity equality Eq. (8) and Rachford-Rice equation [66] Eq. (9). Rachford-Rice equation was derived from the mass conservation of each component.

$$f_{i,l} / f_{i,g} = 1, \tag{8}$$

$$\sum_{i=1}^{N} \left\{ z_i \left(1 - K_i \right) \Big/ \left[1 + \left(K_i - 1 \right) \beta \right] \right\} = 0, \tag{9}$$

$$\sum_{i=1}^{N} x_i = \sum_{i=1}^{N} y_i = 1, \tag{10}$$

where x_i is the mole fraction of component i in liquid phase, y_i is the mole fraction of component i in gas/vapor phase, z_i is the mole fraction of component i in the mixture, β is the vapor mole fraction, K_i is the equilibrium constant of component i, and $K_i = y_i / x_i$.

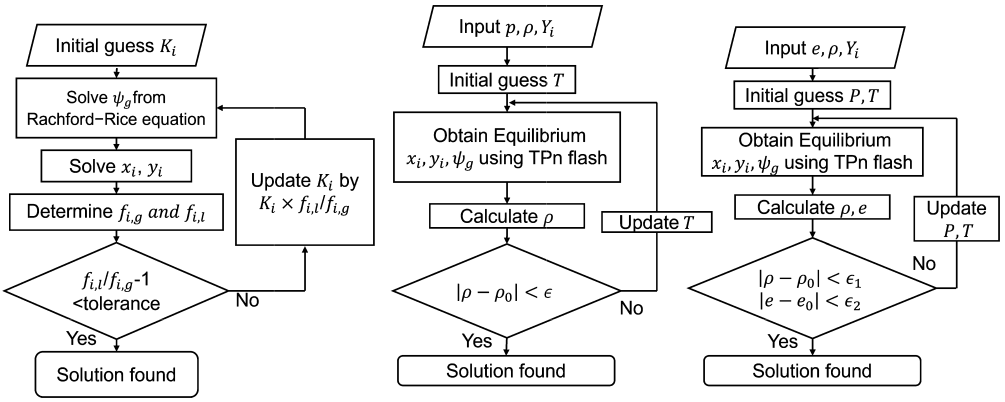

Fig. 1. Flow chart of VLE solver. Left: TP flash solver. Middle: PV flash solver. Right: UV flash solver.

The fugacity formula of PR EOS [39] is shown below:

$$f_{i,p} = P\chi_i \exp\left\{\frac{b_i}{b_p}(Z_p - 1) - \ln(Z_p - \frac{b_p P}{RT}) - \frac{a_p}{2\sqrt{2}b_p RT}\left(\frac{2\sum_j x_j a_j}{a_p} - \frac{b_i}{b_p}\right)\ln\left[\frac{V_p + \left(1+\sqrt{2}\right)b_p}{V_p + \left(1-\sqrt{2}\right)b_p}\right]\right\}, \tag{11}$$

where χ_i is the mole fraction of component i in phase p (for liquid, $\chi_i = x_i$; for vapor phase, $\chi_i = y_i$). Z_p is the compressibility factor of phase p. a_p and b_p are the mixture energy parameter and mixture co-volume parameter of phase p, respectively. Note that specie parameters a_i and b_i only depend on the pure component properties.

Equations (8)-(11) are solved based on the successive substitution method [67]. The flow chart of TP flash solution process is shown in Fig. 1 (left). The Wilson equation [68] is commonly used to obtain initial guess:

$$K_{i,wil} = e^{5.373(1+\omega_i)(1-T_{c_i}/T_i)} P_{c_i}/P_i \tag{12}$$

However, when dealing with high-pressure conditions, the Wilson equation may encounter difficulties in accurately capturing phase separation, leading to the generation of trivial single-phase solutions. To overcome this limitation, we have made modifications to the Wilson equation (see below) to improve its ability to obtain potential phase separation solutions. The idea comes from the fact that components with larger equilibrium constants exhibit a greater inclination to enter the vapor phase, while components with smaller constants tend to remain in the liquid phase. In order to enhance the representation of phase separation, we have further strengthened this inherent tendency. Specifically, we have amplified the impact of larger equilibrium constants by increasing their values, and simultaneously reduced the influence of smaller constants by decreasing their values:

$$K_{i,init} = \begin{cases} K_{i,wil}\phi\gamma & \text{if } K_{i,wil} = max_j\left(K_{j,wil}\right) \\ K_{i,wil}\phi/\gamma & \text{otherwise} \end{cases} \tag{13}$$

$$\gamma = \sqrt{max_j\left(K_{j,wil}\right)/min_j\left(K_{j,wil}\right)} \qquad \phi = \left[max_j\left(K_{j,wil}\right) \times min_j\left(K_{j,wil}\right)\right]^{-0.25} \tag{14}$$

In comparison to the original Wilson equation, the newly proposed initial K values exhibit a larger ratio between the maximum and minimum equilibrium constants. This adjustment provides an initial guess that has stronger phase separation tendencies. In the case of three-component systems, where the species with intermediate K_{init} values can potentially exist in either the liquid or gas phase, an inappropriate initial guess can lead to incorrect solutions. To address this issue, we introduce a modification by taking the reciprocal of the intermediate K_{init} value while keeping the other K_{init} values unchanged. This modification alters the phase tendencies of these intermediate species. Both the modified and original initial conditions are employed as initial guesses to solve the VLE system, considering the gas and liquid tendencies of these intermediate K_{init} species in the calculations.

In a multi-component mixture, there could be multiple nontrivial solutions of the VLE system. When multiple initial conditions are employed, it is possible to obtain multiple solutions for the system. To select the final solution from these options, the conventional approach involves choosing the solution with the lowest Gibbs free energy. However, during the actual calculations, we encountered a unique scenario where an exception arose. In this case, although a solution may possess the minimum Gibbs free energy, the compressibility factor (Z) of both its liquid and gas phases approaches 1. This implies that the phases resemble an ideal gas, and the established phase equilibrium occurs between two immiscible gaseous phases. While such a solution can be obtained computationally, it is not physical and can lead to computation failures. Through simulation tests in Sec. 3, we have identified and ruled out solutions wherein the compressibility factor (Z) exceeds 0.87 in the liquid phase (which is non-physical). This exclusion criterion ensures the proper functioning of the VLE solver and helps prevent the occurrence of non-physical solutions.

Once the initial guess is obtained, we proceed to solve the Rachford-Rice equation (Eq. (9)) using the Newton-Raphson iteration method to determine the vapor mole fraction β. Subsequently, x_i and y_i can be calculated using Eqs. (10). The next step involves

evaluating the fugacity using Eq. (11) and examining whether fugacity equilibrium has been achieved (i.e., $\left|f_{i,l}/f_{i,g}-1\right|<tol$). If equilibrium has not been reached, we update the equilibrium constant K_i by multiplying it with the fugacity ratio $K_i = K_i \times f_{i,l}/f_{i,g}$ and then return to the step of solving the Rachford-Rice equation once again. This iterative process continues until the error falls below a specified tolerance tol, and tol is 10^{-8} in this work, which is sufficient to achieve an accurate solution according to our test.

PV flash and UV flash: The PV flash and UV flash solvers are developed based on the TP flash solver, utilizing iteration methods (see Fig. 1 middle and right, respectively). For both solvers, initial guesses (T for PV flash; T and P for UV flash) are obtained from the cell value in the previous time step, and a TP flash problem is solved in each iteration using the initial guesses. During the iteration process, T and/or P are updated to find a better solution (see details in the next two paragraphs). After multiple iterations, when the error (PV flash error: $|\rho-\rho_0|$; UV flash error: $|e-e_0|$ and $|\rho-\rho_0|$) falls below a specified tolerance, the solver returns the solution of TP flash problem.

In the PV flash solver, as the pressure (P) is already provided as an input, only the temperature (T) needs to be guessed and updated during the iteration. To avoid the computational cost of computing derivatives in the Newton-Raphson method, a secant method is employed to update T.

In contrast, the UV flash solver requires simultaneous guessing and updating of two variables, i.e., temperature (T) and pressure (P), during the iteration. Therefore, the secant method cannot be directly applied. Instead, the Newton-Raphson method is used to solve UV flash problems. The required Jacobian matrices are obtained using the analytical framework described in Sec. 2.4.

2.1.3. Models of transport properties

To evaluate the dynamic viscosity and thermal conductivity at transcritical conditions, the dense fluid formulas (i.e., the Chung's method [69]) are used. This method gives accurate estimations of viscosity and thermal conductivity of polar, non-polar, and associating pure fluids and mixtures. In this method, dynamic viscosity and thermal conductivity have the similar formula:

$$\lambda = \lambda_0 \lambda^* + \lambda_p, \tag{15}$$

where λ represents dynamic viscosity μ or thermal conductivity κ. λ_0 is the property at the low-pressure limit. λ^* and λ_p are high pressure corrections. At high pressures, λ_p is the major contributing term compared to $\lambda_0\lambda^*$. On the other hand, at low pressures, λ^* is approaching unity and the λ_p term is negligible such that Eq. (15) reduces to λ_0. Hence, the transition between low-pressure and high-pressure properties is smoothly described by the model.

For mass diffusivity, we used a mixture-averaged mass diffusion model [70]. The mass diffusion coefficient D_i of specie i is defined as

$$D_i = \frac{1-Y_i}{\sum_{j\neq i}^{N} z_j/D_{j,i}}, \tag{16}$$

where Y_i and z_i are the mass and mole fractions of component i, respectively; $D_{i,j}$ is the binary mass diffusion coefficient of component i in component j, which is evaluated by Fuller's model [71] with Takahashi's correction [72] for high pressures.

2.2. VLE-based CFD framework

In this work, a transcritical multiphase flow solver is developed by coupling a density-based fully compressible CFD solver with several VLE solvers. The governing equations of the CFD solver are the multi-component transport equations, including the mixture density equation (i.e., continuity equation), mixture momentum equations, mixture internal energy equation, and component mass fraction equations as follows:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0, \tag{17}$$

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} = -\frac{\partial P}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j}, \tag{18}$$

$$\frac{\partial \rho e}{\partial t} + \frac{\partial \rho u_i e}{\partial x_i} + \frac{\partial \rho K}{\partial t} + \frac{\partial \rho u_i K}{\partial x_i} + \frac{\partial u_i P}{\partial x_i} = -\frac{\partial q_i}{\partial x_i} + \frac{\partial \tau_{ij} u_j}{\partial x_i}, \tag{19}$$

$$\frac{\partial \rho Y_m}{\partial t} + \frac{\partial \rho Y_m u_i}{\partial x_i} = \frac{\left(\rho Y_m V_{mi}\right)}{\partial x_i}, \tag{20}$$

where u_i is the mixture velocity, ρ is the mixture density, e is the specific internal energy, Y_m is the mass fraction of component m, τ_{ij} is the viscous stress tensor of the Newtonian fluid defined by $\tau_{ij} = -\frac{2}{3}\mu\frac{\partial u_k}{\partial x_k}\delta_{ij} + \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$ where μ is the dynamic viscosity, K is the kinetic energy, and q_i is the heat flux. $q_i = -\kappa\nabla T$ where κ is the thermal conductivity. The Fickian diffusion is used to evaluate the diffusion velocity in the governing equations:

$$V_{mi} = -\frac{D_m}{X_m}\frac{\partial X_m}{\partial x_i} + \sum_{s}^{N}\frac{Y_s D_s}{X_s}\frac{\partial X_s}{\partial x_i},$$

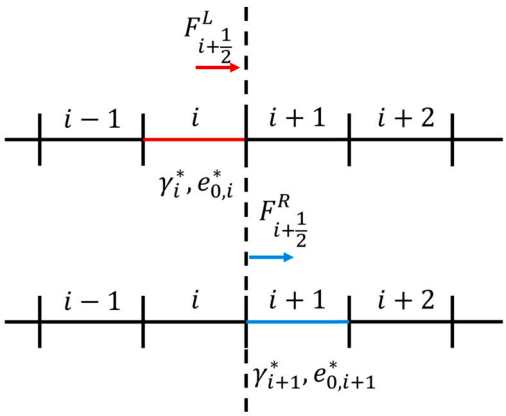

Fig. 2. Schematic of the double flux (DF) method in a one-dimensional (1-D) mesh.

where X_m and D_m are the mole fraction and the mixture-averaged diffusion coefficient of component m, respectively. The second term on the right-hand-side (RHS) is a correction term to ensure global mass conservation $\sum_m^N \rho Y_m V_m = 0$.

The CFD solver was developed based on the central-upwind scheme [60,73], which is a fully conservative (FC) scheme. At each time step, the conservative variables ρ, ρu, ρe, and ρY_m are updated according to the governing equations (i.e., Eqs. (17)-(20)).

To eliminate spurious pressure oscillations, a quasi-conservative approach called the double flux (DF) scheme [22,23,26] is also implemented. This scheme locally freezes the real-fluid properties to those of an equivalent calorically perfect gas. During the calculation of the new time step, in each grid cell, a local calorically perfect-gas EOS is defined using the thermodynamic properties. The pressure field is updated using this newly defined EOS, effectively mitigating the occurrence of spurious pressure oscillations. Subsequently, the remaining thermodynamic properties such as density and component mass fractions are updated using the real-fluid EOS and PV flash based on the updated pressure, density, and component mass fraction values. Mathematical details of the DF scheme are presented in the next paragraph. The use of a calorically perfect-gas EOS in the DF scheme has been shown by Ma et al. [26] to successfully remove spurious pressure oscillations.

To define a locally calorically perfect-gas EOS, an effective specific heat ratio γ^* and an effective internal energy e_0^* are formulated as:

$$\gamma^* = \frac{\rho c^2}{P}, \qquad e_0^* = e - \frac{Pv}{\gamma^* - 1}, \tag{21}$$

where c is the speed of sound, and v is the specific volume. With real-fluid EOS, γ^* and e_0^* can be nonlinear functions of the thermodynamic states and may not be constant. The main idea of the DF scheme is to freeze γ^* and e_0^* in space during each time-step advancement. When calculating a flux of a face of a certain grid cell, the conservative variables used are not directly from the cell values, but reconstructed using γ^*, e_0^*, P, and ρ. Although this face also belongs to the adjacent cell, its γ^*, e_0^* of the adjacent cell are not used. For example, on a one-dimensional (1-D) mesh, the numerical face flux $F^L_{i+\frac{1}{2}}$ for the cell i at $x_{i+\frac{1}{2}}$ can be expressed as (see Fig. 2):

$$F^L_{i+\frac{1}{2}} = F\left(U_{L,i+\frac{1}{2}}, U_{R,i+\frac{1}{2}}\right), \qquad U_{L,i+\frac{1}{2}} = U_L\left(\gamma_i^*, e_{0,i}^*, P, \rho, Y\right), \qquad U_{R,i+\frac{1}{2}} = U_R\left(\gamma_i^*, e_{0,i}^*, P, \rho, Y\right), \tag{22}$$

where U_L and U_R are conservative variables. To calculate U_L and U_R, the following steps are performed. First, the conservative variable ρe in each cell is reconstructed using the calorically perfect-gas Equation of State (EOS) given by Eq. (21). Then, ρe and other conservative variables are interpolated to the face. Bias interpolation methods, such as the upwind scheme, are commonly used, resulting in the definition of two reconstructed values, U_L and U_R, based on the direction of the bias. The subscript "L" is used to denote the left reconstructed value at a face, while "R" represents the right reconstructed value. It is important to note that when calculating U_L and U_R, only the γ^* and e_0^* values from the cell i are used. When using the γ^* and e_0^* values in the cell $i+1$, the right flux $F^R_{i+\frac{1}{2}}$ is obtained (see in Fig. 2). The interpolation method uses van Leer limiter [74] in the current work. In the central-upwind scheme, the formulas become:

$$F^L_{i+\frac{1}{2}} = \alpha u_L U_L + (1-\alpha) u_R U_R + \omega\left(U_R - U_L\right), \qquad \alpha = \frac{\phi_L}{\phi_L + \phi_R}, \qquad \omega = \alpha(1-\alpha)\left(\phi_L + \phi_R\right), \tag{23}$$

$$\phi_L = max\left(c_L + |u_L|, c_R + |u_R|, 0\right), \qquad \phi_R = max\left(c_L - |u_L|, c_R - |u_R|, 0\right), \tag{24}$$

where c and u are the speed of sound and velocity, respectively. They are obtained from the same interpolation scheme as U_L and U_R. In this way, the γ^* and e_0^* are frozen, and the fluid is locally calorically perfect to remove spurious pressure oscillation [26]. Since each face is adjacent to two grid cells, the two different fluxes at the same face (i.e., the left flux $F^L_{i+\frac{1}{2}}$ and the right flux $F^R_{i+\frac{1}{2}}$

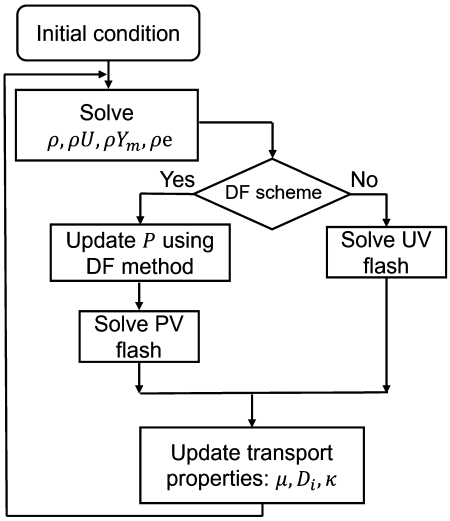

Fig. 3. Flow chart of the VLE-based CFD framework.

in Fig. 2) can be obtained. That is why this scheme is called "double flux", and two fluxes make the numerical scheme no more conservative.

At each time step, ρ, u, and Y_m are updated using the central-upwind scheme. Then, the DF scheme is used to update e and P. After that, T, β, and c are obtained using the PV flash solver. In contrast, in the fully conservative scheme (i.e., without DF), P, T, β, c are directly updated using the UV flash solver. This entire process is shown in Fig. 3. DF method is useful to solve the system with large density gradients, but since it is not an energy-conservative scheme, it could cause other problems. In Sec. 3.2.2, we will compare the results of two schemes, and discuss their pros and cons.

