Communications on Applied Mathematics and Computation (2024) 6:2265–2294
https://doi.org/10.1007/s42967-023-00349-8

ORIGINAL PAPER

A Multiscale Method for Two-Component, Two-Phase Flow with a Neural Network Surrogate

Jim Magiera[1] · **Christian Rohde**[1]

This contribution is dedicated to Professor Dr. Gerald Warnecke in celebration of his contributions to the field of hyperbolic conservation laws and, in particular, to his seminal works on multi-phase flow modeling.

Received: 26 April 2023 / Revised: 3 September 2023 / Accepted: 29 October 2023 /
Published online: 7 May 2024

Abstract
Understanding the dynamics of phase boundaries in fluids requires quantitative knowledge about the microscale processes at the interface. We consider the sharp-interface motion of the compressible two-component flow and propose a heterogeneous multiscale method (HMM) to describe the flow fields accurately. The multiscale approach combines a hyperbolic system of balance laws on the continuum scale with molecular-dynamics (MD) simulations on the microscale level. Notably, the multiscale approach is necessary to compute the interface dynamics because there is—at present—no closed continuum-scale model. The basic HMM relies on a moving-mesh finite-volume method and has been introduced recently for the compressible one-component flow with phase transitions by Magiera and Rohde in (J Comput Phys 469: 111551, 2022). To overcome the numerical complexity of the MD microscale model, a deep neural network is employed as an efficient surrogate model. The entire approach is finally applied to simulate droplet dynamics for argon-methane mixtures in several space dimensions. To our knowledge, such compressible two-phase dynamics accounting for microscale phase-change transfer rates have not yet been computed.

Keywords Phase transition · Hyperbolic balance laws for multi-component fluids · Multiscale modeling · Moving-mesh methods · Deep neural networks

Mathematics Subject Classification 76T10 · 65Z05 · 68T07

✉ Christian Rohde
christian.rohde@mathematik.uni-stuttgart.de

[1] Institute of Applied Analysis and Numerical Simulation, University of Stuttgart, Pfaffenwaldring 57, 70569 Stuttgart, Germany

1 Introduction

In this work, we consider the dynamics of a compressible two-component fluid which allows for phase changes between a liquid and a vapor phase state. Keeping temperature constant, we aim to determine continuum-scale quantities like component-wise densities and momenta on a spatial scale where all phase boundaries are explicitly represented, i.e., without further averaging like in, e.g., Baer-Nunziato modeling [4, 37, 43]. This choice implies that the phase—liquid or vapor—is uniquely determined by the values of the component densities. Having made this decision, we focus on a sharp-interface approach, and refer to [3] and just recently [24] for diffuse-interface models. We start from an extension of the isothermal Euler equations with frictional forces for one-phase flows, as introduced in [6], to the two-phase case, see (1) below. We will focus mainly on the two-component system comprised of the noble gases argon and methane which allows a relatively simple thermodynamical description. These evolution equations are supposed to hold in the liquid-phase domain $\mathbf{\Omega}_-(t)$ and the vapor-phase domain $\mathbf{\Omega}_+(t)$, where they form a set of hyperbolic balance laws. The two bulk domains are separated by a co-dimension-1 manifold, the interface $\mathbf{\Gamma} = \mathbf{\Gamma}(t)$. Thus, the entire fluid domain $\mathbf{\Omega} \subset \mathbb{R}^d$, $d \in \{1, 2, 3\}$, at any time $t \in [0, t_{\text{end}}]$, $t_{\text{end}} > 0$, is partitioned into $\mathbf{\Omega} = \mathbf{\Omega}_+(t) \cup \mathbf{\Gamma}(t) \cup \mathbf{\Omega}_+(t)$, see Fig. 1 for illustration. The basic mathematical model is provided in Sect. 2.

To determine the motion of the interface and to ensure the well posedness of the resulting free boundary value problem one has to prescribe transmission conditions across the interface $\mathbf{\Gamma}(t)$. Obviously, the Rankine-Hugoniot conditions have to be imposed; but it is well known, even in the one-component case, that additional conditions have to be prescribed at the interface [8, 17, 21, 36]. This can be achieved by imposing further algebraic relations that prescribe the entropy dissipation rate across the interface [40]. However, determining the explicit form of this relation remains a largely unsolved issue for complex flow regimes, including in particular the multi-component flow. To overcome this fundamental modeling problem on the continuum scale and to avoid using ad hoc closures, one can look at smaller spatial scales using a molecular-dynamics (MD) approach. On this ab initio level, the modeling of a multi-component fluid becomes accessible since each molecule of a component has the same physical properties regardless of the fluid phase it is part of (see Sect. 3). However, MD simulations can only be run for very limited spatial and temporal scales. We utilize a tailored MD algorithm to solve MD Riemann problems across the (discrete) interface only, whereas the dynamics in the bulk phase domains are approximated by a standard finite-volume method for the first-order two-component system (2). With this hybrid ansatz we propose a heterogeneous multiscale method (HMM, in the sense of [10]) that accounts for physically accurate interfacial mass and momentum transfer, as well as the motion of the phase boundary by an MD interface solver, see Sect. 4. A critical issue in this context is the coupling of the different models, i.e., from MD systems with pairwise interactions to continuum-scale Euler-type equations and vice versa. We rely on classical, statistical averaging to infer continuum-scale quantities from microscale particle simulations. In that way, we transfer the microscale behavior at fluid