2.3. *In situ* adaptive tabulation (ISAT)

The *in situ* adaptive tabulation (ISAT) method, originally introduced by Pope [50], offers an effective approach to reduce the computational cost associated with detailed chemistry calculations. In contrast to traditional tabulation methods that require pre-processing to generate tables before the CFD simulation, ISAT dynamically constructs the table during the simulation itself. This dynamic construction enables us to store only the necessary records, significantly reducing the table size and mitigating the curse of dimensionality. During the CFD simulation, most of the queries can be efficiently retrieved by utilizing linear approximation based on the table records, eliminating the need for direct calculations. This approach not only enhances computational efficiency, but also ensures that the results are obtained with a high degree of accuracy. Particularly, ISAT's ability to provide excellent error control contributes to maintaining the desired level of precision. Given these advantages, employing the ISAT method to tabulate VLE solutions proves to be a promising strategy for accelerating VLE-based CFD simulations, which will be shown in the present study.

The relation between the given condition $\boldsymbol{x}$ and solution $\boldsymbol{y}$ from a VLE solver can be denoted as a function or mapping $\boldsymbol{F}$:

$$\boldsymbol{y} = \boldsymbol{F}(\boldsymbol{x}).$$

Specifically, certain variables are required for the implementation of the FC and DF schemes, which require UV flash and PV flash solvers, respectively. Firstly, the speed of sound c is required to ensure the accuracy and stability of the CFD solver. Additionally, the vapor mole fraction β is used for capturing phase separation/change. c and β are two output variables of all VLE solvers. The input and other output variables depend on which VLE flash solver is used. For the PV flash solver, the input variables consist of the pressure (P), density (ρ), and mole fractions (z_i) of each component i in the mixture. The solver then provides the following output variables: temperature (T), specific internal energy (e), speed of sound (c), and vapor mole fraction (β). In contrast, the UV flash solver takes the specific internal energy (e), density (ρ), and mole fractions (z_i) as input variables. It then calculates and provides the output variables of temperature (T), pressure (P), speed of sound (c), and vapor mole fraction (β).

For every record in the ISAT table, it contains ($\mathbf{x}_0, \mathbf{y}_0, \left.\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0}, \mathbf{M}$). The gradient (i.e., the Jacobian matrix $\left.\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0}$) is evaluated analytically (in Sec. 2.4) and used for local linear approximation:

$$\mathbf{y} \approx \mathbf{y}_{linear} = \mathbf{y}_0 + \left.\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0} \cdot (\mathbf{x} - \mathbf{x}_0).$$

The matrix $\mathbf{M}$ is used to define the region of accuracy (ROA), in which the local error ϵ does not exceed the tolerance ϵ_{tol}. The ROA is defined by an inequality:

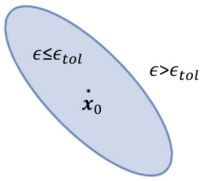

Fig. 4. Sketch of region of accuracy (ROA).

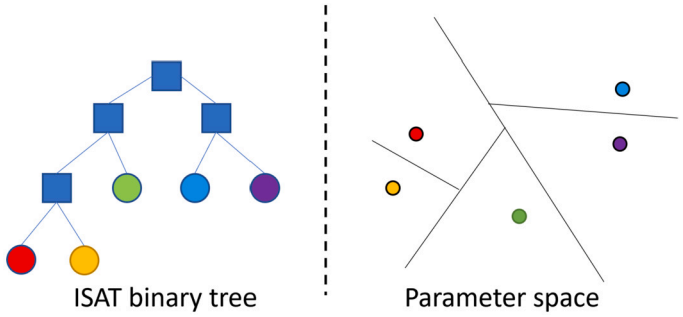

Fig. 5. The binary tree used to manage records in ISAT.

$$\left(\mathbf{x}-\mathbf{x}_0\right)^T \mathbf{M}\left(\mathbf{x}-\mathbf{x}_0\right) \leq 1.$$

The region satisfying this inequality is a hyper-ellipsoid, and hence the ROA is also called the ellipsoid of accuracy (EOA), as shown in Fig. 4. For the initial setting, the linear term $\left.\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0} \cdot \left(\mathbf{x}-\mathbf{x}_0\right)$ is considered as an error. So, the initial $\mathbf{M}$ can be set as

$$\mathbf{M} = \left(\left.\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0}\right)^T \left(\left.\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0}\right) / \epsilon_{tol}^2.$$

The management of records in the ISAT method is facilitated through a binary tree structure, as illustrated in Fig. 5. Each node of the tree stores a vector that defines a hyperplane, dividing the parameter space into two distinct half-spaces. The data corresponding to each half-space is stored in the respective sub-trees. By employing this tree structure, the entire parameter space can be divided into numerous small cells, with the records being stored at the leaf nodes. During the search operation, at each layer of the tree, the sub-tree is selected based on which half-space contains the input parameters. This process continues until the desired record is found. It is important to note that the record obtained through this method may not be the closest to the input data. Nevertheless, this approach is simple and efficient enough to satisfy the requirements of the simulation. When inserting a record A into the table, the following steps are performed. First, a record B is obtained from the table using the search operation, with A used as input. Using A and B records, a median plane is determined. Based on this median plane, a new node is created and replaces the record B in the tree. Both A and B are then inserted as leaves of this new node. This process ensures that the new record A is properly integrated into the existing binary tree structure. The ISAT binary tree is not a balanced tree. Consequently, during simulations, a large number of records may be concentrated in one of the two sub-trees, resulting in an increased height of the tree structure. This can negatively impact the performance of look-up and insertion operations. To prevent such degradation, a rebuilding process is implemented. Specifically, at every 20 time steps, if the tree height exceeds twice the height of a perfectly balanced tree, a new tree structure is constructed using the existing records to restore the perfect balance and optimize the performance. During the construction of each layer of the binary tree, the direction of maximum variance is identified. This direction is used to establish a tree node that separates the records into two groups of equal size. Subsequently, two subtrees are constructed using the two groups of records. This process is then recursively applied to construct the two groups of records in the same manner. Through this approach, a perfectly balanced tree is created.

In a simulation, for the first query, a new record is calculated and added to the table. For subsequent queries ($\mathbf{x}_{new}$), the closest record ($\mathbf{x}_0, \mathbf{y}_0, \left.\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0}, \mathbf{M}$) is searched out, and then one of the following three operations is conducted, as shown in Fig. 6:

(1) Retrieval: If $\mathbf{x}_{new}$ is within the EOA of the record, then the linear approximation $\mathbf{y}_{linear}$ is returned. Otherwise $\mathbf{y}_{new} = \mathbf{F}\left(\mathbf{x}_{new}\right)$ is calculated.

(2) Growth: If $\left|\mathbf{y}_{new} - \mathbf{y}_{linear}\right| \leq \epsilon_{tol}$, then EOA is grown to include $\mathbf{x}_{new}$. Specifically, the new EOA is the smallest ellipsoid covering both the old EOA and $\mathbf{x}_{new}$. Finally, $\mathbf{y}_{new}$ is returned.

(3) Addition: Otherwise $\left|\mathbf{y}_{new} - \mathbf{y}_{linear}\right| > \epsilon_{tol}$, then the new record is added to the ISAT table, and $\mathbf{y}_{new}$ is returned.

In the ISAT method, a maximum number of records is set due to the limitation of memory. When the number of records reaches the maximum, the table is full, and new addition actions cannot be taken. In Lu and Pope [75], three actions were compared: (a) do not add the new record to the table; (b) remove the least recently used record; (c) remove the least frequently used record. It was found that for non-stationary problems, action (b) is likely the optimal choice. Lu and Pope [75] maintained a linked list (called

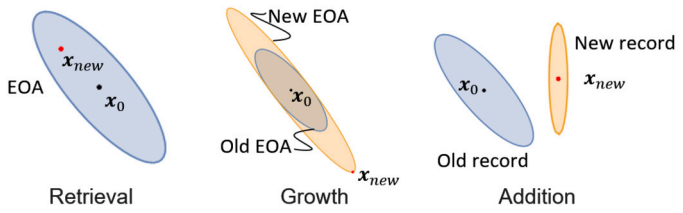

Fig. 6. Sketch showing the three operations in the ISAT method.

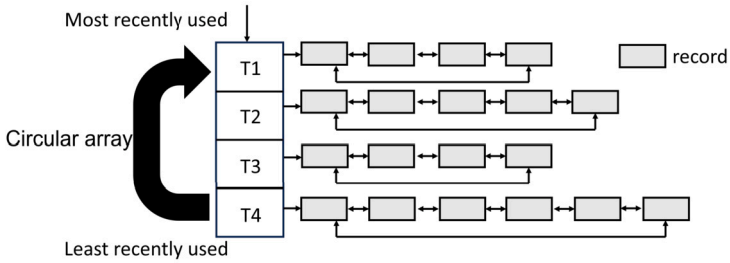

Fig. 7. Schematic of the data structure to support proposed methods for removing redundant records.

MFU list) to obtain the least recently used record. In the list, the records are in the order in which they have most frequently been used. We follow their idea of action (b) and further improve the method to remove records (see below for our improvement).

In our tests, we realized that the height of the binary tree becomes larger when there are more records, making the search slower. Moreover, in the non-stationary simulation, a large number of records are required only in a certain time interval, after which many records are redundant and only degrade the performance of the search. These redundant records should be accurately detected and quickly deleted to improve the ISAT performance. To address these additional issues, we propose two methods:

- delete data that has not been used in the last few time steps.
- use an adaptive table maximum size, and the size is formulated as:

$$N_{max} = C \sum_{i=0}^{M} N_{\text{inactive},i}, \tag{25}$$

where $N_{\text{inactive},i}$ is the number of records of which the last call was i time steps ago. M is the number of time steps considered and C is a multiplier ($C > 1$). M and C are tunable parameters.

To determine the parameters used in the two methods, we analyze the distribution of the number of steps since the last call. Based on our observations, we can give a general range of values. For the first method, For the first method, a maximum unused step of 70-90 can be utilized. In Sec. 3.2.3, we selected 80 as the maximum unused step. If a record has not been used within this number of time steps, it is highly unlikely to be used again. For the second method, we calculated the proportion of the most commonly used records to the total number of records to determine the parameters. To achieve faster removal of redundant data, the value of M should be a small number, and within the range of 3-10, and C should be within the range of 3-7. In Sec. 3.2.3, we used $C = 5.8$ and $M = 5$. Conducting a test simulation can provide further insights into the usage of records and help determine the optimal parameters for the specific case at hand.

We propose a new data structure to support the proposed methods. The data structure is a list of linked lists, as shown in Fig. 7. It has multiple layers (4 layers in Fig. 7), T_1 stores the records just used in the previous time step, T_2 stores the records used in the time step before the previous time step, and so on and so forth. T_4 has all records not been used within 3 latest time steps. With this list, all records are sorted according to the distance between the current time step and the time step of the last call, and the proposed deletion methods can be efficiently implemented. The main list ($T_1 - T_4$ list) is a circular array, the list head is the layer of the most recently used records and the tail is the layer of the least recently used records. In a new time step, the linked list of the tail (T_4) is merged with the previous layer (T_3), and this empty layer (T_4) becomes the new head, and all records called during this time step will be inserted into this empty layer. The previous layer (T_3) becomes the new tail. Since we use a combination of a circular array and linked lists, the above operations can be accomplished achieved with O(1) time complexity.

2.4. Analytical framework for VLE

The first-order derivatives of thermodynamic properties considering phase change are required when calculating the speed of sound (Eq. (40)), solving the VLE system (e.g., using the Newton-Raphson iteration method), and using the ISAT-VLE method (e.g., evaluating the Jacobian matrix $\left.\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0}$). These derivatives can be obtained numerically by a perturbation method, which is computationally expensive and introduces additional errors. Most importantly, profiles of thermodynamic properties can have dis-

continuities/jumps at the phase boundaries, and hence the numerical derivatives from the perturbation method can be incorrect and diverge the simulation. To address all the above challenges, we developed an analytical framework to evaluate the required VLE derivatives, built upon the analytic work of Tudisco and Menon [76].

In order to calculate the derivatives of thermodynamic properties considering phase change, we need to first obtain a few derivatives listed below:

$$\left(\frac{\partial x_k}{\partial T}\right)_{P,\mathbf{z}}, \left(\frac{\partial y_k}{\partial T}\right)_{P,\mathbf{z}}, \left(\frac{\partial x_k}{\partial P}\right)_{T,\mathbf{z}}, \left(\frac{\partial y_k}{\partial P}\right)_{T,\mathbf{z}}, \left(\frac{\partial x_k}{\partial z_i}\right)_{T,P,z_{s,s\neq i}}, \left(\frac{\partial y_k}{\partial z_i}\right)_{T,P,z_{s,s\neq i}} \tag{26}$$

The analytic formulas of these terms are provided in Appendix A, and their derivation is based on Tudisco and Menon [76].

For single-phase properties, the temperature derivatives can be obtained using the chain rule:

$$\left(\frac{\partial(\cdot)_l}{\partial T}\right)_{P,\mathbf{z}} = \left(\frac{\partial(\cdot)_l}{\partial T}\right)_{P,\mathbf{x}} + \sum_{k=1}^{N}\left(\frac{\partial(\cdot)_l}{\partial x_k}\right)_{P,T,x_{s,s\neq k}}\left(\frac{\partial x_k}{\partial T}\right)_{P,\mathbf{z}} \tag{27}$$

$$\left(\frac{\partial(\cdot)_v}{\partial T}\right)_{P,\mathbf{z}} = \left(\frac{\partial(\cdot)_v}{\partial T}\right)_{P,\mathbf{y}} + \sum_{k=1}^{N}\left(\frac{\partial(\cdot)_v}{\partial y_k}\right)_{P,T,y_{s,s\neq k}}\left(\frac{\partial y_k}{\partial T}\right)_{P,\mathbf{z}} \tag{28}$$

The pressure derivatives use similar formulas with all the $\left(\frac{\partial(\cdot)}{\partial T}\right)$ replaced by $\left(\frac{\partial(\cdot)}{\partial P}\right)$.

The z derivatives are formulated as:

$$\left(\frac{\partial(\cdot)_l}{\partial z_i}\right)_{P,T} = \sum_{k=1}^{N}\left(\frac{\partial(\cdot)_l}{\partial x_k}\right)_{P,T,x_{s,s\neq k}}\left(\frac{\partial x_k}{\partial z_i}\right)_{P,T,\mathbf{z}} \tag{29}$$

$$\left(\frac{\partial(\cdot)_v}{\partial z_i}\right)_{P,T} = \sum_{k=1}^{N}\left(\frac{\partial(\cdot)_v}{\partial x_k}\right)_{P,T,x_{s,s\neq k}}\left(\frac{\partial y_k}{\partial z_i}\right)_{P,T,\mathbf{z}} \tag{30}$$

Note that the derivatives, such as $\left(\frac{\partial(\cdot)_v}{\partial T}\right)_{P,\mathbf{y}}$, $\left(\frac{\partial(\cdot)_v}{\partial P}\right)_{T,\mathbf{y}}$, and $\left(\frac{\partial(\cdot)_v}{\partial y_k}\right)_{P,T,y_{s,s\neq k}}$, are single-phase derivatives, and no phase change calculation is involved. These terms can be directly obtained by taking derivatives of the analytical form of EOS (including the departure functions derived from EOS [77]) and NASA polynomials.

In TP flash, the thermodynamic properties are considered as functions of mixture temperature T, pressure P, and composition $\mathbf{z}$. The first-order derivatives are required by TP flash calculation (e.g., using the Newton-Raphson iteration method) and also the ISAT of TP flash solutions (e.g., evaluating the Jacobian matrix $\left.\frac{\partial \mathbf{F}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0} = \left.\frac{\partial(\rho,e,\beta,c)}{\partial(P,T,\mathbf{z})}\right|_{\mathbf{x}_0}$).

First, the derivative of vapor mole fraction β is formulated as (details are shown in Appendix A):

$$\left(\frac{\partial \beta}{\partial T}\right)_{P,\mathbf{z}} = \sum_{k=1}^{N}\left(\frac{\partial v_k}{\partial T}\right)_{P,\mathbf{z}}, \left(\frac{\partial \beta}{\partial P}\right)_{T,\mathbf{z}} = \sum_{k=1}^{N}\left(\frac{\partial v_k}{\partial P}\right)_{T,\mathbf{z}}, \left(\frac{\partial \beta}{\partial z_i}\right)_{T,P,z_{s,s\neq i}} = \sum_{j=1}^{N}\left(\frac{\partial N_{j,v}}{\partial N_i}\right)_{T,P,N_{s,s\neq j}} - \beta, \tag{31}$$

where v_k is the mole fraction of component k in the vapor phase with respect to the mixture (i.e., $v_k = \beta y_k$), N_i is the total number of moles of component i in the mixture, and $N_{k,v}$ is the number of moles of component k in vapor phase. The derivative of ρ and e are obtained from Eq. (6) and Eq. (7):

$$\left(\frac{\partial \rho}{\partial T}\right)_{P,\mathbf{z}} = -\frac{\rho^2}{M}\left\{\left(\frac{\partial \beta}{\partial T}\right)_{P,\mathbf{z}}\left(\frac{M_v}{\rho_v} - \frac{M_l}{\rho_l}\right) + \frac{\beta}{\rho_v^2}\left[\rho_v\left(\frac{\partial M_v}{\partial T}\right)_{P,\mathbf{z}} - M_v\left(\frac{\partial \rho_v}{\partial T}\right)_{P,\mathbf{z}}\right]\right. \tag{32}$$

$$\left. + \frac{1-\beta}{\rho_l^2}\left[\rho_l\left(\frac{\partial M_l}{\partial T}\right)_{P,\mathbf{z}} - M_l\left(\frac{\partial \rho_l}{\partial T}\right)_{P,\mathbf{z}}\right]\right\} \tag{33}$$

$$\left(\frac{\partial e}{\partial T}\right)_{P,\mathbf{z}} = \frac{1}{M}\left\{\left(\frac{\partial \beta}{\partial T}\right)_{P,\mathbf{z}}\left(M_v e_v - M_l e_l\right) + \beta\left[e_v\left(\frac{\partial M_v}{\partial T}\right)_{P,\mathbf{z}} + M_v\left(\frac{\partial e_v}{\partial T}\right)_{P,\mathbf{z}}\right]\right. \tag{34}$$

$$\left. + (1-\beta)\left[e_l\left(\frac{\partial M_l}{\partial T}\right)_{P,\mathbf{z}} + M_l\left(\frac{\partial e_l}{\partial T}\right)_{P,\mathbf{z}}\right]\right\} \tag{35}$$

The derivatives of single phase properties, such as $\left(\frac{\partial e_v}{\partial T}\right)_{P,\mathbf{z}}$, $\left(\frac{\partial M_v}{\partial T}\right)_{P,\mathbf{z}}$, $\left(\frac{\partial \rho_v}{\partial T}\right)_{P,\mathbf{z}}$, can be calculated using Eqs. (27)-(28).

For the pressure derivatives, similar formulas can be used, with all the $\left(\frac{\partial(\cdot)}{\partial T}\right)_{P,\mathbf{z}}$ replaced by $\left(\frac{\partial(\cdot)}{\partial P}\right)_{T,\mathbf{z}}$.

Since $\left(\frac{\partial M}{\partial z_i}\right)_{P,T} \neq 0$, the z derivatives have different formulas:

$$\left(\frac{\partial \rho}{\partial z_i}\right)_{P,T} = -\frac{\rho^2}{M}\left\{\left(\frac{\partial \beta}{\partial z_i}\right)_{P,T}\left(\frac{M_v}{\rho_v} - \frac{M_l}{\rho_l}\right) + \frac{\beta}{\rho_v^2}\left[\rho_v\left(\frac{\partial M_v}{\partial z_i}\right)_{P,T} - M_v\left(\frac{\partial \rho_v}{\partial z_i}\right)_{P,T}\right]\right. \tag{36}$$

$$+\frac{1-\beta}{\rho_l^2}\left[\rho_l\left(\frac{\partial M_l}{\partial z_i}\right)_{P,T}-M_l\left(\frac{\partial \rho_l}{\partial z_i}\right)_{P,T}\right]-\frac{M_i-M}{\rho}\Bigg\} \tag{37}$$

$$\left(\frac{\partial e}{\partial z_i}\right)_{P,T}=\frac{1}{M}\left\{\left(\frac{\partial \beta}{\partial z_i}\right)_{P,T}\left(M_v e_v-M_l e_l\right)+\beta\left[e_v\left(\frac{\partial M_v}{\partial z_i}\right)_{P,T}+M_v\left(\frac{\partial e_v}{\partial z_i}\right)_{P,T}\right]\right. \tag{38}$$

$$\left.+(1-\beta)\left[e_l\left(\frac{\partial M_l}{\partial z_i}\right)_{P,T}+M_l\left(\frac{\partial e_l}{\partial z_i}\right)_{P,T}\right]-e\left(M_i-M\right)\right\} \tag{39}$$

The single phase properties derivatives, such as $\left(\frac{\partial e_v}{\partial z_i}\right)_{P,T}$, $\left(\frac{\partial M_v}{\partial z_i}\right)_{P,T}$, and $\left(\frac{\partial \rho_v}{\partial z_i}\right)_{P,T}$, can be calculated using Eqs. (29)-(30).

The last property, speed of sound c, is formulated as $c^2=1/(\kappa_s\rho)$, where κ_s is the isentropic compressibility: $\kappa_s=\kappa_T-T\alpha_p^2/\rho c_p$. The isobaric expansivity α_p and isothermal compressibility κ_T are defined as: $\alpha_p=-\left(\frac{\partial \rho}{\partial T}\right)_{P,\mathbf{z}}/\rho$ and $\kappa_T=\left(\frac{\partial \rho}{\partial P}\right)_{T,\mathbf{z}}/\rho$, respectively. Combining all these equations, the speed of sound c is formulated as:

$$c=\frac{1}{\sqrt{\left(\frac{\partial \rho}{\partial P}\right)_{T,\mathbf{z}}-\frac{T}{\rho^2}\left(\frac{\partial \rho}{\partial T}\right)_{P,\mathbf{z}}\Big/\left[\left(\frac{\partial e}{\partial T}\right)_{P,\mathbf{z}}-\frac{P}{\rho^2}\left(\frac{\partial \rho}{\partial T}\right)_{P,\mathbf{z}}\right]}}. \tag{40}$$

The speed of sound depends on the first derivatives, and hence, its derivatives require the computation of second derivatives, which need very complicated calculations. Because the derivative of sound speed is only used in the ISAT-VLE method to provide a better linear approximation, in fact, by setting its derivatives to zero, we can still use ISAT error control to obtain a sufficiently accurate solution, as we tested.

Based on the formulas of the $(T,P,\mathbf{z})$ system for the TP flash solver, the first derivatives of the $(\rho,P,\mathbf{z})$ system for the PV flash solver and the $(e,\rho,\mathbf{z})$ system for the UV flash solver can be derived using the chain rule.

Specifically, for the $(\rho,P,\mathbf{z})$ system, a property F is written as $F(\rho(T,P,z),P,z)$, and calculating temperature derivative using chain rule gives $\left(\frac{\partial F}{\partial \rho}\right)_{P,\mathbf{z}}\left(\frac{\partial \rho}{\partial T}\right)_{P,\mathbf{z}}=\left(\frac{\partial F}{\partial T}\right)_{P,\mathbf{z}}$. Hence, the density derivative is:

$$\left(\frac{\partial(\cdot)}{\partial \rho}\right)_{P,\mathbf{z}}=\left(\frac{\partial(\cdot)}{\partial T}\right)_{P,\mathbf{z}}\Big/\left(\frac{\partial \rho}{\partial T}\right)_{P,\mathbf{z}} \tag{41}$$

To derive the pressure and z_i derivatives, $\left(\frac{\partial T}{\partial P}\right)_{\rho,\mathbf{z}}$ is needed. Taking pressure derivative of constant density equation $\rho(T(P,z),P,z)=Const$ gives $\left(\frac{\partial T}{\partial P}\right)_{\rho,\mathbf{z}}\left(\frac{\partial \rho}{\partial T}\right)_{P,\mathbf{z}}+\left(\frac{\partial \rho}{\partial P}\right)_{T,\mathbf{z}}=0$, and hence, $\left(\frac{\partial T}{\partial P}\right)_{\rho,\mathbf{z}}=-\left(\frac{\partial \rho}{\partial P}\right)_{T,\mathbf{z}}/\left(\frac{\partial \rho}{\partial T}\right)_{P,\mathbf{z}}$. Taking pressure and z_i derivative of $F(\rho(T,P,z),P,z)$ and substituting $\left(\frac{\partial T}{\partial P}\right)_{\rho,\mathbf{z}}$ into the results give:

$$\left(\frac{\partial(\cdot)}{\partial P}\right)_{\rho,\mathbf{z}}=-\left(\frac{\partial(\cdot)}{\partial T}\right)_{P,\mathbf{z}}\left(\frac{\partial \rho}{\partial P}\right)_{T,\mathbf{z}}\Big/\left(\frac{\partial \rho}{\partial T}\right)_{P,\mathbf{z}}+\left(\frac{\partial(\cdot)}{\partial P}\right)_{T,\mathbf{z}} \tag{42}$$

$$\left(\frac{\partial(\cdot)}{\partial z_i}\right)_{P,\rho}=-\left(\frac{\partial(\cdot)}{\partial T}\right)_{P,\mathbf{z}}\left(\frac{\partial \rho}{\partial z_i}\right)_{T,P,z_{j,j\neq i}}\Big/\left(\frac{\partial \rho}{\partial T}\right)_{P,\mathbf{z}}+\left(\frac{\partial(\cdot)}{\partial z_i}\right)_{T,P,z_{j,j\neq i}} \tag{43}$$

For $(e,\rho,\mathbf{z})$ system, because the Jacobian matrix of $(e,\rho,\mathbf{z})$ and $(T,P,\mathbf{z})$ has the relation $\frac{\partial(e,\rho,\mathbf{z})}{\partial(T,P,\mathbf{z})}\frac{\partial(T,P,\mathbf{z})}{\partial(e,\rho,\mathbf{z})}=I$, where I is the identity matrix. The derivative of any property F in $(e,\rho,\mathbf{z})$ system can be obtained by $\frac{\partial F}{\partial(e,\rho,\mathbf{z})}=\frac{\partial F}{\partial(T,P,\mathbf{z})}\frac{\partial(T,P,\mathbf{z})}{\partial(e,\rho,\mathbf{z})}=\frac{\partial F}{\partial(T,P,\mathbf{z})}\left[\frac{\partial(e,\rho,\mathbf{z})}{\partial(T,P,\mathbf{z})}\right]^{-1}$. Calculating the inverse matrix gives:

$$\Delta=\left(\frac{\partial e}{\partial T}\right)_{P,\mathbf{z}}\left(\frac{\partial \rho}{\partial P}\right)_{T,\mathbf{z}}-\left(\frac{\partial e}{\partial P}\right)_{T,\mathbf{z}}\left(\frac{\partial \rho}{\partial T}\right)_{P,\mathbf{z}} \tag{44}$$

$$\left(\frac{\partial(\cdot)}{\partial e}\right)_{\rho,\mathbf{z}}=\frac{1}{\Delta}\left[\left(\frac{\partial(\cdot)}{\partial T}\right)_{P,\mathbf{z}}\left(\frac{\partial \rho}{\partial P}\right)_{T,\mathbf{z}}-\left(\frac{\partial(\cdot)}{\partial P}\right)_{T,\mathbf{z}}\left(\frac{\partial \rho}{\partial T}\right)_{P,\mathbf{z}}\right] \tag{45}$$

$$\left(\frac{\partial(\cdot)}{\partial \rho}\right)_{e,\mathbf{z}}=\frac{1}{\Delta}\left[-\left(\frac{\partial(\cdot)}{\partial T}\right)_{P,\mathbf{z}}\left(\frac{\partial e}{\partial P}\right)_{T,\mathbf{z}}+\left(\frac{\partial(\cdot)}{\partial P}\right)_{T,\mathbf{z}}\left(\frac{\partial e}{\partial T}\right)_{P,\mathbf{z}}\right] \tag{46}$$

$$\left(\frac{\partial(\cdot)}{\partial z_i}\right)_{e,\rho,z_{j,j\neq i}}=\left(\frac{\partial(\cdot)}{\partial z_i}\right)_{T,P,z_{j,j\neq i}}+\left(\frac{\partial(\cdot)}{\partial T}\right)_{P,\mathbf{z}}\left(\frac{\partial T}{\partial z_i}\right)_{e,\rho,z_{j,j\neq i}}+\left(\frac{\partial(\cdot)}{\partial P}\right)_{T,\mathbf{z}}\left(\frac{\partial P}{\partial z_i}\right)_{e,\rho,z_{j,j\neq i}} \tag{47}$$

$$\left(\frac{\partial T}{\partial z_i}\right)_{e,\rho,z_{j,j\neq i}}=\frac{1}{\Delta}\left[-\left(\frac{\partial \rho}{\partial P}\right)_{T,\mathbf{z}}\left(\frac{\partial e}{\partial z_i}\right)_{e,\rho,z_{j,j\neq i}}+\left(\frac{\partial e}{\partial P}\right)_{T,\mathbf{z}}\left(\frac{\partial \rho}{\partial z_i}\right)_{e,\rho,z_{j,j\neq i}}\right] \tag{48}$$

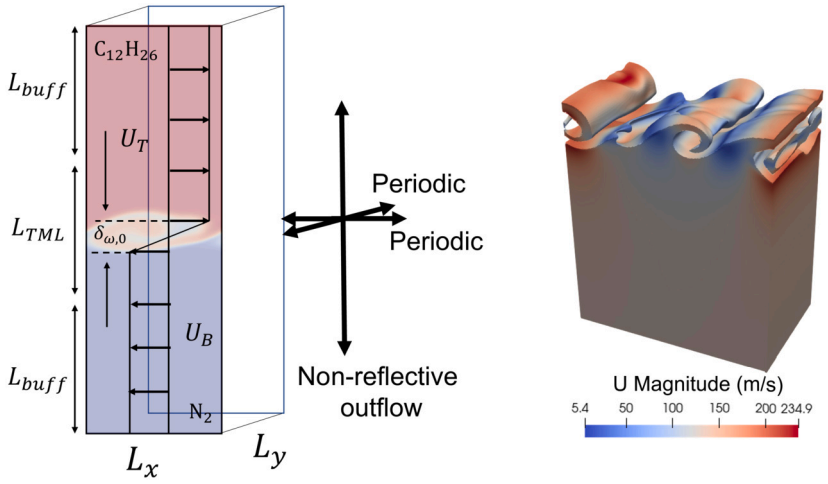

Fig. 8. Left: Configuration of the transcritical temporal mixing layer (TML) simulations. Right: 3D VLE-based CFD simulation of the transcritical TML at $t = 2 \times 10^{-7}$ s; iso-surface: mass fraction of n-dodecane $Y_{C_{12}H_{26}} = 0.3$; color: velocity magnitude.

$$\left(\frac{\partial P}{\partial z_i}\right)_{e,\rho,z_{j,j\neq i}} = \frac{1}{\Delta}\left[\left]\left(\frac{\partial \rho}{\partial T}\right)_{T,\mathbf{z}}\left(\frac{\partial e}{\partial z_i}\right)_{e,\rho,z_{j,j\neq i}} - \left(\frac{\partial e}{\partial T}\right)_{T,\mathbf{z}}\left(\frac{\partial \rho}{\partial z_i}\right)_{e,\rho,z_{j,j\neq i}}\right] \tag{49}$$

3. Results and discussion

We have integrated the aforementioned methods to develop an ISAT-VLE transcritical flow solver. In this section, we evaluate the performance and error control of the ISAT-VLE method through two sets of simulations: transcritical temporal mixing layer (Sec. 3.1) and transcritical shock-droplet interaction (Sec. 3.2). These scenarios are related to important applications. The temporal mixing layer often appears in shearing layers of jet flows and shock-droplet interaction is involved in liquid-fueled detonation modeling [78]. These simulations are still laminar flow in our scope of study, so the subgrid-scale models for turbulence are not used. Both 3D and 2D settings are utilized for these simulations. The 2D simulations (by sequential computing) primarily serve as means to test the ISAT-VLE method, as parallel computing introduces additional factors such as thread synchronization and inter-node communication, which can hinder an accurate assessment of ISAT-VLE's true performance. Furthermore, the 2D results can provide a suitable reference for determining the error tolerance in the 3D simulations. For the 3D configurations, we discuss the phase separation phenomena at transcritical conditions, along with demonstrating the tabulation error and ISAT-VLE performance. Furthermore, in the transcritical shock-droplet interaction simulations (Sec. 3.2), we compare the behavior of the fully compressible (FC) and double flux scheme (DF) schemes, both with ISAT-VLE. Additionally, redundant record deletion methods are also evaluated in Sec. 3.2.3.

3.1. Transcritical temporal mixing layer (TML)

In this section, a series of high-pressure transcritical temporal mixing layer (TML) cases are simulated using our VLE-based CFD solver. The simulations are conducted to investigate the ISAT-VLE method performance, and only the FC scheme is used for simulations. The schematic of the transcritical TML configuration is shown in Fig. 8. The computational domain is a rectangular box of size $L_x \times L_y \times L_z$. The upper half is filled with n-dodecane (n-$C_{12}H_{26}$) which flows to the right with the velocity U_T, and the lower half is filled with N_2 which flows to the left with the velocity U_B. In the vertical z direction, the domain is split into 3 subdomains: $L_z = L_{buff} + L_{TML} + L_{buff}, L_{buff} = L_{TML} = 1/3L_z$. The focus of this simulation is on TML evolution in the center cubic subdomain $L_x \times L_y \times L_{TML}$. The center subdomain is uniformly discretized in all directions (256 × 152 × 256), and the top and bottom subdomain have a stretched grid along the z direction (256 × 152 × 64; aspect ratio near the center subdomain: 1.01; aspect ratio at top and bottom boundary: 10.1). The configuration is the same as that in Miller et al. [79]. The initial cross-stream mean profiles, including velocity, temperature, and mass fractions, are set using an error function profile: $\mathrm{erf}(\sqrt{\pi}z/\delta_{\omega,0})$, where $\delta_{\omega,0}$ is the initial vorticity layer thickness. The stream-wise velocity is determined using the condition of stationary vortices (Eq. (50)), which is derived by Miller et al. [79] to obtain stationary vortices in TML simulation. Since the condition of stationary vortices is derived under the assumption of calorically perfect gases, vortices do not stay truly stationary for real-fluid simulations.

$$U_T = \frac{2M_c c_T}{1+\sqrt{\gamma_T/\gamma_B}}, U_B = -U_T \frac{c_B}{c_T}\sqrt{\gamma_T/\gamma_B}, \tag{50}$$

where M_c is the convective Mach number, and M_c is fixed to 0.3 in the present research; γ and c are the heat capacity ratio and speed of sound, respectively. The simulations are conducted on a domain with $L_x = 4\lambda_x = 29.16\delta_{\omega,0}$, $L_y = 0.6L_x$, and $L_{TML} = L_x$. The initial Reynolds number is defined as $Re_0 = [0.5(\rho_U + \rho_L)\Delta_{U_0}\delta_{\omega,0}]/\mu, \mu = 0.5(\mu_T + \mu_B)$.

Table 1
The initial condition of the transcritical temporal mixing layer (TML).