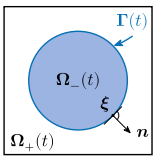

Fig. 1 Partitioned domain $\mathbf{\Omega}$ with interface $\mathbf{\Gamma}(t)$ and bulk domains $\mathbf{\Omega}_\pm$ for $d = 2$ spatial dimensions

interfaces to the continuum scale via a data-based approach. Concerning the numerical discretization on the continuum scale we employ a moving-mesh finite-volume method which enables us to include the MD dynamics directly at the phase boundary, while resolving the sharp interface directly within the mesh. The corresponding moving mesh algorithms are presented in [7, 32], see [1] for the corresponding open-source code.

Finally, to overcome the computational costs of expensive MD simulations, we apply a surrogate model for the MD interface solver. This surrogate model is based on the constraint-aware neural networks that have been developed in [30], which come up with the same computational efficiency, but ensure mass conservation of the numerical discretization, in particular across the interface.

In Sect. 5, we present numerical results for the HMM for the compressible two-component flow with phase transitions. We focus on a mixture of the noble gases argon and methane using their exact physical properties. In particular, this includes a specific equation of state (EOS) that is used in the bulk domains. The results comprise simulations in one, two, and three space dimensions and settle, for multiple space dimensions, around the impingement of pressure waves on droplets inducing oscillation and condensation processes. We summarize our findings in the concluding Sect. 6.

Up to our knowledge, the continuum-scale phase-transition regimes in Sect. 5 for realistic argon-methane mixtures have not been computed before using a physically accurate model for the interfacial motion. This contribution is the main novelty of the paper at hand, showing that the approach we have developed for the much simpler single-component cases in [31–33] can be readily extended to complex scenarios lacking any knowledge about the proper choice of transmission conditions on the continuum scale.

In the remainder of the introductory section, we embed our work into the existing literature for sharp-interface modeling in the phase-transition flow. In fact, two-phase fluid flow models with sharp interfaces for a single component are widely investigated, see, e.g., [11, 12, 14, 27, 38, 39] for various modeling approaches in the mathematical realm. As mentioned above, much less is known for the multi-component flow. We refer to the works [19, 20] which address the closure problem but do not provide a full continuum-mechanical closure. An alternative ansatz [18] builds on the integration of first-order models into phase-field models [9].

MD simulations have been readily and for a long time used tools for the dynamics of phase boundaries. However, multiscale modeling that connects to the continuum-scale dynamics for the compressible liquid-vapor flow has not been done up to our knowledge. The works in, e.g., [13, 22] indeed support the excellent match of up-scaled quantities if an appropriate closed continuum-scale model exists.

2 Modeling of Two-Component, Two-Phase Flow

In this section, we first review a continuum-scale model from [6] that governs the description of the two-phase flow for a fluid that consists of several components. We focus on argon-methane mixtures with comparably well-known fluid properties. Albeit the fact that the governing equations take the form of first-order balance laws, it turns out that the analytical understanding of hyperbolicity and the properties of solutions for, e.g., Riemann problems is quite limited, and cannot serve as the basis for numerical simulation.

Throughout this work, we consider all variables in combination with their physical units. This is important for converting microscale quantities to continuum-scale quantities later on. Moreover, the simulations in Sect. 5 refer to realistic fluid regimes for the argon-methane mixture.

2.1 An Isothermal Two-Component Flow Model

We consider isothermal, inviscid, non-reactive fluids consisting of two components, which we mark by an index $a \in \{0, 1\}$. Later, the index 0 indicates argon and the index 1 methane.

To model two-component flow, we choose the so-called class-II model for the multi-component flow which has been derived in [6] as a thermodynamically consistent ansatz. It includes Maxwell-Stefan diffusion to account for the crucial frictional interaction between components. The model can be written in terms of a first-order balance law. Precisely, we have for each component $a \in \{0, 1\}$ the equations

$$\begin{cases} \partial_t \rho_a + \nabla \cdot (\rho_a \boldsymbol{v}_a) = 0, \\ \partial_t (\rho_a \boldsymbol{v}_a) + \nabla \cdot (\rho_a \boldsymbol{v}_a \otimes \boldsymbol{v}_a) + \rho_a \nabla \mu_a = -T f_{ab} \rho_a \rho_b (\boldsymbol{v}_a - \boldsymbol{v}_b) \end{cases} \tag{1}$$

in $\boldsymbol{\Omega}_{\pm}(t)$ for $t \in (0, t_{\text{end}})$. The primary variables are the partial mass densities ρ_a [$\text{kg} \cdot \text{m}^{-3}$] and the partial velocities $\boldsymbol{v}_a$ [$\text{m} \cdot \text{s}^{-1}$]. The constant reference temperature of the mixture is denoted by T [K]. The function $\mu_a = \mu_a(\rho_0, \rho_1; T)$ [$\text{J} \cdot \text{kg}^{-1}$] represents the chemical potential of component $a \in \{0, 1\}$ with respect to the partial mass densities and temperature. We will discuss a specific choice for μ_a below.

The term $f_{01} = f_{10} > 0$ denotes the friction factor between the components 0, 1 and is proportional to the reciprocal of the Maxwell-Stefan diffusion coefficients $\mathcal{D}_{01}$ [$\text{m}^2 \cdot \text{s}^{-1}$]. More specifically, f_{01} computes as

$$f_{01} = \frac{R}{M_0 M_1 c} \cdot \frac{1}{\mathcal{D}_{01}}. \tag{2}$$

Here, c [$\text{mol} \cdot \text{m}^{-3}$] denotes the total molar concentration, which is defined by $c = c_0 + c_1$, where $c_a = \rho_a / M_a$ [$\text{mol} \cdot \text{m}^{-3}$] are the molar concentrations, and M_a [$\text{kg} \cdot \text{mol}^{-1}$] the molar masses of component $a \in \{0, 1\}$. The ideal gas constant is given by $R \approx 8.314\ \text{J} \cdot \text{mol}^{-1} \cdot \text{K}^{-1}$. Another relevant quantity in the context of the two-component flow is the mole fractions x_a [–], which are defined as $x_a = c_a / c$, $a \in \{0, 1\}$. They describe the ratio of the number of 0/1-component particles to the total amount of particles. For an in-depth discussion of friction in multi-component systems, we refer to, e.g., [25].

The initial data of (1) are comprised of a connected initial phase boundary $\boldsymbol{\Gamma}(0)$, initial partial mass densities $\rho_{a,0}(\boldsymbol{x})$, and initial partial velocity fields $\boldsymbol{v}_{a,0}(\boldsymbol{x})$, such that

$$\rho_a(\boldsymbol{x}, 0) = \rho_{a,0}(\boldsymbol{x}), \quad \boldsymbol{v}_a(\boldsymbol{x}, 0) = \boldsymbol{v}_{a,0}(\boldsymbol{x}) \quad \text{for } a \in \{0, 1\} \text{ and } \boldsymbol{x} \in \boldsymbol{\Omega}_{\pm}(0). \tag{3}$$

We define $\boldsymbol{U}_0(\boldsymbol{x}) = (\rho_{0,0}(\boldsymbol{x}), \rho_{1,0}(\boldsymbol{x}), (\rho_{0,0}\boldsymbol{v}_{0,0})(\boldsymbol{x})^{\top}, (\rho_{1,0}\boldsymbol{v}_{1,0})(\boldsymbol{x})^{\top})^{\top}$. Boundary conditions will be specified in Sect. 5.

To close the system (1), we have to provide expressions for the chemical potentials μ_0, μ_1. This means in the isothermal case that an EOS has to be defined which describes the functional dependencies for the chemical potentials $\mu_a = \mu_a(\rho_0, \rho_1; T)$. For the two-component flow, we choose the PC-SAFT EOS [15]. Its underlying theoretical framework provides us in particular with precise parameter values describing argon-methane mixtures, see Appendix A. This is the first reason for choosing this mixture in Sect. 5 on realistic numerical simulations (see Sect. 3 for the second reason related to molecular-dynamical modeling). We do not give the highly complex functional form of the PC-SAFT EOS. Rather, in Fig. 2, we display a plot of the mixture pressure for the argon-methane mixture at $T = 110\,\text{K}$ as a function of the total density $\rho_0 + \rho_1$ and the mole fraction x_0 of argon. We observe a vapor-state domain and, separated by a spinodal region, a liquid-state domain.

Although no proof is available we claim that for sufficiently small temperatures the set $\{(\rho_0, \rho_1, \boldsymbol{m}_0^\top := \rho_0 \boldsymbol{v}_0^\top, \boldsymbol{m}_1^\top := \rho_1 \boldsymbol{v}_1^\top)^\top \mid \rho_0, \rho_1 > 0, \boldsymbol{v}_0, \boldsymbol{v}_1 \in \mathbb{R}^d\}$ is separated into a liquid-state region $\mathcal{P}_-$ and a vapor-state region $\mathcal{P}_-$ which define the phases in our case. Having defined a liquid and a vapor phase, we assume that the two-component fluid in $\boldsymbol{\Omega}$ partitions the domain in two subdomains $\boldsymbol{\Omega}_\pm = \boldsymbol{\Omega}_\pm(t)$ separated by a single, sharp phase boundary $\boldsymbol{\Gamma}(t)$, see Fig. 1. Just as for the topology of the state space and the splitting, there is up to now no proof that the left-hand side of system (1) is hyperbolic in the state space

$$\mathcal{U} = \mathcal{P}_- \cup \mathcal{P}_+,$$

and elliptic in the spinodal region. Motivated by the situation of the single-component, two-phase flow this is exactly what we assume from now in [36]. In all the numerical simulations, we have not encountered contrary evidence.