Properties	U_T (m/s)	U_B (m/s)	T_T (K)	L_x (m)	$\delta_{\omega,0}$
Case 1	77.94	134.27	600	2.79×10^{-5}	9.58×10^{-7}
Case 2	59.67	89.01	800	3.55×10^{-5}	1.22×10^{-6}

Subscripts "T" and "B" refer to Top and Bottom, respectively. $L_y = 0.6\times L_x$, and $Re_0 = 1000$. Mesh in the center subdomain is $256\times152\times256$, and the top and bottom subdomain mesh is $256\times152\times64$. $P = 50$ bar, $T_B = 293$ K, $M_c = 0.3$, $A_i = 0.25, 0.5, 1$, $B_i = 0.05, 0, 1$, and $F_{2D} = F_{3D} = 0.05$.

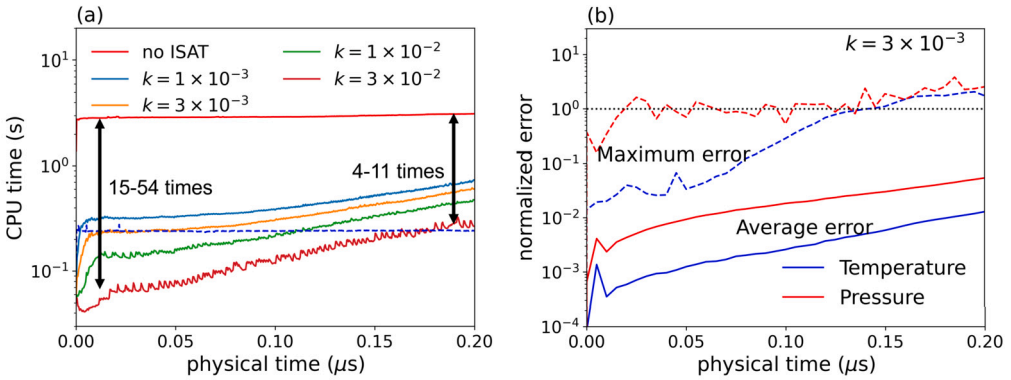

Fig. 9. (a) The performance of the ISAT-VLE method, in terms of the CPU time spent at every time step. The Blue dotted line is the CPU time of all other parts of the CFD solver, except the VLE model; the top red line is the CPU time of the VLE model without ISAT. In addition, 4 lines of the ISAT-VLE model with different error tolerances (k) are provided. (b) The error control of the ISAT-VLE method, in terms of the normalized average error and the maximum error versus physical time, with $k = 3\times10^{-3}$. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

The TML configuration sets periodic boundary conditions along the streamwise (x) and spanwise (y) direction, and the non-reflective outflow condition is used along the crosswise (z) direction (in Fig. 8). A velocity perturbation is superimposed to trigger the instability, which was derived in Masi et al. [80]. The velocity perturbation in the x-z plane is:

$$a_1 = \frac{1}{4}e^{\pi(\delta_{\omega,0}/\lambda_{xi})^2}(\mathrm{erf}(\sqrt{\pi}(z/\delta_{\omega,0} + \delta_{\omega,0}/\lambda_{xi})) - 1);$$

$$a_2 = -\frac{1}{4}e^{\pi(\delta_{\omega,0}/\lambda_{xi})^2}(\mathrm{erf}(\sqrt{\pi}(z/\delta_{\omega,0} - \delta_{\omega,0}/\lambda_{xi})) + 1);$$

$$u_x = F_{2D}\Delta U\sum_i A_i sin(2\pi x/\lambda_{xi})(-a_1 e^{2\pi z/\lambda_{xi}} + a_2 e^{-2\pi z/\lambda_{xi}});$$

$$u_z = F_{2D}\Delta U\sum_{i=0}^{2} A_i cos(2\pi x/\lambda_{xi})(-a_1 e^{2\pi z/\lambda_{xi}} + a_2 e^{-2\pi z/\lambda_{xi}}),$$

where $\Delta U = |U_T - U_B|$, and $\lambda_{xi} = 2^i\lambda_x$. F_{2D} and A_i are coefficients to determine the perturbation magnitude [80]. The velocity perturbation in the y-x plan uses the same form, but replacing λ_x by $\lambda_y = 0.6\lambda_x$. Its perturbation magnitude (z direction) are controlled using F_{3D} and B_i. The initial conditions for the two simulation cases in the present study are shown in Table 1. Two cases use the same Re_0, M_c, and T_B. In Case 2, the initial n-$C_{12}H_{26}$ temperature (800 K) is higher than Case 1 (600 K). The other configurations, including U_T, U_B, L_x, and $\delta_{\omega,0}$, are obtained from the Eq. (50), and the formulas of Re_0 and L_x mentioned above.

3.1.1. Speed-up and error control of the ISAT-VLE method in 2D TML simulations

In this section, we utilize the configuration of Case 1 in Table 1. For 2D simulation, the velocity in the y direction is set to zero. A set of error tolerance values are employed to assess the ISAT-VLE method performance. Using the FC scheme, the ISAT-VLE method provides pressure, temperature, vapor fraction, and speed of sound. The parameter k is utilized to adjust the error tolerance, where the error tolerance $(T_{tol}, P_{tol}, \beta_{tol}, c_{tol}) = k(10^2, 10^6, 1, 10^2)$. This error tolerance is defined based on the order of the magnitude of the properties, and hence k can be roughly considered as a maximum relative error, in which the order of magnitude of the tolerances depends on the magnitude of the properties. Hence, k can be roughly considered as a maximum relative error. The simulation is run to 2×10^{-7} s to capture the entire vortex formation process starting from a flat interface under the influence of initial perturbation.

Fig. 9(a) illustrates the performance of the ISAT-VLE method. The blue dotted line represents the CPU time consumed by all other models in the CFD solver, which is significantly lower than the time consumed by the VLE model when ISAT method is not deployed. Without the ISAT method, the VLE model accounts for 92.9% of the computational resources. The other curves in Fig. 9(a) depict the accelerated performance of the VLE model by the ISAT method for different k values (error tolerances), where a larger tolerance yields higher speed-up. At the initial stage, the ISAT-VLE model runs approximately 15-54 times faster than the original VLE model, and in the final stage, it achieves a speed-up factor of 4-11. This is because the CPU time of all ISAT-VLE simulations increases

Table 2
The performance and errors of the ISAT-VLE method in the 2D transcritical TML cases with different error tolerances.

k	1×10^{-3}	3×10^{-3}	1×10^{-2}	3×10^{-2}
Speed-up	6.7	8.5	11.6	19.9
P mean error (bar)	8.2×10^{-4}	1.5×10^{-3}	7×10^{-4}	1.6×10^{-2}
P max error (bar)	0.06	0.07	0.27	1.0
P mean relative error (%)	3.2×10^{-3}	5.4×10^{-3}	3.1×10^{-3}	5.3×10^{-2}
P max relative error (%)	0.16	0.20	0.71	2.8
T mean error (K)	1.6×10^{-3}	3.5×10^{-3}	1.4×10^{-3}	4×10^{-2}
T max error (K)	0.16	0.6	2.1	6.4
T mean relative error (%)	4.2×10^{-4}	9.1×10^{-4}	3.2×10^{-4}	1.4×10^{-3}
T max relative error (%)	0.05	0.21	0.73	2.1

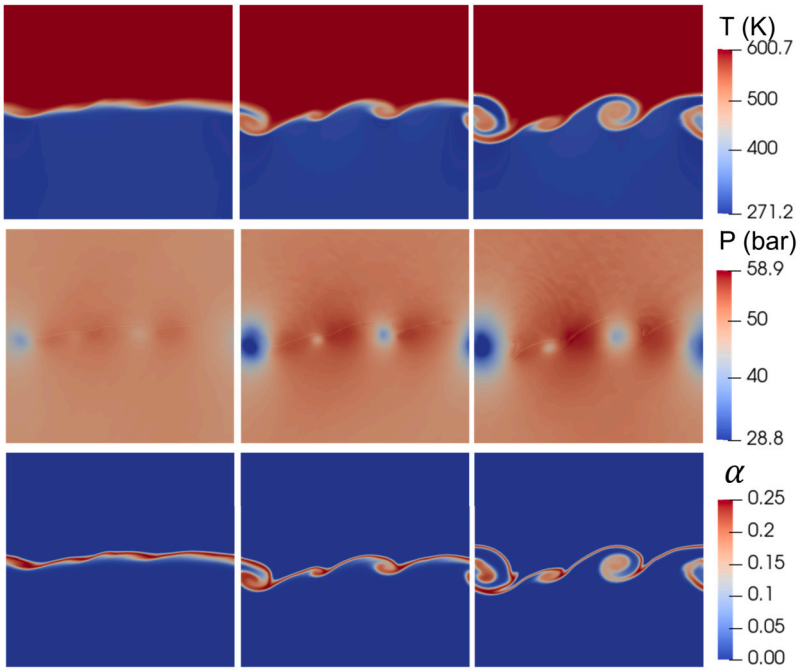

Fig. 10. The vortex development in the 3D transcritical temporal mixing layer (TML): three snapshots at $t = 1\times10^{-7}, 1.5\times10^{-7}, 2\times10^{-7}$ s (from left to right) are shown in the figure.

as the mixing process generates more and more thermodynamic states. The performance curves of the ISAT-VLE method exhibit oscillation, which is attributed to the re-balancing of the binary tree when the overall structure becomes highly unbalanced (see Sec. 2.3). Simulations with larger tolerances exhibit smaller CPU times spent at each time step, making the influence of re-balancing on performance (i.e., oscillation) more pronounced.

Fig. 9(b) shows the error control of the ISAT-VLE method. The errors presented in the plot are from the simulation with $k = 3\times10^{-3}$. The error is normalized by the tolerance, and the dashed line indicates that the error is equal to the tolerance (i.e., normalized error = 1). The average error is evaluated using $\frac{\sum_i |e_i|V_i}{\sum_i V_i}$, where V_i is the volume of cell i, and e_i is the error in cell i. With $k = 3\times10^{-3}$, the average error is controlled to be 2 to 3 orders of magnitude smaller than the tolerance. Although the maximum error can slightly exceed the tolerance, it remains within the same order of magnitude. Table 2 shows the absolute and relative errors for all cases. As the tolerance decreases, the error is better controlled. For this particular 2D simulation case, $k = 3\times10^{-3}$ is proven to be a suitable choice, ensuring pressure and temperature errors to be kept within 0.5% while achieving a total speed-up factor of 8.5.

3.1.2. 3D TML simulations using the ISAT-VLE method

As recommended in the preceding section, $k = 3\times10^{-2}$ is used for the 3D transcritical TML simulations. Simulations of Case 1 and Case 2 are conducted with the configurations detailed in Table 1. A grid convergence study is conducted using the configuration of Case 1, as shown in Appendix B.1. The development of vortices is depicted in Fig. 10. At $t = 1\times10^{-7}$ s, the introduction of velocity perturbation in the initial condition causes distortion of the interface. Due to the Kelvin-Helmholtz (K-H) instability, three vortices form on the interface at 1.5×10^{-7} s. Within the vortex center, a low-pressure area is generated, and acoustic waves propagate

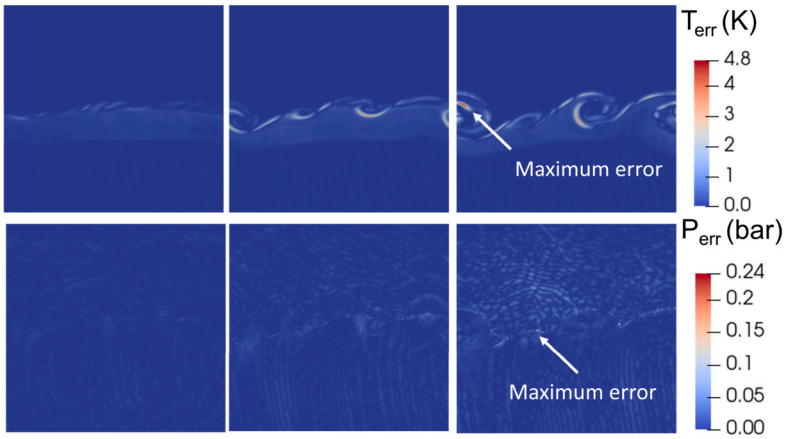

Fig. 11. The errors of the ISAT-VLE method in the 3D transcritical temporal mixing layer (TML): three snapshots at $t = 1 \times 10^{-7}, 1.5 \times 10^{-7}, 2 \times 10^{-7}$ s (from left to right) are shown in the figure.

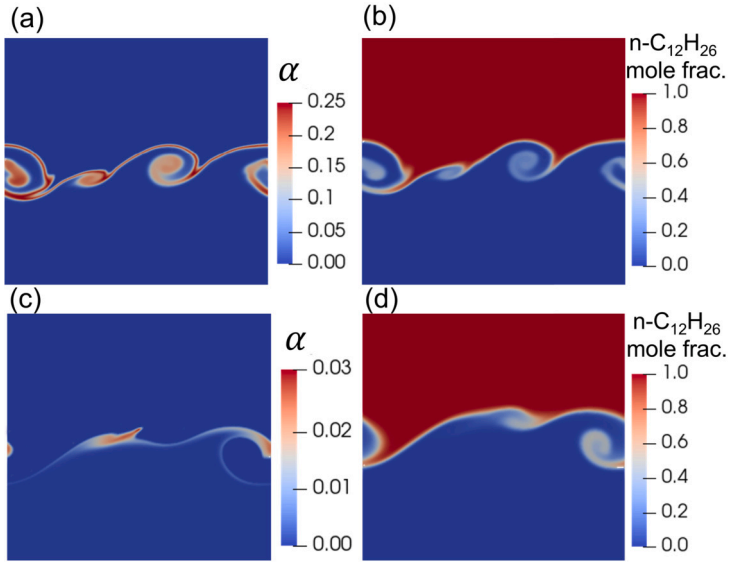

Fig. 12. 3D transcritical temporal mixing layer (TML) contours at $\tau = 45$ (in which $\tau = t\Delta U/\delta_{\omega,0}$). Top (a,b): Case 1; Bottom (c,d): Case 2. Left (a,c): α (which is the indicator of the two-phase interface: $\alpha = \beta(1-\beta)$, where β is vapor mole fraction); Right (b,d): mole fraction of n-$C_{12}H_{26}$.

outward continuously, which can be clearly seen in the pressure contours at 2×10^{-7} s. The quantity $\alpha = \beta(1-\beta)$ serves as an indicator of the two-phase interface and phase separation. The α contours reveal that at these conditions, phase separation is highly pronounced at the interface between n-$C_{12}H_{26}$ and N_2.

Fig. 11 shows the temporal evolution of pressure and temperature errors. These values are obtained by calculating the absolute difference between the results obtained with the ISAT-VLE method and those obtained without it (e.g., $|T_{ISAT} - T_{noISAT}|$). As shown in Fig. 11, both the temperature and pressure errors exhibit a gradual and slow increase over time. However, the overall error remains at a very low level. The temperature deviation is primarily concentrated at the interface, with a maximum deviation of less than 5 K and a relative error of less than 1%. The pressure error is more widespread due to the propagation of acoustic waves, but the majority of errors remain below 0.1 bar. The maximum error occurs at the interface and reaches approximately 0.2 bar. This error level is fully acceptable for the simulation, considering the average ambient pressure of around 50 bar, and the relative pressure error remains within 0.5%.

To compare the results of Case 1 and Case 2, a normalized time parameter $\tau = t\Delta U/\delta_{\omega,0}$ is employed. Fig. 12 compares the results at $\tau = 45$, corresponding to the formation of the vortex. In both cases, a two-phase region is observed at the interface, as indicated by the high α values. However, in Case 2, the phase separation (i.e., α value) is significantly weaker despite the presence of cold N_2 (293 K). This can be attributed to the higher temperature of n-$C_{12}H_{26}$ (800 K), which hinders its cooling by the cold N_2. As a result, even at the interface, the majority of mixture remains in the gas phase. However, in the vicinity of the vortex, where mixing is more pronounced, a slightly stronger phase separation is observed. The crucial role of the VLE model becomes apparent in capturing these differences in phase separation at different conditions.

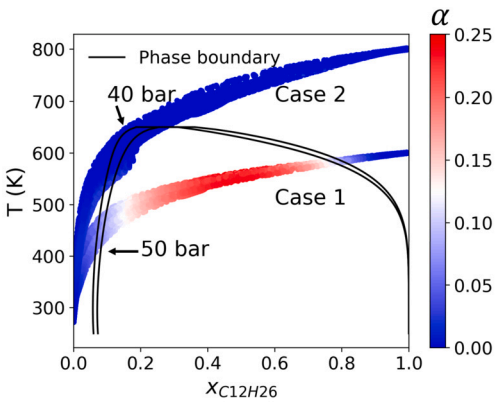

Fig. 13. The temperature-mole fraction (T-X) phase diagram of the transcritical temporal mixing layer (TML) simulation: the phase boundary of $N_2/C_{12}H_{26}$ at 40 and 50 bar is shown in the plot as a solid black curve.

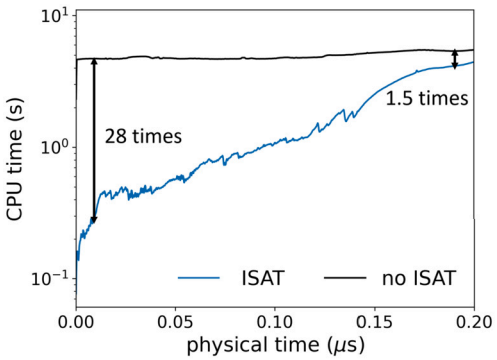

Fig. 14. Performance of the ISAT-VLE method in the 3D simulation of the transcritical temporal mixing layer (TML).