2.2 Planar Two-Phase Solutions and the Riemann Problem

The system (1) consists of first-order evolution laws. To gain insight into the two-phase behavior of the solution, we fix $t \in [0, t_{\text{end}}]$ and some point $\boldsymbol{\xi}$ on the interface $\boldsymbol{\Gamma}(t)$. We consider planar solutions of (1) on the line depicted by the normal $\boldsymbol{n} = \boldsymbol{n}(\boldsymbol{\xi}, t) \in \mathbb{S}^{d-1}$ (pointing in the direction of the vapor-state domain $\boldsymbol{\Omega}_+(t)$), see again Fig. 1.

We define, for some state $\boldsymbol{U} \in \mathcal{U} \subset \mathbb{R}^{2+2d}$, the normal velocities $v_0 = \boldsymbol{v}_0 \cdot \boldsymbol{n}$, $v_1 = \boldsymbol{v}_1 \cdot \boldsymbol{n}$, and the $\boldsymbol{n}$-rotated state

$$\boldsymbol{u} := \boldsymbol{U}_{\|\boldsymbol{n}} = (\rho_0, \rho_1, m_0 := \rho_0 v_0, m_1 := \rho_1 v_1)^\top := (\rho_0, \rho_1, \rho_0 \boldsymbol{v}_0 \cdot \boldsymbol{n}, \rho_1 \boldsymbol{v}_1 \cdot \boldsymbol{n})^\top \in \mathbb{R}^4. \tag{4}$$

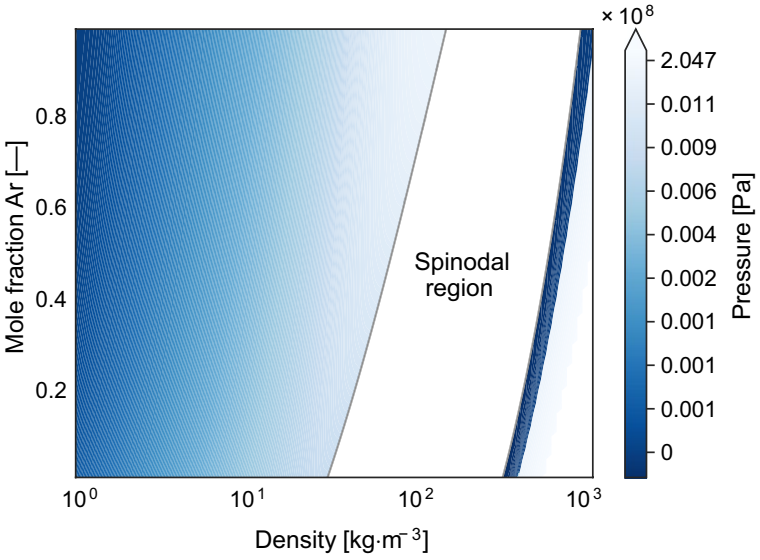

Fig. 2 Pressure of the PC-SAFT EOS for an argon-methane mixture at $T = 110\,\text{K}$. The left-hand (right-hand) region displays vapor (liquid) states

Furthermore, we set

$$\boldsymbol{q}(\boldsymbol{u}) := -Tf_{01}\rho_0\rho_1\big(0, 0, v_0 - v_1, v_1 - v_0\big)^\top.$$

Then, the rotational invariance of the Euler-type system (1) ensures that there is a matrix $\boldsymbol{A} = \boldsymbol{A}(\boldsymbol{u}; \boldsymbol{n}) \in \mathbb{R}^{4\times 4}$ such that the solution $\boldsymbol{u} = \boldsymbol{u}(x, t)$ of the $\boldsymbol{n}$-rotated (one-dimensional) problem

$$\begin{cases} \partial_t \boldsymbol{u} + \boldsymbol{A}(\boldsymbol{u}; \boldsymbol{n})\partial_x \boldsymbol{u} = \boldsymbol{q}(\boldsymbol{u}) & \text{in } \mathbb{R} \times (0, t_{\text{end}}), \\ \boldsymbol{u}(\cdot, 0) = \boldsymbol{u}_0 := \boldsymbol{U}_{0,\|\boldsymbol{n}} & \text{in } \mathbb{R} \end{cases} \tag{5}$$

defines a planar solution of (1). We recover the original states in $\mathcal{U}$ by the operator

$$P_{\mathbf{n}} \colon (\rho_0, \rho_1, m_0, m_1)^\top \mapsto (\rho_0, \rho_1, m_{0,1}n_1, \cdots, m_{0,d}n_d, m_{1,1}n_1, \cdots, m_{1,d}n_d)^\top \tag{6}$$

with $n_1, \cdots, n_d$ being the entries of $\boldsymbol{n}$. Now, posing the Riemann problem for the system (5) means to choose initial data of the form

$$\boldsymbol{u}_0(x) = \begin{cases} \boldsymbol{u}_- & \text{for } x < 0, \\ \boldsymbol{u}_+ & \text{for } x > 0, \end{cases} \tag{7}$$

where $\boldsymbol{u}_\pm$ are computed from given Riemann states $\boldsymbol{U}_\pm \in \mathcal{U}$ by

$$\boldsymbol{u}_\pm := (\boldsymbol{U}_\pm)_{\|\boldsymbol{n}}.$$

The Riemann states $\boldsymbol{U}_\pm \in \mathcal{P}_\pm$ should be understood as traces states in $\boldsymbol{\Omega}_\pm(t)$, respectively.