The distinction between Case 1 and Case 2 can be better understood by analyzing through a phase diagram. Since the pressure variation in the transcritical TML simulation is relatively small, a temperature-mole fraction (T-X) diagram is employed to compare the two cases. Fig. 13 shows the phase boundaries at 50 bar and 40 bar. It is important to note that these phase boundaries are not highly sensitive to changes in pressure. Thus, they serve as reliable reference lines for understanding the characteristics of the two-phase region. In Case 1, the lower initial temperature of n-$C_{12}H_{26}$ (600 K) causes the thermodynamic states of the mixing layer fluid ($X_{C_{12}H_{26}}$ within the range of 0.2-0.6) to overlap with the two-phase region, resulting in significant phase separation. In contrast, in Case 2, the higher temperature of n-$C_{12}H_{26}$ (800 K) leads to points with $X_{C_{12}H_{26}} > 0.3$ having temperatures above the two-phase boundary, thereby preventing phase separation in this region. Fig. 13 also illustrates that the temperature of the phase boundary increases rapidly within the range of $X_{C_{12}H_{26}} < 0.2$, making it difficult for the fluid in this region to remain outside the two-phase region. Consequently, Case 2 still exhibits weak phase separation near $X_{C_{12}H_{26}} = 0.2$. We further conducted tests with even higher initial n-$C_{12}H_{26}$ temperatures (up to 1200 K), resulting in further weakening of the phase separation, although it still persists.

Fig. 14 shows the performance of the ISAT-VLE method in the 3D simulation of Case 1. The simulation utilizes 128 CPU cores, and the CPU times presented in the figure represent the maximum time cost among all MPI processes at each time step. Initially, the ISAT-VLE method provides a speed-up of approximately 28 times. However, as the simulation progresses and more vortices form, more thermodynamic states are created, and hence the speed-up of the ISAT-VLE method is reduced. At the final time step, ISAT-VLE only achieves a speed-up of 1.5 times. This performance is very different from the one shown in Fig. 9(a). The discrepancy arises because the ISAT-VLE simulation is more expensive in the interface region than the pure component region, which is attributable to the increased dispersion of thermodynamic states in the interface region, necessitating more direct calculations of the VLE model. The distinct characteristics of different regions in the computational domain prevent the MPI processes from synchronizing effectively (i.e., a lot of MPI processes with smaller computation loads are waiting for those with larger loads). Consequently, the impact of data synchronization and communication between the MPI processes becomes more pronounced, leading to longer waiting times. Future work of dynamic loading-balanced ISAT-VLE method is likely to address this limitation. These challenges make it hard to accurately evaluate the true performance of the ISAT-VLE method. Moreover, these issues are not solely related to the ISAT-VLE method itself, but are also influenced by the overall CFD solver and hardware. Therefore, the 2D results in Fig. 9(a) provide a better reflection of the performance of the ISAT-VLE method.

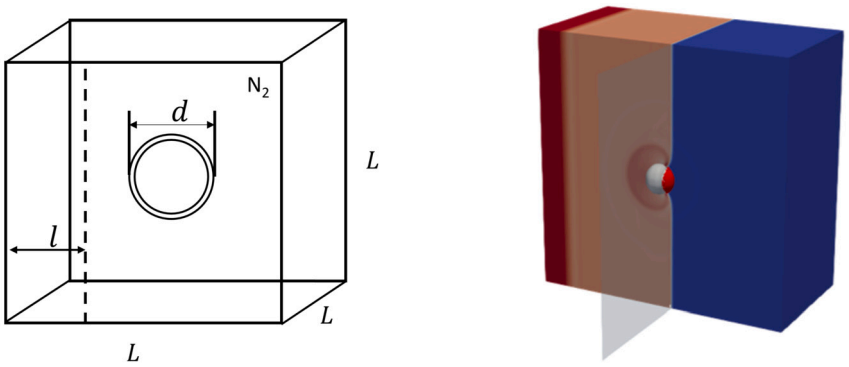

Fig. 15. Left: geometry of the shock-droplet interaction simulations. The computational domain is of squared shape with side $L = 2$ mm. A high-pressure region with thickness $l = 0.5$ mm is used to generate a shock wave. Droplet diameter: $d = 0.25$ mm. Right: pressure contours of the 3D shock-droplet interaction. Initial temperature: 620 K; Shock front: iso-surface of $P = 30$ MPa; White color on the droplet surface: iso-surface of $Y_{C_{12}H_{26}} = 0.7$; Red color on the droplet surface: iso-surface of $\beta = 0.9$ to visualize the subcritical two-phase interface.

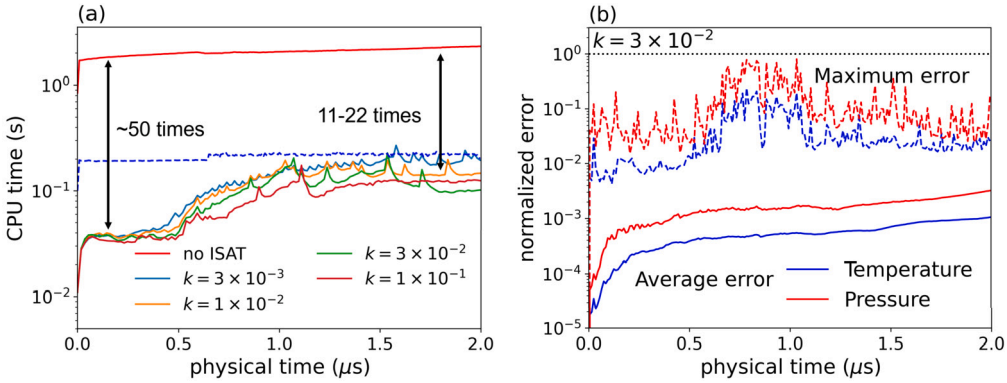

Fig. 16. (a) Performance of the ISAT-VLE method in the transcritical shock-droplet interaction simulation using the fully compressible (FC) scheme: the blue dotted line is the CPU time of all parts of the CFD solver other than the VLE model; the red line is the CPU time of VLE model without ISAT; and 4 lines of the ISAT-VLE model with different tolerances (k) are also provided. (b) The normalized error of the ISAT-VLE method with an error control of $k = 3 \times 10^{-2}$.

3.2. Transcritical shock-droplet interaction

In this section, transcritical shock-droplet interaction simulations are conducted to investigate the speed-up performance and error control of the ISAT-VLE method. The geometry of the initial setting is shown in Fig. 15, where $L = 2$ mm and $l = 0.5$ mm. A droplet of n-$C_{12}H_{26}$ with a diameter $d = 0.25$ mm is positioned at the center of the domain, and the domain is filled with N_2. The initial pressure is set to 200 bar, and a high-pressure region of 800 bar is positioned on the left to generate a shock wave. The interfacial mass fraction of N_2 is initialized using $\tanh(x/\omega)$, where $\omega = 2 \times 10^{-6}$. The performance of the ISAT-VLE method is tested through both 2D and 3D simulations. For the 2D simulations, a uniform mesh size of 256×256 is utilized in the x-y plane. For the 3D simulations, the z direction consists of 256 uniformly spaced grid intervals.

3.2.1. Speed-up and error control of the ISAT-VLE method in 2D shock-droplet interaction simulations

2D simulations are utilized to evaluate the error control of the ISAT-VLE method and select an appropriate error tolerance for later simulations. Throughout the entire computational domain, the initial temperature is set to 565 K, which is chosen specifically to examine the phase change effect in a multi-component droplet (please refer to Sec. 3.2.4). Initially, the fully conservative (FC) scheme results are presented. A range of error tolerance values is employed to assess performance, where the error tolerance $(T_{tol}, P_{tol}, \beta_{tol}, c_{tol})$ is scaled by a factor k: specifically, the error tolerance values are set as $k(10^2, 10^7, 1, 10^2)$. The magnitude of these tolerances is based on the observation in these simulations. The error tolerance cases run until a physical time of 1×10^{-6} s, at which point the shock wave has completely passed through the droplet. Fig. 16(a) illustrates the performance of the ISAT-VLE method. The top red line represents the CPU time cost of the VLE model without using the ISAT method at each time step, while the blue dotted line corresponds to the CPU time cost of the remaining parts of the CFD solver. Both lines are insensitive to time. Compared with the rest parts of the CFD solver, the VLE model without ISAT is computationally expensive, consistently utilizing around 90% of the computational resources throughout the simulation. Prior to the interaction between the shock wave and the droplet, the computational costs of all ISAT-VLE cases with varying tolerances are similar. This is attributed to the simple thermodynamic states involved, where a limited data set in the ISAT-VLE table is sufficient to provide accurate approximations. Therefore, in this stage, the ISAT-VLE method exhibits excellent performance, with a speed-up factor of approximately 50, and the error tolerance has minimal impact on

Table 3
The speed-up performance and errors of the ISAT-VLE method in the transcritical shock-droplet interaction cases with different error tolerances.

k	3×10^{-3}	1×10^{-2}	3×10^{-2}	1×10^{-1}
Speed-up	15.3	17.9	20.9	23.8
P mean error (bar)	2.4×10^{-4}	1.8×10^{-3}	9.6×10^{-3}	8.8×10^{-2}
P max error (bar)	0.26	0.86	2.4	7.7
P mean relative error (%)	6.5×10^{-5}	5.1×10^{-4}	4.5×10^{-3}	3.7×10^{-2}
P max relative error (%)	0.07	0.2	0.57	1.9
T mean error (K)	1.0×10^{-4}	6.4×10^{-4}	3.1×10^{-3}	3.5×10^{-2}
T max error (K)	0.074	0.17	0.71	2.2
T mean relative error (%)	2.1×10^{-5}	6.7×10^{-5}	4.0×10^{-4}	3.3×10^{-3}
T max relative error (%)	0.012	0.029	0.12	0.36

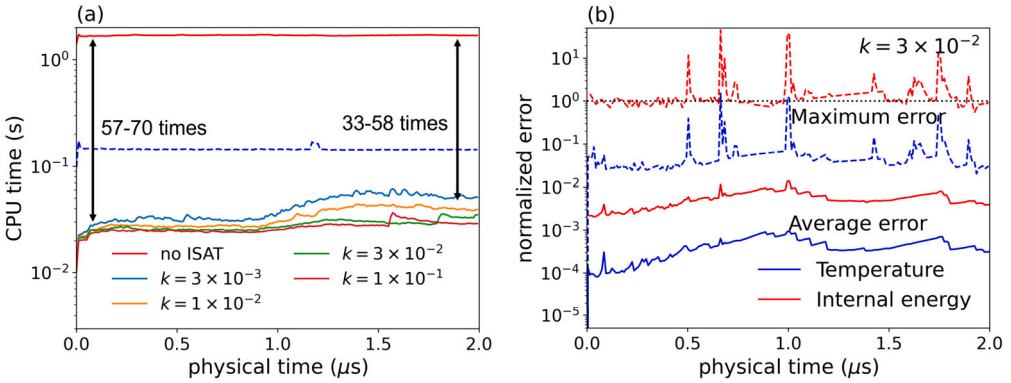

Fig. 17. (a) Performance of the ISAT-VLE method in the transcritical shock-droplet interaction simulation using the double flux (DF) scheme: the blue dotted line is the CPU time of all parts of CFD solver other than the VLE model; the red line is the CPU time of VLE model without ISAT; and 4 lines of the ISAT-VLE model with different tolerances (k) are provided. (b) The normalized error of the ISAT-VLE method with an error control of $k=3\times10^{-2}$.

the performance. As the shock-droplet interaction commences, the computational cost begins to increase, and the influence of the error tolerance becomes evident. Cases with larger tolerances demonstrate faster computation, and by a physical time of 1.5×10^{-6} s, when the droplet has passed through the shock wave, the CPU times stabilize. At this stage, the case with the largest tolerance is approximately 22 times faster than the VLE model without ISAT, while even the slowest case is still around 11 times faster.

The normalized ISAT error for the case with $k=3\times10^{-2}$ is depicted in Fig. 16(b). The error is normalized by its tolerance, and the dashed line represents an error equal to the normalized tolerance (i.e., normalized error $=1$), below which the error is controlled within the specified tolerance. The average error is three orders of magnitude smaller than the tolerance, and the maximum error is also within the tolerance. Table 3 presents the errors and speed-up factors for all cases. Considering the initial pressure range (200 - 800 bar) and temperature (565 K) settings, the errors are acceptable, staying within 2% for all different tolerances. For subsequent shock-droplet simulations, we choose $k=2\times10^{-3}$, which maintains the error within 1% and provides sufficient accuracy.

3.2.2. Behavior of double flux (DF) scheme

We proceeded to perform a simulation using the double flux (DF) scheme to compare its behavior with the FC scheme. The ISAT-VLE error tolerance is set as $(T_{tol}, e_{tol}, \beta_{tol}, c_{tol}) = k(10^2, 10^4, 1, 10^2)$. Similar to the FC method, the speed-up performance improves when a larger tolerance is employed (see Fig. 17(a)), but in comparison, the DF scheme runs even faster. Fig. 17(b) displays the normalized error of the ISAT-VLE method using $k=3\times10^{-2}$. It can be observed that the average error is controlled three orders of magnitude below the error tolerance. Although the maximum error cannot be entirely controlled within the tolerance due to the high non-linearity of the VLE system, the overall error remains within an acceptable range. The error and speed-up performance are listed in Table 4. When compared to the FC solver, the ISAT-VLE implementation in the DF solver achieves larger speed-up factors ranging from 40 to 60, and exhibits significantly smaller pressure errors but larger temperature errors. This discrepancy arises because, in the DF method, the pressure is directly obtained from the CFD solver, and its error is only influenced by the tabulation error of other properties, resulting in minimal pressure error. However, the DF scheme introduces stronger temperature oscillations (as depicted in Fig. 18 and discussed below), which contribute to an increased temperature error, which is a potential issue for reacting flow simulations with strong temperature dependencies. Additionally, the DF scheme effectively suppresses pressure oscillations, resulting in more concentrated thermodynamic states and improved efficiency of tabulation. As a result, we achieve a larger speed-up factor in our calculations.

To mitigate the spurious oscillation resulting from real-fluid effects, quasi-conservative methods are commonly employed. One popular approach among them is the DF method. However, it is important to note that these methods come at the cost of breaking the energy conservation law. In Fig. 18, a comparison is made between the DF and FC results in terms of pressure and temperature along the centerline of the 2D simulation of transcritical shock-droplet interaction in the x direction. This comparison is conducted

Table 4
The speed-up performance and error of the ISAT-VLE method using the double flux (DF) scheme in the transcritical shock-droplet interaction cases with different error tolerances.

k	3×10^{-3}	1×10^{-2}	3×10^{-2}	1×10^{-1}
Speed-up	41.1	50.0	60.9	63.9
P mean error (bar)	6.7×10^{-5}	2.2×10^{-4}	2.4×10^{-4}	1.2×10^{-3}
P max error (bar)	3.7×10^{-3}	7.4×10^{-3}	8.4×10^{-3}	5.2×10^{-2}
P mean relative error (%)	4.7×10^{-6}	1.3×10^{-5}	1.6×10^{-5}	7.4×10^{-5}
P max relative error (%)	0.001	0.002	0.003	0.016
T mean error (K)	3.9×10^{-3}	1.1×10^{-3}	2.8×10^{-3}	7.5×10^{-3}
T max error (K)	0.14	4.43	4.52	9.39
T mean relative error (%)	6.2×10^{-4}	1.3×10^{-4}	4.1×10^{-4}	1.0×10^{-3}
T max relative error (%)	0.02	0.76	0.78	1.5

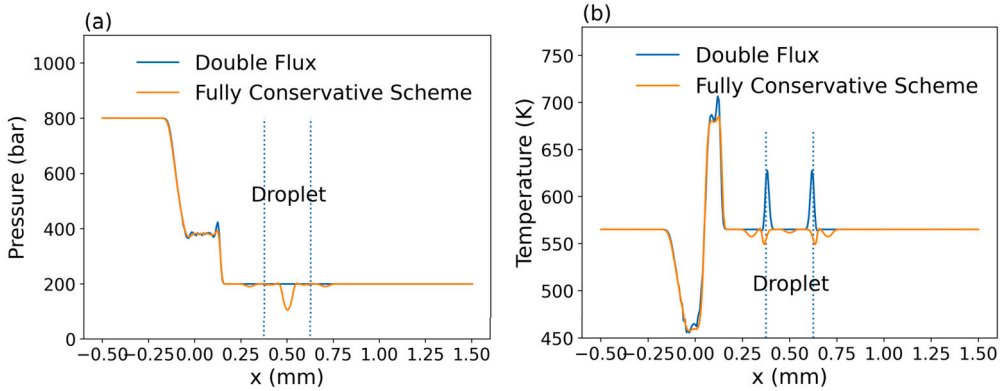

Fig. 18. Comparison between the fully conservative (FC) and double flux (DF) schemes: (a) pressure, and (b) temperature. The results show the properties at the horizontal center line along the x-direction through the droplet center. The physical time of the results is $t = 2\times10^{-7}$ s.

using the results at $t = 2\times10^{-7}$ s when the shock wave has not yet reached the droplet position, allowing for a separate analysis of the effects of the two methods on the shock wave and droplet. Without the DF method, the real-fluid EOS generates pressure and temperature oscillations near the droplet interface and inside the droplet. In contrast, the DF method successfully eliminates pressure oscillations at those locations, which is the intended goal of DF scheme's design. However, in order to achieve this, energy is added/removed to the contact discontinuity (i.e., the droplet interface), resulting in larger temperature oscillations near the droplet interface. Additionally, breaking the energy conservation law also introduces larger oscillations near the shock wave front. While some researchers argue that the energy introduced by the DF method decreases as the mesh is refined, it is often impractical to use a fine enough mesh to solve the droplet interface for 2D or 3D simulations. Therefore, breaking the energy conservation law (e.g., the DF scheme) is not considered as an optimal solution.

When the pressure oscillation in the simulation is substantial enough to cause the solution to diverge, it is preferable to allow for larger temperature oscillations while eliminating pressure oscillations. In addition, in simulations where temperature does not play a crucial role, the DF scheme can also be a suitable option. Conversely, when the simulation results are highly sensitive to temperature (e.g., transcritical combustion), the FC scheme is a better choice. This is precisely why we predominantly utilize the FC methods in our simulations, as the VLE process is particularly sensitive to temperature at the interface.