The Riemann problem for the $\boldsymbol{n}$-rotated system (5) will play a crucial role in the construction of the HMM in Sect. 4. The most convenient situation would be to have an exact Riemann solver for $(5)_1$. Motivated by the one-phase case, one can then expect that the solution of the Riemann problem (5), (7) includes exactly one isolated phase boundary connecting two constant states in $\mathcal{P}_\pm$. Its speed $s \in \mathbb{R}$ and the values of the adjacent states $\boldsymbol{u}^*_\pm := (\rho^*_{0,\pm}, \rho^*_{1,\pm}, v^*_{0,\pm}, v^*_{1,\pm})^\top \in \mathcal{P}_\pm$ would then provide all the necessary information about the future behavior of the solution locally at $\boldsymbol{\xi} \in \boldsymbol{\Gamma}(t)$. Let us denote such a (perfect) interface solver in terms of the original variables for (1), i.e., using the mapping (4) and the back projection $P_{\mathbf{n}}$, by

$$\mathcal{R} \colon (\boldsymbol{U}_-, \boldsymbol{U}_+; \boldsymbol{n}) \mapsto (\boldsymbol{U}^*_-, \boldsymbol{U}^*_+, s). \tag{8}$$

Actually, an exact Riemann solver $\mathcal{R} \colon \mathcal{P}_- \times \mathcal{P}_+ \times \mathbb{S}^{d-1} \to \mathcal{P}_- \times \mathcal{P}_+ \times \mathbb{R}$ for arbitrary two-phase Riemann data $\boldsymbol{U}_\pm$ is not available for the multicomponent flow (but see [18, 19] for some works in this direction). This problem is of course related to the lacking information about hyperbolicity of (1). We aim to substitute the missing Riemann solver by an interface solver based on MD simulations which will be discussed in the next section.

3 Molecular Dynamics for Multicomponent Fluids

One big advantage of the MD is its versatility to simulate fluid mixtures consisting of more than one fluid component. Fluid components in this context denote different types of molecules that make up the composition of a fluid.

In this section, we review the molecular-dynamical modeling for binary mixtures. Then we extend a numerical method that has been developed in [32] for one-component fluids to the

case of binary mixtures, i.e., fluid mixtures consisting of two components. We also discuss the compatibility of the MD modeling for argon-methane mixtures with the PC-SAFT EOS introduced in Sect. 2.1. As before, we indicate the two components by the numbers 0 and 1.

3.1 Molecular Dynamics for Binary Mixtures

We focus on fluid components that can be modeled by spherical, rotational invariant Lennard-Jones particles, such as noble gases or relatively simple molecules like methane. In the two-component setting, each particle belongs to either component 0, or to component 1. We denote the particle type for the i-th particle by ς_i. Accordingly, the mass of each particle is m_0 [u], if $\varsigma_i = 0$, or alternatively m_1 [u], if $\varsigma_i = 1$.

On the atomistic scale the dynamics of an ensemble of $N = N_0 + N_1$ particles with $N_0 \in \mathbb{N}$ particles of component 0 and $N_1 \in \mathbb{N}$ particles of component 1 are governed by the ordinary initial value problem [2]

$$\begin{cases} \boldsymbol{x}_i'(\tau) = \boldsymbol{v}_i(\tau), \quad \boldsymbol{v}_i'(\tau) = \boldsymbol{a}_i(\tau), \quad \boldsymbol{a}_i(\tau) = -\sum_{j \neq i} \nabla_{\boldsymbol{x}_i} \phi_{\varsigma_i \varsigma_j} \Big(\big\| \boldsymbol{x}_i(\tau) - \boldsymbol{x}_j(\tau) \big\| \Big) / m_{\varsigma_i}, \\ \boldsymbol{x}_i(0) = \boldsymbol{x}_i^0, \quad \boldsymbol{v}_i(0) = \boldsymbol{v}_i^0 \end{cases} \tag{9}$$

for the particle positions $\boldsymbol{x}_i(\tau) = (x_i(\tau), y_i(\tau), z_i(\tau))^\top \in \mathbb{R}^3$, velocities $\boldsymbol{v}_i(\tau) \in \mathbb{R}^3$, and accelerations $\boldsymbol{a}_i(\tau) \in \mathbb{R}^3$ of the i-th particle, with $\tau \in [0, \tau_{\text{end}}]$. Here $\tau_{\text{end}} > 0$ is the MD end time. The MD simulations are carried out in the atomistic-scale units [K] for energy, [Å] for distances, and [u] for mass. The crucial ingredients for the MD model (9) are the Lennard-Jones potentials ϕ_{ab} with

$$\phi_{ab}(r) = 4\varepsilon_{ab} \left(\left(\frac{\sigma_{ab}}{r} \right)^{12} - \left(\frac{\sigma_{ab}}{r} \right)^{6} \right). \tag{10}$$

The Lennard-Jones parameters $\varepsilon_{01} = \varepsilon_{10}$, $\sigma_{01} = \sigma_{10}$ depend on the corresponding parameters ε_0, σ_0, ε_1, σ_1 for single-component potentials. If both interacting particles belong to the same component $a \in \{0, 1\}$, the parameters are given by $\varepsilon_{aa} := \varepsilon_a, \sigma_{aa} := \sigma_a$. If they belong to different components, the Lorentz-Berthelot combination rules [5, 26] suggest for some scaling numbers η [–] and ξ [–]

$$\sigma_{01} = \sigma_{10} = \eta \frac{\sigma_0 + \sigma_1}{2}, \quad \varepsilon_{01} = \varepsilon_{10} = \xi \sqrt{\varepsilon_0 \varepsilon_1}. \tag{11}$$

In Fig. 3, the Lennard-Jones potentials for different parameters are plotted.

As discretization for (9) we use the classical Velocity-Verlet algorithm [16] which amounts to performing the following steps with a given time step $\Delta\tau > 0$, $N_{\text{end}} = \lceil \tau_{\text{end}} / \Delta\tau \rceil$ times:

$$\begin{cases} \text{(i)} \quad \boldsymbol{x}_i^{n+1} = \boldsymbol{x}_i^n + \Delta\tau\, \boldsymbol{v}_i^n + \frac{1}{2}\Delta\tau^2\, \boldsymbol{a}_i^n \text{ for all } i = 1, \cdots, N. \\ \text{(ii)} \quad \boldsymbol{v}_i^{n+1/2} = \boldsymbol{v}_i^n + \frac{1}{2}\Delta\tau\, \boldsymbol{a}_i^n \text{ for all } i = 1, \cdots, N. \\ \text{(iii)} \quad \text{Compute accelerations } \boldsymbol{a}_i^{n+1} \text{ for all } i = 1, \cdots, N \text{ as in (9),} \\ \qquad \text{using the positions } \boldsymbol{x}_j^{n+1}. \\ \text{(iv)} \quad \boldsymbol{v}_i^{n+1} = \boldsymbol{v}_i^{n+1/2} + \frac{1}{2}\Delta\tau\, \boldsymbol{a}_i^{n+1} \text{ for all } i = 1, \cdots, N. \end{cases} \tag{12}$$

Remark 1 (Reduced units, time scales, and cutoff-potential)

(i) The MD simulations using the Velocity-Verlet algorithm (12) are done in reduced units according to Table 1 below. With this procedure, we follow [2]. It avoids floating point over-/underflow in performing the numerical algorithm.

The number of particles N, the time step $\Delta\tau$ (in reduced units), and the number of time steps N_{end} will enter among others as parameters in Algorithm 1 for solving an MD Riemann problem. The precise values of the method's parameters are listed in Table A1. These parameters are chosen in a way that allows a reliable extraction of Riemann data in the final multiscale method which acts on the continuum scale. Note that we expect the MD Riemann problem to have a self-similar solution such that we can extract data at any point in time. Without any information about numerical errors the ad hoc choice of these parameters remains a weakness of the entire multiscale approach.

(ii) The application of the algorithm (12) leads to a quadratic complexity in terms of the total particles number N. This is due to the particle-wise interaction in (10). A remedy is the cut-off potentials. For a two-component mixture, we substitute ϕ_{ab} for $a, b \in \{0, 1\}$ by

$$\phi_{ab}(r; r_{\text{cutoff}}) := \begin{cases} \phi_{ab}(r): \ r < \sigma_{ab} r_{\text{cutoff}}, \\ 0: \ r \geqslant \sigma_{ab} r_{\text{cutoff}}. \end{cases} \tag{13}$$

Like all numerical parameters for the MD simulations, the actually used value for the cut-off radius r_{cutoff} can be found in Table A1 in the appendix. The error introduced by the cutoff functional will be corrected by the introduction of long-range interaction forces on the discrete level. We refer to [28] for a special choice for two-component flows.

We are interested in running MD simulations that provide an approximation of the solution of the Riemann problem (5), (7). For this purpose, the $\boldsymbol{n}$-rotated Riemann initial data $\boldsymbol{u}_{\pm}$ from (7) have to be represented as initial data for (9), i.e., we need an initial particle and velocity distribution in 3D. In [32], we have suggested a linked-cell algorithm to realize this transfer of single-component, two-phase constant Riemann data into a cubic domain Σ that consists of two cuboids $\Sigma_{\pm}$ separated by the atomistic interface. The algorithm includes a thermalization process that makes sure that the MD particle

Table 1 Unit conversion table

	Reduced units	SI units
Mass	1 u	1.66×10^{-27} kg
Length	1 σ	10^{-10} m
Energy	1 ε	1.38×10^{-23} J
Velocity	1 $\varepsilon^{0.5}$ u$^{-0.5}$	91.16 m $\cdot$ s^{-1}
Time	1 σ u$^{0.5}$ $\varepsilon^{-0.5}$	1.10×10^{-12} s
Pressure	1ε σ^{-3}	1.38×10^{7} Pa
Temperature	1ε k_{B}^{-1}	1 K
Density	1u σ^{-3}	1.66×10^{3} kg $\cdot$ m^{-3}