3.2.3. Performance of the redundant ISAT record deletion methods

In Sec. 2.3, we introduced new methods for deleting redundant ISAT records. In this section, we assess their performance through the 2D transcritical shock-droplet interaction simulations using the FC scheme. Fig. 19 illustrates the test results. First, we examined the outcomes without active data removal (FS: fixed maximum table size). Three different maximum table sizes were employed (FS1: 140,000, FS2: 130,000, FS3: 120,000). It can be observed that the FS3 result is slower compared to FS1 and FS2. This is because the maximum table size in FS3 is insufficient to store an adequate number of records, leading to more direct calculation of VLE model and thus reducing the speed-up performance. Once the table size is adequately large, further increasing the maximum size no longer impacts the speed-up performance. Hence, the results for FS1 and FS2 are close to the limits of this method.

Next, we conducted separate tests for our proposed methods: the maximum unused step method (MUS) and the adaptive maximum table size method (AS). In MUS, we actively eliminate data that has not been used within the last 80 time steps. In AS, we utilized parameters $C = 5.8$ and $M = 5$ to update the maximum table size (by Eq. (25)). Both methods resulted in reduced CPU time and achieved approximately a 4% performance improvement compared to FS2. When both methods were used concurrently (AS+MUS), the performance was further enhanced by approximately 2%. It is worth noting that the ISAT-VLE method itself has already undergone substantial optimization, effectively utilizing computational resources. However, we identified additional op-

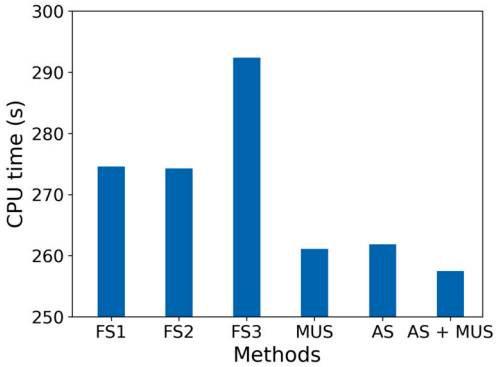

Fig. 19. Comparison between redundant ISAT record deletion methods. The fixed maximum table size method (FS) only removes data when the table is full, and the least recently used record is removed. The maximum table sizes for the 3 settings are FS1: 140,000, FS2: 130,000, and FS3: 120,000. The maximum unused step method (MUS) removes records that have not been used for a given number of time steps. 80 time steps are used for MUS. The adaptive maximum table size method (AS) uses Eq. (25) to determine maximum table size with $C = 5.8, M = 5$.

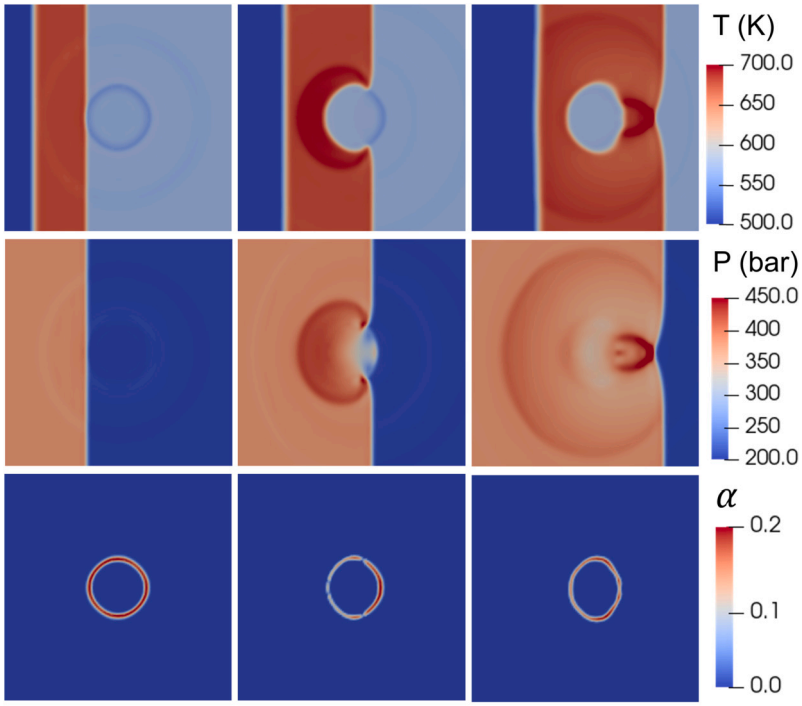

Fig. 20. 3D transcritical shock-droplet interaction simulation with an initial temperature of 565 K: the top figures are density, and the bottom figures are α, where $\alpha = \beta(1-\beta)$ is a two-phase interface indicator. From left to right are the time instants of 5×10^{-7} s, 8.3×10^{-7} s, and 1.2×10^{-6} s.

timization opportunities in Sec. 2.3, and our methods still yield a 6% performance improvement by the AU+MUS method. This AS+MUS method was employed for all simulations in this study.

3.2.4. 3D simulations of transcritical shock-droplet interaction using the ISAT-VLE method

In this section, we perform a series of 3D transcritical shock-droplet interaction simulations using the FC scheme. A grid convergence study is conducted, and the results are shown in Appendix B.2. Fig. 20 displays the 3D results at three different time instances: 5×10^{-7} s, 8.3×10^{-7} s, and 1.2×10^{-6} s, illustrating the propagation of the shock wave passing through the droplet. The results show the reflection wave and the deformation of the droplet induced by the shock wave. To gain a better understanding of the phase change, the thermodynamic state is plotted in the pressure-temperature (P-T) phase diagram (Fig. 21). In Fig. 21, the thermodynamic state is from the results at $t = 1.2 \times 10^{-6}$, when shock wave has passed droplet completely, to understand post-shock phenomenon. Since our focus is solely on the shock wave front and the interaction between the shock and droplet, data points corresponding to the expansion wave are excluded. In Fig. 21, all data points can be clearly categorized into two distinct parts. The first part consists of points forming a curve in the lower right region, which represents the shock front connecting the pre-shock and post-shock ambient conditions. The second part comprises the remaining points, forming a band that corresponds to the droplet. This is attributed to the

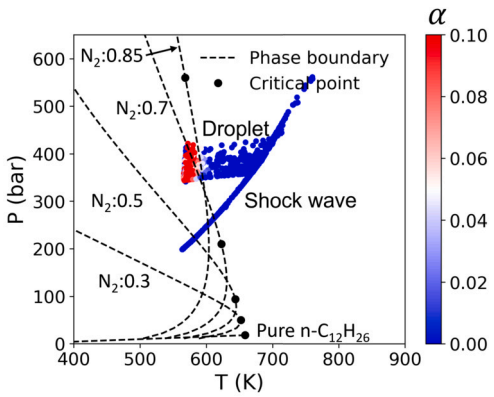

Fig. 21. Phase diagram of the 3D transcritical shock-droplet interaction simulation with an initial temperature 565 K. The labels are the mole fractions of N_2. The data points are taken from the time instance of 1.2×10^{-6} s, when the shock wave has passed the droplet completely. The data points of the expansion wave are removed. The labels in the figure are mole fractions of N_2.

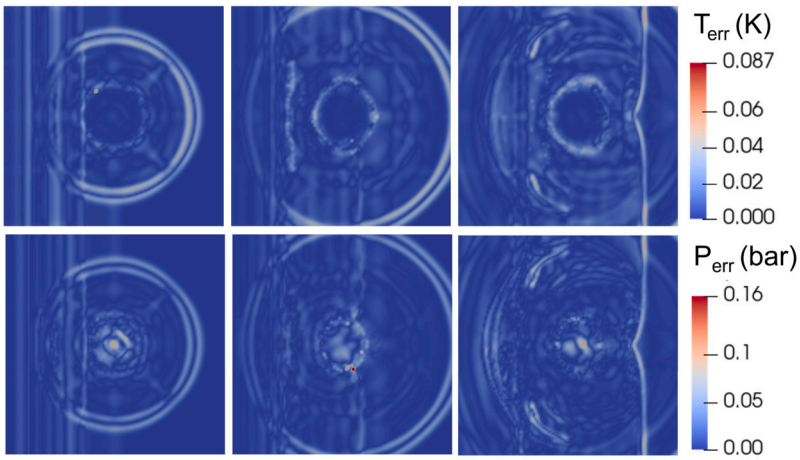

Fig. 22. The error contours of the ISAT-VLE method in the 3D transcritical shock-droplet interaction simulation. From left to right are the time instants of 5×10^{-7} s, 8.3×10^{-7} s, and 1.2×10^{-6} s.

influence of n-$C_{12}H_{26}$, which is liquid and harder to compress compared to N_2, causing the post-shock condition of the droplet to have lower pressure and temperature compared to the ambient post-shock condition. For the droplet points, the pressure undergoes a dramatic increase due to the shock wave, while the temperature remains relatively unchanged. A significant portion of these data points falls within the two-phase region (indicated by high α values), which requires VLE modeling. Moreover, despite the working pressure (350 bar - 400 bar) being much higher than the critical pressure of pure n-$C_{12}H_{26}$ (18 bar), the multi-component effect yields very high mixture critical pressures (560 bar at $X_{N_2} = 0.85$). As a result, the thermodynamic state is pushed back to the subcritical two-phase region (as depicted in Fig. 21), leading to phase separation (as observed in Fig. 20).

Fig. 22 illustrates the evolution of temperature and pressure errors in the 3D transcritical shock-droplet interaction simulation. It can be observed that these errors primarily manifest at the shock front and the acoustic waves generated by spurious pressure oscillations. A notable distinction between the temperature and pressure errors is that the temperature error remains localized near the surface of the droplet, whereas the pressure error extends into the droplet's interior. This distinction arises due to the minimal influence of the shock wave on the droplet's interior temperature during its passage (as shown in Fig. 20). Although the acoustic wave propagates the errors of both temperature and pressure over a wider region, their magnitudes are well-controlled. The temperature error is less than 0.1 K, and the pressure error remains below 0.2 bar. Both relative errors are maintained below 0.1%. Consequently, the ISAT-VLE method demonstrates an excellent error control in the 3D simulations.

When the initial temperature is increased from 565 K to 620 K, the phase separation at the droplet interface changes significantly. Initially, as shown in Fig. 23, the two-phase indicator α indicates that the droplet interface is situated within the subcritical two-phase region. However, as the shock wave passes through the droplet, the interface transitions into the single-phase region, eventually leading to the complete disappearance of phase separation. This process is further illustrated in the P-T phase diagram depicted in Fig. 24. The data points of the initial condition lie within the phase boundary of the mixture of $X_{N_2} = 0.7, X_{C_{12}H_{26}} = 0.3$, confirming that the droplet interface is in the subcritical two-phase region. As the shock wave elevates the droplet's pressure, the thermodynamic conditions move outside of the phase boundary, resulting in the elimination of phase separation.

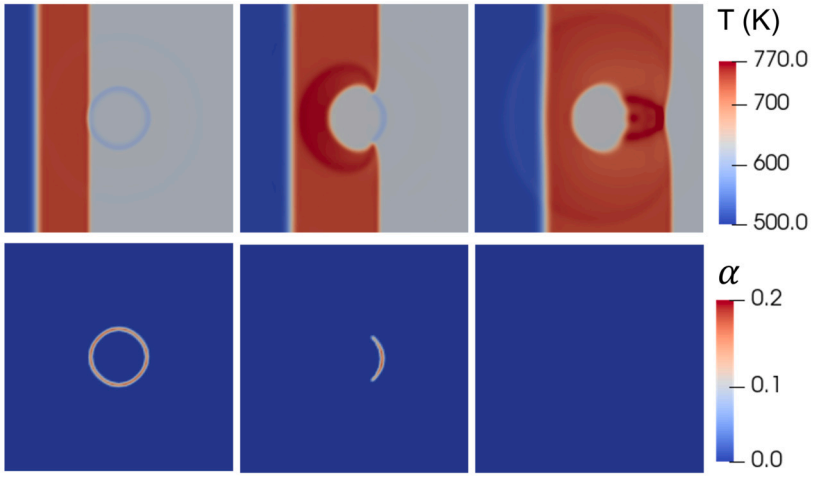

Fig. 23. 3D transcritical shock-droplet interaction simulation with an initial temperature of 620 K: the top figures are density, the bottom figures are α, where $\alpha = \beta(1-\beta)$ is a two-phase interface indicator. From left to right are the time instants of 5×10^{-7} s, 8.3×10^{-7} s, and 1.2×10^{-6} s.

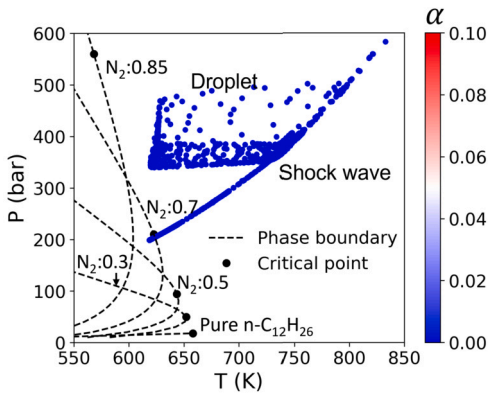

Fig. 24. Phase diagram of the 3D transcritical shock-droplet interaction simulation with an initial temperature 600 K. Data points are taken from the time instance of 1.2×10^{-6} s, when the shock wave has passed the droplet completely. The data points of the expansion wave are removed. The labels in the figure are mole fractions of N_2.

Next, we explore the behavior of a two-component droplet. In this simulation, the droplet consists of 60% n-$C_{12}H_{26}$ and 40% n-C_8H_{18} (by mole). The initial temperature is set to 565 K, which is consistent with the first case in this section. In Fig. 25, as the shock wave passes through the droplet, we observe the disappearance of part of the two-phase boundary. This is because n-C_8H_{18} has a lower critical temperature of 568.9K compared to n-$C_{12}H_{26}$ (658.2 K), which changes the phase boundary (see Fig. 26). In addition, a new phenomenon is observed: even after the shock wave completely passes through the droplet, some sections of the droplet interface remain in the two-phase region. This is attributed to the relatively low pressure in the wake region behind the droplet, allowing the interface to remain in the subcritical two-phase region. This reasoning is clearly depicted in the phase diagram of Fig. 26, where the three-component system (n-$C_{12}H_{26}$/n-C_8H_{18}/N_2) exhibits a two-phase region overlapping with the low-pressure corner of the post-shock data points.

The 3D transcritical shock-droplet interaction simulations with a single-component droplet and an initial temperature of 565 K were conducted using 128 CPU cores. The speed-up performance of the ISAT-VLE method is illustrated in Fig. 27. Initially, the ISAT-VLE case exhibited a speed-up of approximately 33 times compared to the VLE case without ISAT. Even in the later stages, it still maintained a significant speed-up of 15 times. On average, the ISAT-VLE method achieved a speed-up factor of approximately 17, showing its effectiveness in accelerating the 3D VLE-based CFD simulations. Furthermore, it is evident that compared with the 3D transcritical TML simulation (as shown in Fig. 14), the computational workload difference between different regions is not as pronounced, making the results closer to those obtained from 2D simulations.

4. Conclusions

Vapor-liquid equilibrium (VLE) is a family of first-principled thermodynamic models for simulating high-pressure transcritical multiphase flows, as it accurately captures phase transitions at high-pressure conditions that are challenging to handle with other methods. However, the computational cost of the VLE method limits its widespread application to real-world systems with many

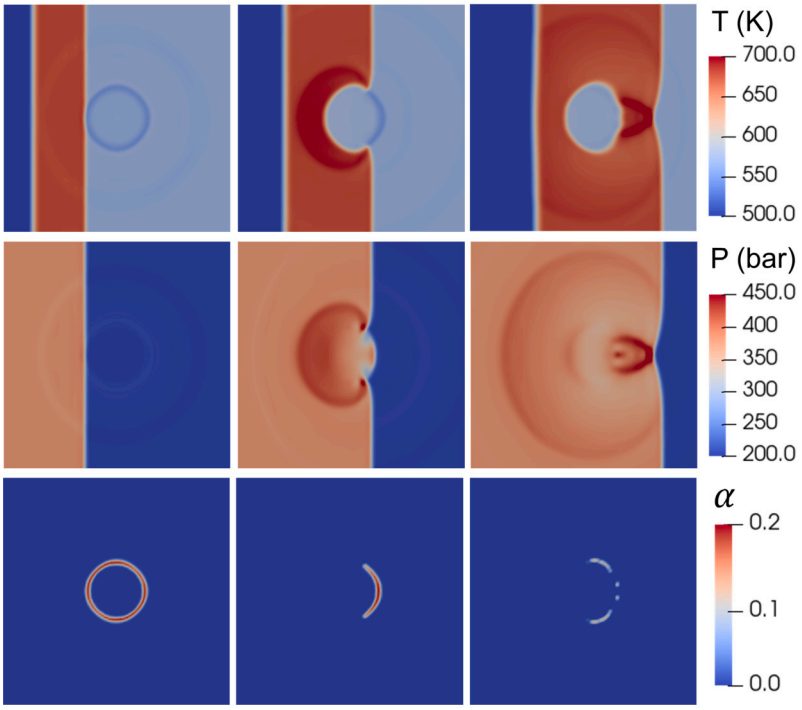

Fig. 25. 3D transcritical shock-droplet interaction simulation with an initial temperature of 565 K, and the droplet consists of 60% n-$C_{12}H_{26}$ and 40% n-C_8H_{18} (by mole): the top figures are density, and the bottom figures are α, where $\alpha = \beta(1-\beta)$ is a two-phase interface indicator. From left to right are the time instants of 5×10^{-7} s, 8.3×10^{-7} s, 120×10^{-7} s.