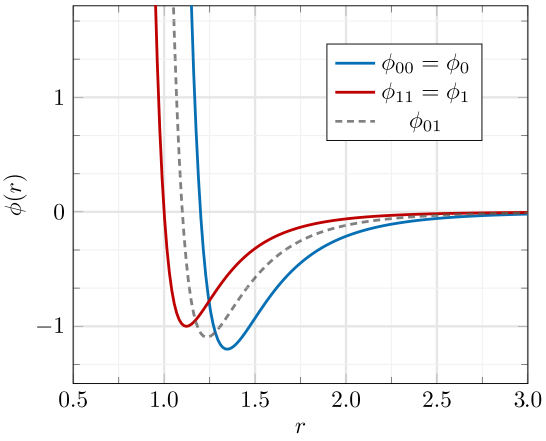

Fig. 3 Two-component Lennard-Jones potential for the parameters $\varepsilon_0 = 1.2$, $\sigma_0 = 1.2$, and $\varepsilon_1 = 1$, $\sigma_1 = 1$. The two solid-line graphs display intra-component interaction, and the dashed one the interaction potential between the two components, using the combination rules (11)

and particle velocity distributions match the constant reference temperature T, see (15) below for the definition of atomistic-scale temperatures. The linked-cell algorithm can be readily applied to the two-component case and is not further detailed.

In the next step, the system (9) is solved numerically by the scheme (12). We can again utilize a procedure from [32] to determine the approximate position of the atomistic interface and in particular an approximate interface speed $s \in \mathbb{R}$ like in the exact interface solver $\mathcal{R}$ from (8). To obtain all output data in (8) we need to find approximations of adjacent continuum-scale states $\boldsymbol{U}_{\pm}^{*}$, $\boldsymbol{u}_{\pm}^{*}$, respectively. Thus, we need to define continuum-scale quantities from the MD-simulation data.

Let us consider a system of N particles. The component-wise mass concentrations $\widehat{\rho}_0(\boldsymbol{x}, \tau)$, $\widehat{\rho}_1(\boldsymbol{x}, \tau)$ [u·Å^{-3}], and momentum concentrations $\widehat{\boldsymbol{m}}_0(\boldsymbol{x}, \tau)$, $\widehat{\boldsymbol{m}}_1(\boldsymbol{x}, \tau)$ [K$^{0.5}$ ·u$^{0.5}$· Å^{-3}] can be computed by the Irving-Kirkwood formulas in the following way, for $a \in \{0, 1\}$,

$$\widehat{\rho}_a(\boldsymbol{x}, \tau) = \sum_{\substack{i=1 \\ \varsigma_i = a}}^{N} m_i \, \delta_{\boldsymbol{x}_i(\tau)}(\boldsymbol{x}), \qquad \widehat{\boldsymbol{m}}_a(\boldsymbol{x}, \tau) = \sum_{\substack{i=1 \\ \varsigma_i = a}}^{N} m_i \, \boldsymbol{v}_i(\tau) \, \delta_{\boldsymbol{x}_i(\tau)}(\boldsymbol{x}), \tag{14}$$

where m_i, $\boldsymbol{x}_i$, $\boldsymbol{v}_i$ are the mass, position, and velocity of the i-th particle. The local temperature distribution of the mixture can be formally understood as

$$\widehat{T}(\boldsymbol{x}, \tau) = \frac{1}{3} \left(\sum_{i=1}^{N} \delta_{\boldsymbol{x}_i(\tau)}(\boldsymbol{x}) \right)^{-1} \sum_{i=1}^{N} (m_i (\boldsymbol{v}_i - \overline{\boldsymbol{v}}) \cdot (\boldsymbol{v}_i - \overline{\boldsymbol{v}})) \, \delta_{\boldsymbol{x}_i(\tau)}(\boldsymbol{x}), \tag{15}$$

where $\overline{\boldsymbol{v}} = \overline{\boldsymbol{v}}(\boldsymbol{x}, \tau)$ [K$^{0.5}$ ·u$^{-0.5}$] denotes the local barycentric average velocity of the mixture. The local average velocity can be realized by, e.g.,

$$\overline{\boldsymbol{v}}(\boldsymbol{x}, \tau) = \left(\sum_{i=1}^{N} K^{\epsilon}\left(\left| \boldsymbol{x} - \boldsymbol{x}_i(\tau) \right| \right) \right)^{-1} \left(\sum_{i=1}^{N} \boldsymbol{v}_i(\tau) K^{\epsilon}\left(\left| \boldsymbol{x} - \boldsymbol{x}_i(\tau) \right| \right) \right), \tag{16}$$

where $K^\epsilon : \mathbb{R} \to \mathbb{R}$ is a smooth kernel with length-scale parameter $\epsilon > 0$. It cannot be expected that the temperature field $\widehat{T}$ from (15) equals the constant temperature T in the isothermal system (1). This can be partially circumvented by applying a thermostat, but we will observe in Sect. 5 that substantial discrepancies can occur for the vapor-phase region.