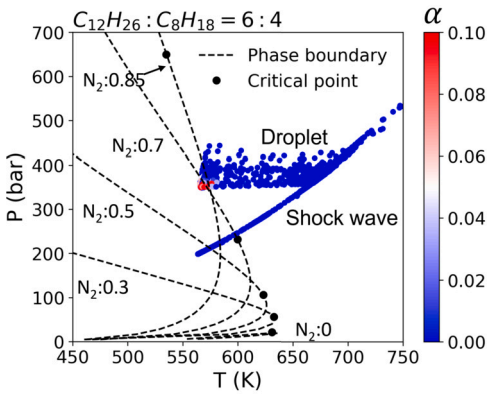

Fig. 26. Phase diagram of the 3D transcritical shock-droplet interaction simulation with an initial temperature of 565 K, and the droplet consists of 60% n-$C_{12}H_{26}$ and 40% n-C_8H_{18} (by mole). Data points are taken from the time instance of 1.2×10^{-6} s, when the shock wave has passed the droplet completely. The data points of the expansion wave are removed. The labels in the figure are mole fractions of N_2.

components. Our tests reveal that direct VLE calculations consume approximately 90% of the computational resources. In this study, we have developed a new ISAT-VLE method based on the original *in situ* adaptive tabulation (ISAT) method to enhance the computational efficiency of VLE-based CFD simulation, while minimizing the memory usage. Recognizing the dependence of the ISAT algorithm on the VLE solver's convergence characteristics, a modified Wilson equation based initial guess for equilibrium constant K_i is also introduced.

We developed two versions of ISAT-VLE method for both fully conservative (FC) and double flux (DF) schemes. To evaluate the performance and error control of the ISAT-VLE method, we conducted simulations for two representative transcritical multiphase flow scenarios: transcritical temporal mixing layer (TML) and transcritical shock-droplet interaction. These scenarios represent shearing layers in jet flows and liquid-fueled detonation waves, respectively. Both 2D and 3D configurations were employed for testing the ISAT-VLE method.

In transcritical TML simulations, the ISAT method accelerated VLE model calculations by a factor of approximately 10, while keeping the maximum error within 1%. The 3D results demonstrated that the VLE model effectively captured the formation of

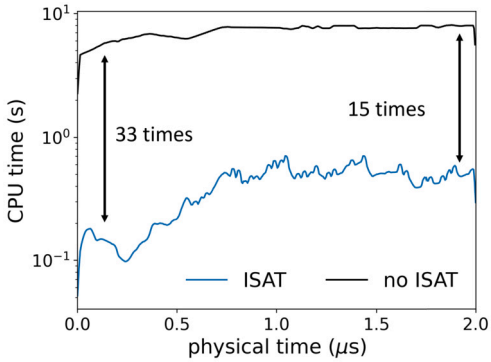

Fig. 27. Speed-up performance of the ISAT-VLE method in the 3D transcritical shock-droplet interaction simulation.

vortices and phase separation at the interface. In shock-droplet interaction simulations, the ISAT-VLE method achieved a larger speed-up factor (around 20) with a similar error control.

We also compared the DF scheme with the FC scheme. The DF scheme achieved an even larger speed-up factor (about 60). However, the DF scheme, being a non-conservative scheme, introduced energy errors, leading to greater temperature oscillations near the contact discontinuities (e.g., droplet interface). When choosing a scheme, if the results are temperature-sensitive, the FC scheme should be considered; otherwise, the DF scheme is a good choice. In some cases, the strong spurious pressure oscillation caused the simulations to crash, and the DF scheme proved effective in addressing this issue. However, it is important to note that neither fully conservative schemes nor quasi-conservative schemes (represented by the DF scheme) can perfectly solve the problem of spurious pressure oscillations from real-fluid EOS and VLE, and the development of better methods is still desired.

We also proposed new ISAT record deletion methods to enhance the ISAT search performance by accurately detecting and quickly removing redundant ISAT records. These methods involve active deletion of records that have not been accessed within a specified number of recent time steps, along with adaptive updates to the maximum table size. To facilitate the implementation of these methods, a new data structure has been devised using a combination of a circular array and linked lists, enabling efficient support for various operations. These novel methods result in a performance improvement of 6% over the already highly optimized ISAT-VLE algorithm.

We further investigated the results at different initial temperature conditions in transcritical shock-droplet interaction simulations. The findings showed that at appropriate conditions, the high temperature and pressure generated by the shock wave could push a droplet of n-$C_{12}H_{26}$ into the single-phase state, eliminating the two-phase interface. In this state, droplet breakup is primarily governed by diffusion. Additionally, we changed the configuration by replacing the n-$C_{12}H_{26}$ droplet with a two-component (n-$C_{12}H_{26}$/n-C_8H_{18}) droplet. Due to the change in composition, this two-component droplet entered the single-phase state at a lower temperature.

Moreover, we observed an intriguing phenomenon. After the shock wave passed through the droplet, a low-temperature and low-pressure annular section persisted at the wake of the droplet for a certain period. At specific conditions, phase separation disappears at the droplet interface except for the annular region, while the annular region still exhibits a two-phase boundary.

While we have not tested the performance of ISAT-VLE model in cases when large numbers of species (e.g., 10-20 species) present in the systems, our development of important features of the ISAT-VLE model (such as run-time ISAT growth, error control, analytical Jacobians, and robust memory management through the removal of old and redundant entries) promises more efficient performance compared to standard tabulation techniques, even in high-dimensional tabulations with large numbers of species.

In conclusion, the ISAT-VLE method developed in this study achieves significant acceleration in VLE-based CFD simulations while controlling the errors within 1% or lower, enabling VLE-based CFD simulations of more complex high-pressure transcritical multiphase flows with many components.

CRediT authorship contribution statement

Hongyuan Zhang: Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft. **Navneeth Srinivasan:** Data curation, Investigation, Methodology, Software, Writing – review & editing. **Suo Yang:** Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, 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

S. Yang gratefully acknowledges the support from the Office of Naval Research (ONR) grant under Award No. N00014-22-1-2287 under the supervision of project monitor Dr. Steven Martens, and the National Science Foundation (NSF) grant under Award No. CBET 2023932. H. Zhang gratefully acknowledges the support from the 3M Science and Technology Doctoral Fellowship, UMII Mn-DRIVE Graduate Assistantship, and Frontera Computational Science Fellowship. The authors gratefully acknowledge the computing resources provided by the Minnesota Supercomputing Institute (MSI).

Appendix A. Analytical framework for VLE

The analytical framework is based on the following derivatives:

$$\left(\frac{\partial x_k}{\partial T}\right)_{P,\mathbf{z}}, \left(\frac{\partial y_k}{\partial T}\right)_{P,\mathbf{z}}, \left(\frac{\partial x_k}{\partial P}\right)_{T,\mathbf{z}}, \left(\frac{\partial y_k}{\partial P}\right)_{T,\mathbf{z}}, \left(\frac{\partial x_k}{\partial z_i}\right)_{T,P,z_{s,s\neq i}}, \left(\frac{\partial y_k}{\partial z_i}\right)_{T,P,z_{s,s\neq i}} \tag{51}$$

To obtain them, we follow Tudisco and Menon's work [76] by calculating the following derivatives:

$$x_{T,k} = \left(\frac{\partial v_k}{\partial T}\right)_{P,\mathbf{z}}, \quad x_{P,k} = \left(\frac{\partial v_k}{\partial P}\right)_{T,\mathbf{z}}, \quad x^i_{N,j} = \left(\frac{\partial N_{j,v}}{\partial N_i}\right)_{T,P,N_{k,k\neq i}}, \tag{52}$$

where v_k is the mole fraction of component k in the vapor phase of the mixture (i.e., $v_k = \beta y_k$), $\mathbf{z}$ is the mixture mole fraction in the vector form, N_i is the total number of moles of component i in the mixture, and $N_{k,v}$ are the number of moles of component k in the vapor phase.

These derivatives are given by the solution of three linear equation systems:

$$\boldsymbol{A}\boldsymbol{x}_T = \boldsymbol{b}_T, \quad \boldsymbol{A}\boldsymbol{x}_P = \boldsymbol{b}_P, \quad \boldsymbol{C}\boldsymbol{x}^i_N = \boldsymbol{b}^i_N, i = 1, ..., N, \tag{53}$$

where N is the number of components.

The coefficients of the linear systems derivatives are shown below:

$$A_{ik} = \left(1 - \frac{\delta_{ki} z_k}{y_k x_k}\right) + \sum_{j=1}^{N} \left[(1-\beta)\left(\frac{\partial \ln \phi_{i,v}}{\partial y_j}\right)_{T,P,y_{s,s\neq j}} (y_i - \delta_{kj}) + \beta \left(\frac{\partial \ln \phi_{i,l}}{\partial x_j}\right)_{T,P,x_{s,s\neq j}} (x_j - \delta_{jk})\right] \tag{54}$$

$$C_{kj} = \sum_{s=1}^{N} \left[\left(\frac{\partial \ln \phi_{k,l}}{\partial x_s}\right)_{T,p,x_{r,r\neq s}} \left(\frac{\delta_{sj} - x_s}{1-\beta}\right) + \left(\frac{\partial \ln \phi_{k,v}}{\partial y_s}\right)_{T,p,y_{r,r\neq s}} \left(\frac{\delta_{sj} - y_s}{\beta}\right)\right] + \frac{1}{x_k}\frac{\delta_{kj} - x_k}{1-\beta} + \frac{1}{y_k}\frac{\delta_{kj} - y_k}{\beta} \tag{55}$$

$$b_{T,i} = \beta(1-\beta)\left[\left(\frac{\partial \ln \phi_{i,v}}{\partial T}\right)_{P,\mathbf{y}} - \left(\frac{\partial \ln \phi_{i,l}}{\partial T}\right)_{P,\mathbf{x}}\right], b_{P,i} = \beta(1-\beta)\left[\left(\frac{\partial \ln \phi_{i,v}}{\partial P}\right)_{T,\mathbf{y}} - \left(\frac{\partial \ln \phi_{i,l}}{\partial P}\right)_{T,\mathbf{x}}\right] \tag{56}$$

$$b^i_{N,k} = \sum_{j=1}^{N} \delta_{ji} \left\{ \sum_{s=1}^{N} \left[\left(\frac{\partial \ln \phi_{k,l}}{\partial x_s}\right)_{T,p,x_s} \left(\frac{\delta_{sj} - x_s}{1-\beta}\right)\right] + \frac{\delta_{ji}}{x_k}\frac{\delta_{kj} - x_k}{1-\beta} \right\} \tag{57}$$

These three linear equation systems are obtained by taking the derivative of the fugacity equation, and the derivatives with respect to pressure and temperature have the same coefficient matrix $\boldsymbol{A}$. The linear systems are solved using the LU decomposition method in the current work. Using the relation $\beta = \sum_i v_i$, the β derivative is obtained:

$$\left(\frac{\partial \beta}{\partial T}\right)_{P,\mathbf{z}} = \sum_{k=1}^{N} \left(\frac{\partial v_k}{\partial T}\right)_{P,\mathbf{z}}, \left(\frac{\partial \beta}{\partial P}\right)_{T,\mathbf{z}} = \sum_{k=1}^{N} \left(\frac{\partial v_k}{\partial P}\right)_{T,\mathbf{z}} \tag{58}$$

Using the relation between v_i, β, x_i, y_i, the first two terms in Eq. (51) can be obtained:

$$\left(\frac{\partial x_i}{\partial T}\right)_{P,\mathbf{z}} = \left(\frac{\partial (z_i - v_i)/(1-\beta)}{\partial T}\right)_{P,\mathbf{z}} = \frac{x_i}{1-\beta} \sum_{k=1}^{N} \left(1 - \frac{\delta_{ik}}{x_k}\right) \left(\frac{\partial v_k}{\partial T}\right)_{P,\mathbf{z}}$$

$$\left(\frac{\partial y_i}{\partial T}\right)_{P,\mathbf{z}} = \left(\frac{\partial v_i/\beta}{\partial T}\right)_{P,\mathbf{z}} = -\frac{y_i}{\beta} \sum_{k=1}^{N} \left(1 - \frac{\delta_{ik}}{y_k}\right) \left(\frac{\partial v_k}{\partial T}\right)_{P,\mathbf{z}}$$

For derivatives with respect to pressure, similar formulas can be used, with all the $\left(\frac{\partial(\cdot)}{\partial T}\right)_{P,\mathbf{z}}$ replaced by $\left(\frac{\partial(\cdot)}{\partial P}\right)_{T,\mathbf{z}}$.

For the last two terms in Eq. (51), we need to deal with the constraints of $\mathbf{z}$ (i.e., $\sum_i z_i = 1$), which makes a single z_i not a free variable. Here, we use an extension method to extend $\mathbf{z}$ to the entire positive space:

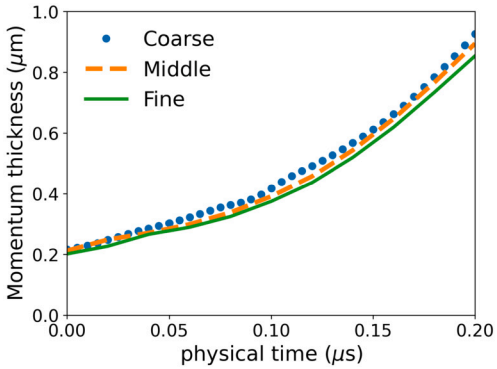

Fig. 28. Comparison of the transcritical TML momentum thickness predicted by three meshes.

$$F(\mathbf{z}) = \begin{cases} F(\mathbf{z}) & \text{if } \sum_i z_i = 1; \\ F\left(\frac{\mathbf{z}}{\sum_k z_k}\right) & \text{otherwise,} \end{cases} \tag{59}$$

where F represents any function of mixture mole fraction z_i. With this extension, $F(\mathbf{z}) = F(\mathbf{N})$, where $\mathbf{N}$ is the vector of all components' mole number. Note that although we extend the function, we only use $\mathbf{z}$ that meets the constraint for calculation, so we can further simplify the results with the constraint $\sum_i z_i = \sum_i N_i = 1$. Considering $\beta = \frac{\sum_k N_{k,v}}{\sum_k N_k}$, we have:

$$\left(\frac{\partial \beta}{\partial z_i}\right)_{T,P,z_{s,s\neq i}} = \left(\frac{\partial \frac{\sum_k N_{k,v}}{\sum_k N_k}}{\partial N_i}\right)_{T,P,z_{s,s\neq i}} = \sum_{j=1}^{N} \left(\frac{\partial N_{j,v}}{\partial N_i}\right)_{T,P,N_{s,s\neq j}} - \beta \tag{60}$$

Noticing the relation $y_k = \frac{N_{k,v}}{\beta \sum_j N_j}$ and $x_k = \frac{N_k - N_{k,v}}{(1-\beta)\sum_j N_j}$, the last two terms in Eq. (51) can be derived:

$$\left(\frac{\partial y_k}{\partial z_i}\right)_{T,P,z_{s,s\neq i}} = \frac{\partial \frac{N_{k,v}}{\beta \sum_j N_j}}{\partial N_i} = \frac{1}{\beta} \sum_{j=1}^{N} \left(\delta_{jk} - y_k\right) \left(\frac{\partial N_{j,v}}{\partial N_i}\right)_{T,P,N_{s,s\neq j}} \tag{61}$$

$$\left(\frac{\partial x_k}{\partial z_i}\right)_{T,P,z_{s,s\neq i}} = \frac{\partial \frac{N_k - N_{k,v}}{(1-\beta)\sum_j N_j}}{\partial N_i} = \frac{1}{(1-\beta)} \left[\sum_{s=1}^{N} \left(x_k - \delta_{jk}\right) \left(\frac{\partial N_{j,v}}{\partial N_i}\right)_{T,P,N_{s,s\neq j}} + \delta_{ik} - x_k\right] \tag{62}$$

Appendix B. Grid convergence study

B.1. 3D transcritical temporal mixing layer

We performed a grid convergence study for the transcritical temporal mixing layer simulations using the configuration of Case 1 in Table 1. Two additional meshes were generated based on the original mesh, with one being refined by a factor of two in each direction and the other coarsened by a factor of two in each direction. To compare the results obtained from these three meshes, we analyzed the momentum thickness defined as:

$$\delta_m = \frac{\int_{z_{min}}^{z_{max}} \left[\langle \rho u_x \rangle_{z_{max}} - \langle \rho u_x \rangle\right] \left[\langle \rho u_x \rangle - \langle \rho u_x \rangle_{z_{min}}\right] \mathrm{d}z}{\left(\langle \rho u_x \rangle_{z_{max}} - \langle \rho u_x \rangle_{z_{min}}\right)^2}, \tag{63}$$

which is a measure of mixing layer growth, where $\langle\rangle$ symbolizes averages over the (x, y) planes. The momentum thicknesses of three meshes versus time are presented in Fig. 28, which exhibit a maximum relative error of approximately 4.4%. The meshes with different resolutions do not have any significant impact on the results.

B.2. Transcritical shock-droplet interaction

We conducted a grid convergence study using the n-$C_{12}H_{26}$ droplet setup as described in Sec. 3.2. The simulations were performed using the fully conservative (FC) scheme. Based on the base mesh with dimensions of 256 × 256 × 256, we refined and coarsened the mesh by a factor of two in each direction, obtaining two additional meshes. In order to assess the impact of these meshes, we

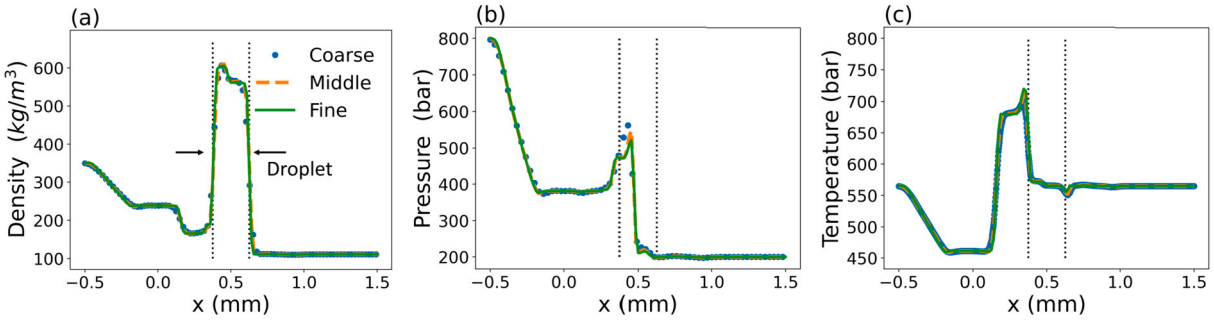

Fig. 29. Comparison of 3D transcritical shock-droplet interaction results on the center-line along the x direction for three meshes, at $t = 6.6 \times 10^{-7}$ s (when the shock wave reaches the center of the droplet).

compared the properties along the center-line in the x-direction when the shock wave reaches the center of the droplet ($t = 6.6 \times 10^{-7}$ s). In Fig. 29, the density, temperature, and pressure curves did not exhibit any significant differences among the different meshes, with a relative error of approximately 5%.

References

[1] L. Jofre, J. Urzay, Transcritical diffuse-interface hydrodynamics of propellants in high-pressure combustors of chemical propulsion systems, Prog. Energy Combust. Sci. 82 (2021) 100877.
[2] M. Fathi, S. Hickel, D. Roekaerts, Large eddy simulations of reacting and non-reacting transcritical fuel sprays using multiphase thermodynamics, Phys. Fluids 34 (2022) 085131.
[3] W.O. Mayer, A.H. Schik, B. Vielle, C. Chauveau, I. Gokalp, D.G. Talley, R.D. Woodward, Atomization and breakup of cryogenic propellants under high-pressure subcritical and supercritical conditions, J. Propuls. Power 14 (1998) 835–842.
[4] B. Chehroudi, D. Talley, E. Coy, Visual characteristics and initial growth rates of round cryogenic jets at subcritical and supercritical pressures, Phys. Fluids 14 (2002) 850–861.
[5] S. Candel, M. Juniper, G. Singla, P. Scouflaire, C. Rolon, Structure and dynamics of cryogenic flames at supercritical pressure, Combust. Sci. Technol. 178 (2006) 161–192.
[6] M. Oschwald, J. Smith, R. Branam, J. Hussong, A. Schik, B. Chehroudi, D. Talley, Injection of fluids into supercritical environments, Combust. Sci. Technol. 178 (2006) 49–100.
[7] A. Roy, C. Joly, C. Segal, Disintegrating supercritical jets in a subcritical environment, J. Fluid Mech. 717 (2013) 193–202.
[8] B. Chehroudi, D. Talley, E. Coy, Initial growth rate and visual characteristics of a round jet into a sub-to supercritical environment of relevance to rocket, gas turbine, and diesel engines, in: 37th Aerospace Sciences Meeting and Exhibit, 1999, p. 206.
[9] J.J. Smith, W. Mayer, Fundamentals of Supercritical Mixing and Combustion of Cryogenic Propellants, Liquid Rocket Thrust Chambers, American Institute of Aeronautics and Astronautics, Reston, VA, 2004, pp. 339–367.
[10] D.-Y. Peng, D.B. Robinson, A new two-constant equation of state, Ind. Eng. Chem. Fundam. 15 (1976) 59–64.
[11] G. Soave, Equilibrium constants from a modified Redlich-Kwong equation of state, Chem. Eng. Sci. 27 (1972) 1197–1203.
[12] H. Müller, C.A. Niedermeier, J. Matheis, M. Pfitzner, S. Hickel, Large-eddy simulation of nitrogen injection at trans- and supercritical conditions, Phys. Fluids 28 (2016) 015102.
[13] M. Benedict, G.B. Webb, L.C. Rubin, An empirical equation for thermodynamic properties of light hydrocarbons and their mixtures i. Methane, ethane, propane and n-butane, J. Chem. Phys. 8 (1940) 334–345.
[14] J. Gross, G. Sadowski, Perturbed-chain saft: an equation of state based on a perturbation theory for chain molecules, Ind. Eng. Chem. Res. 40 (2001) 1244–1260.
[15] T. Schmitt, L. Selle, A. Ruiz, B. Cuenot, Large-eddy simulation of supercritical-pressure round jets, AIAA J. 48 (2010) 2133–2144.
[16] H. Terashima, M. Koshi, Approach for simulating gas–liquid-like flows under supercritical pressures using a high-order central differencing scheme, J. Comput. Phys. 231 (2012) 6907–6923.
[17] A. Ruiz, Unsteady numerical simulations of transcritical turbulent combustion in liquid rocket engines, Ph.D. thesis, 2012.
[18] J.-P. Hickey, P.C. Ma, M. Ihme, S.S. Thakur, Large eddy simulation of shear coaxial rocket injector: real fluid effects, in: 49th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, 2013, p. 4071.
[19] S. Kawai, H. Terashima, H. Negishi, A robust and accurate numerical method for transcritical turbulent flows at supercritical pressure with an arbitrary equation of state, J. Comput. Phys. 300 (2015) 116–135.
[20] E. Johnsen, F. Ham, Preventing numerical errors generated by interface-capturing schemes in compressible multi-material flows, J. Comput. Phys. 231 (2012) 5705–5717.
[21] G. Lacaze, T. Schmitt, A. Ruiz, J.C. Oefelein, Comparison of energy-, pressure- and enthalpy-based approaches for modeling supercritical flows, Comput. Fluids 181 (2019) 35–56.
[22] R. Abgrall, S. Karni, Computations of compressible multifluids, J. Comput. Phys. 169 (2001) 594–623.
[23] G. Billet, R. Abgrall, An adaptive shock-capturing algorithm for solving unsteady reactive flows, Comput. Fluids 32 (2003) 1473–1495.
[24] G. Billet, J. Ryan, A Runge–Kutta discontinuous Galerkin approach to solve reactive flows: the hyperbolic operator, J. Comput. Phys. 230 (2011) 1064–1083.
[25] Y. Lv, M. Ihme, Discontinuous Galerkin method for multicomponent chemically reacting flows and combustion, J. Comput. Phys. 270 (2014) 105–137.
[26] P.C. Ma, Y. Lv, M. Ihme, An entropy-stable hybrid scheme for simulations of transcritical real-fluid flows, J. Comput. Phys. 340 (2017) 330–357.
[27] P. Tudisco, S. Menon, Numerical investigations of phase-separation during multi-component mixing at super-critical conditions, Flow Turbul. Combust. 104 (2020) 693–724.
[28] P. Van Konynenburg, R. Scott, Critical lines and phase equilibria in binary van der Waals mixtures, Philos. Trans. R. Soc. Lond. Ser. A, Math. Phys. Sci. 298 (1980) 495–540.
[29] H. Zhang, P. Yi, S. Yang, Multicomponent effects on the supercritical co2 systems: mixture critical point and phase separation, Flow Turbul. Combust. 109 (2022) 515–543.
[30] Y. Pei, S. Som, E. Pomraning, P.K. Senecal, S.A. Skeen, J. Manin, L.M. Pickett, Large eddy simulation of a reacting spray flame with multiple realizations under compression ignition engine conditions, Combust. Flame 162 (2015) 4442–4455.

[31] H. Kahila, A. Wehrfritz, O. Kaario, M.G. Masouleh, N. Maes, B. Somers, V. Vuorinen, Large-eddy simulation on the influence of injection pressure in reacting spray a, Combust. Flame 191 (2018) 142–159.
[32] M. Gadalla, J. Kannan, B. Tekgül, S. Karimkashi, O. Kaario, V. Vuorinen, Large-eddy simulation of ecn spray a: sensitivity study on modeling assumptions, Energies 13 (2020) 3360.
[33] V. Yang, Modeling of supercritical vaporization, mixing, and combustion processes in liquid-fueled propulsion systems, Proc. Combust. Inst. 28 (2000) 925–942.
[34] G. Lacaze, A. Misdariis, A. Ruiz, J.C. Oefelein, Analysis of high-pressure diesel fuel injection processes using les with real-fluid thermodynamics and transport, Proc. Combust. Inst. 35 (2015) 1603–1611.
[35] J.R. Edwards, R.K. Franklin, M.-S. Liou, Low-diffusion flux-splitting methods for real fluid flows with phase transitions, AIAA J. 38 (2000) 1624–1633.
[36] A.M. Star, J.R. Edwards, K.-C. Lin, S. Cox-Stouffer, T.A. Jackson, Numerical simulation of injection of supercritical ethylene into nitrogen, J. Propuls. Power 22 (2006) 809–819.
[37] J. Matheis, S. Hickel, Multi-component vapor-liquid equilibrium model for les of high-pressure fuel injection and application to ecn spray a, Int. J. Multiph. Flow 99 (2018) 294–311.
[38] S. Yang, P. Yi, C. Habchi, Real-fluid injection modeling and les simulation of the ecn spray a injector using a fully compressible two-phase flow approach, Int. J. Multiph. Flow 122 (2020) 103145.
[39] P. Yi, S. Yang, C. Habchi, R. Lugo, A multicomponent real-fluid fully compressible four-equation model for two-phase flow with phase change, Phys. Fluids 31 (2019) 026102.
[40] B. Yang, Q. Duan, B. Liu, K. Zeng, Parametric investigation of low pressure dual-fuel direct injection on the combustion performance and emissions characteristics in a rcci engine fueled with diesel and ch4, Fuel 260 (2020) 116408.
[41] P. Tudisco, S. Menon, A vapor–liquid equilibrium induced Lewis number effect in real-gas shear layers: a theoretical study, Phys. Fluids 32 (2020) 112111.
[42] N. Srinivasan, H. Zhang, S. Yang, A vle-based reacting flow solver for high-pressure transcritical two-phase combustion, in: AIAA SCITECH 2023 Forum, 2023, p. 1858.
[43] P. Yi, S. Jafari, S. Yang, C. Habchi, F. Rueil-Malmaison, Numerical analysis of subcritical evaporation and transcritical mixing of droplet using a tabulated multicomponent vapor-liquid equilibrium model, in: ILASS–Europe 2019, 29th Conference on Liquid Atomization and Spray Systems, 2019.
[44] P. Koukouvinis, A. Vidal-Roncero, C. Rodriguez, M. Gavaises, L. Pickett, High pressure/high temperature multiphase simulations of dodecane injection to nitrogen: application on ecn spray-a, Fuel 275 (2020) 117871.
[45] S. Jafari, H. Gaballa, C. Habchi, J.-C. de Hemptinne, Towards understanding the structure of subcritical and transcritical liquid–gas interfaces using a tabulated real fluid modeling approach, Energies 14 (2021) 5621.
[46] S. Jafari, H. Gaballa, C. Habchi, J.-C. De Hemptinne, P. Mougin, Exploring the interaction between phase separation and turbulent fluid dynamics in multi-species supercritical jets using a tabulated real-fluid model, J. Supercrit. Fluids 184 (2022) 105557.
[47] H. Gaballa, S. Jafari, C. Habchi, J.-C. de Hemptinne, Numerical investigation of droplet evaporation in high-pressure dual-fuel conditions using a tabulated real-fluid model, Int. J. Heat Mass Transf. 189 (2022) 122671.
[48] S. Wang, N. Sobecki, D. Ding, L. Zhu, Y.-S. Wu, Accelerating and stabilizing the vapor-liquid equilibrium (vle) calculation in compositional simulation of unconventional reservoirs using deep learning based flash calculation, Fuel 253 (2019) 209–219.
[49] P. Koukouvinis, C. Rodriguez, J. Hwang, I. Karathanassis, M. Gavaises, L. Pickett, Machine learning and transcritical sprays: a demonstration study of their potential in ecn spray-a, Int. J. Eng. Res. 23 (2022) 1556–1572.
[50] S.B. Pope, Computationally efficient implementation of combustion chemistry using in situ adaptive tabulation, Combust. Theory Model. 1 (1997) 41–63.
[51] J.J. Shah, R.O. Fox, Computational fluid dynamics simulation of chemical reactors: application of in situ adaptive tabulation to methane thermochlorination chemistry, Ind. Eng. Chem. Res. 38 (1999) 4200–4212.
[52] N.H. Kolhapure, R.O. Fox, A. Daiß, F.-O. Mähling, Pdf simulations of ethylene decomposition in tubular ldpe reactors, AIChE J. 51 (2005) 585–606.
[53] S. Mazumder, Modeling full-scale monolithic catalytic converters: challenges and possible solutions, J. Heat Transf. 129 (2006) 526–535.
[54] S. Mazumder, Adaptation of the in situ adaptive tabulation (isat) procedure for efficient computation of surface reactions, Comput. Chem. Eng. 30 (2005) 115–124.
[55] A. Varshney, A. Armaou, Multiscale optimization using hybrid pde/kmc process systems with application to thin film growth, Chem. Eng. Sci. 60 (2005) 6780–6794.
[56] A. Arsenlis, N. Barton, R. Becker, R. Rudd, Generalized in situ adaptive tabulation for constitutive model evaluation in plasticity, Comput. Methods Appl. Mech. Eng. 196 (2006) 1–13.
[57] J.D. Hedengren, T.F. Edgar, Approximate nonlinear model predictive control with in situ adaptive tabulation, Comput. Chem. Eng. 32 (2008) 706–714.
[58] C.E. Lacey, A.G. Novoselov, M.E. Mueller, In-situ adaptive manifolds: enabling computationally efficient simulations of complex turbulent reacting flows, Proc. Combust. Inst. 38 (2021) 2673–2680.
[59] A. Kurganov, E. Tadmor, New high-resolution central schemes for nonlinear conservation laws and convection–diffusion equations, J. Comput. Phys. 160 (2000) 241–282.
[60] C.J. Greenshields, H.G. Weller, L. Gasparini, J.M. Reese, Implementation of semi-discrete, non-staggered central schemes in a colocated, polyhedral, finite volume framework, for high-speed viscous flows, Int. J. Numer. Methods Fluids 63 (2010) 1–21.
[61] R.C. Reid, J.M. Prausnitz, T.K. Sherwood, The Properties of Liquids and Gases (Stichworte Teil 2), McGraw-Hill, 1977.
[62] G. Eliosa-Jiménez, F. García-Sánchez, G. Silva-Oliver, R. Macías-Salinas, Vapor–liquid equilibrium data for the nitrogen+ n-octane system from (344.5 to 543.5) k and at pressures up to 50 mpa, Fluid Phase Equilib. 282 (2009) 3–10.
[63] T. García-Córdova, D.N. Justo-García, B.E. García-Flores, F. García-Sánchez, Vapor-liquid equilibrium data for the nitrogen+ dodecane system at temperatures from (344 to 593) k and at pressures up to 60 mpa, J. Chem. Eng. Data 56 (2011) 1555–1564.
[64] M.L. Michelsen, The isothermal flash problem. Part i. Stability, Fluid Phase Equilib. 9 (1982) 1–19.
[65] S. Saha, J.J. Carroll, The isoenergetic-isochoric flash, Fluid Phase Equilib. 138 (1997) 23–41.
[66] H. Rachford Jr, J. Rice, et al., Procedure for use of electronic digital computers in calculating flash vaporization hydrocarbon equilibrium, J. Pet. Technol. 4 (10) (1952) 327–328.
[67] M.L. Michelsen, J.M. Mollerup, Thermodynamic Models: Fundamentals & Computational Aspects, vol. 775, Tie-Line Publications Holte, Denmark, 2007.
[68] G.M. Wilson, Vapor-liquid equilibrium. xi. A new expression for the excess free energy of mixing, J. Am. Chem. Soc. 86 (1964) 127–130.
[69] T.H. Chung, M. Ajlan, L.L. Lee, K.E. Starling, Generalized multiparameter correlation for nonpolar and polar fluid transport properties, Ind. Eng. Chem. Res. 27 (1988) 671–679.
[70] R.J. Kee, F.M. Rupley, E. Meeks, J.A. Miller, CHEMKIN-III: a FORTRAN chemical kinetics package for the analysis of gas-phase chemical and plasma kinetics, Technical Report, Sandia National Lab. (SNL-CA), Livermore, CA (United States), 1996.
[71] E.N. Fuller, P.D. Schettler, J.C. Giddings, New method for prediction of binary gas-phase diffusion coefficients, Ind. Eng. Chem. 58 (1966) 18–27.
[72] S. Takahashi, Preparation of a generalized chart for the diffusion coefficients of gases at high pressures, J. Chem. Eng. Jpn. 7 (1975) 417–420.
[73] A. Kurganov, S. Noelle, G. Petrova, Semidiscrete central-upwind schemes for hyperbolic conservation laws and Hamilton–Jacobi equations, SIAM J. Sci. Comput. 23 (2001) 707–740.
[74] B. Van Leer, Towards the ultimate conservative difference scheme. ii. Monotonicity and conservation combined in a second-order scheme, J. Comput. Phys. 14 (1974) 361–370.

[75] L. Lu, S.B. Pope, An improved algorithm for in situ adaptive tabulation, J. Comput. Phys. 228 (2009) 361–386.
[76] P. Tudisco, S. Menon, Analytical framework for real-gas mixtures with phase-equilibrium thermodynamics, J. Supercrit. Fluids 164 (2020) 104929.
[77] B.E. Poling, J.M. Prausnitz, J.P. O'connell, Properties of Gases and Liquids, McGraw-Hill Education, 2001.
[78] D.A. Schwer, D. Kessler, Liquid-Fueled Detonation Modeling at the U.S. Naval Research Laboratory, 2018.
[79] R.S. Miller, K.G. Harstad, J. Bellan, Direct numerical simulations of supercritical fluid mixing layers applied to heptane–nitrogen, J. Fluid Mech. 436 (2001) 1–39.
[80] E. Masi, J. Bellan, K.G. Harstad, N.A. Okong'o, Multi-species turbulent mixing under supercritical-pressure conditions: modelling, direct numerical simulation and analysis revealing species spinodal decomposition, J. Fluid Mech. 721 (2013) 578–626.