Using the formulas (14), we obtain the adjacent states by sampling over small spatial slices $\Sigma^*_\pm$ next to the interface. That means we compute for $a \in \{0, 1\}$

$$\begin{cases} \rho^*_{a,\pm}(\tau) = \mathrm{vol}(\Sigma^*_\pm)^{-1} \sum\limits_{x_i \in \Sigma^*_\pm, \varsigma_i = a} m_i, \\ \boldsymbol{v}^*_{a,\pm}(\tau) = \left| x_i \in \Sigma^*_\pm, \varsigma_i = a \right|^{-1} \sum\limits_{x_i \in \Sigma^*_\pm, \varsigma_i = a} \boldsymbol{v}_i, \end{cases} \tag{17}$$

where $\mathrm{vol}(\Sigma^*_\pm)$ denotes the volume of the sampling region $\Sigma^*_\pm$, and $\left| x_i \in \Sigma^*_\pm, \varsigma_i = a \right|$ the number of component-a particles inside it. We can obtain the local interface states for every time step during an MD simulation. For the final outcome we perform time-averaging of these instantaneous states over a fraction $\alpha_{\tau\text{-smpl}}$ of the total simulation time τ_{end}. Note that the conversion of MD quantities and continuum-scale quantities has to account for the unit conversion given in Table 1.

We summarize all steps for solving an MD-Riemann problem in Algorithm 1, see also Algorithm 2 in the context of the multiscale method.

Algorithm 1 Atomistic-Scale Interface Solver for Binary Mixtures

Input: initial states $(\rho_{0,-}, \rho_{1,-}, \mathbf{m}_{0,-}, \mathbf{m}_{1,-}) \in \mathcal{P}_-$, $(\rho_{0,+}, \rho_{1,+}, \mathbf{m}_{0,+}, \mathbf{m}_{1,+}) \in \mathcal{P}_+$, normal direction $\mathbf{n} \in \mathbb{S}^{d-1}$.
Parameters: total number of particles $N_{\mathrm{particles}}$, number of time steps N_{end}, time step $\Delta\tau > 0$, time sampling ratio $\alpha_{\tau-\mathrm{smpl}} \in [0, 1]$, sampling regions $\Sigma^*_\pm$.

Algorithm

- Compute initial particle configuration with linked-cell algorithm from [1] starting from the $\mathbf{n}$-rotated Riemann data $\mathbf{u}_\pm = (\rho_{0,\pm}, \rho_{1,\pm}, v_{0,\pm}, v_{1,\pm})^\top$.
- For $n = 1, \cdots, N_{\mathrm{end}}$:
  - Run method (12), using the interaction potential (13).
  - Compute the interface positions $\Gamma(\tau_n)$ and interface speeds $s(\tau_n)$.
  - Compute continuum-scale states $(\rho^*_{0,\pm}, \rho^*_{1,\pm}, \mathbf{v}^*_{0,\pm}, \mathbf{v}^*_{1,\pm})(\tau_n)$ like in (17).
- Compute the time averaged values $(\rho^*_{0,\pm}, \rho^*_{1,\pm}, v^*_{0,\pm}, v^*_{1,\pm})$, s from $(\rho^*_{0,\pm}, \rho^*_{1,\pm}, \mathbf{v}^*_{0,\pm}, \mathbf{v}^*_{1,\pm})(\tau_n)$, $s(\tau_n)$ for all $n = \widetilde{\alpha}_{\tau-\mathrm{smpl}} N_{\mathrm{end}}, \cdots, N_{\mathrm{end}}$.

Result: interface states $(\rho^*_{0,-}, \rho^*_{1,-}, v^*_{0,-}, v^*_{1,-})$, $(\rho^*_{0,+}, \rho^*_{1,+}, v^*_{0,+}, v^*_{1,+})$, and interface speed s.

Note that the states $v^*_{a,\pm}(\tau_n)$ are computed from $\boldsymbol{v}^*_{a,\pm}(\tau_n)$ by the back projection $P_{\mathbf{n}}$, and that $\tau_n = n\Delta\tau$, $n \in \{0, \cdots, N_{\text{end}}\}$.

3.2 Notes on the Argon-Methane Mixture

Up to now, we considered a generic binary mixture. In the following, we focus on two-component mixtures consisting of argon and methane. In this case, parameters for the Lennard-Jones potential (10) can be found in [41], and even experimental data are available [35]. For the Lennard-Jones parameters in (11) we take the values from [41] listed in Table 2. The corresponding combination parameters in (11) are $\eta = 1.001\,41$ and $\xi = 0.964\,00$, see also [41]. For methodological background on the used approach, we refer to [42].

Remark 2 (Temperature parameterization of Lennard-Jones parameters) The chosen Lennard-Jones parameters are given in [41] for temperatures $T = 111$ K (argon) and $T = 140$ K (methane). The combination parameters refer to the temperature $T = 115$ K and have also been computed in [41]. In this work, the quantities have been shown to be in good fit with available experimental data in [35]. Up to our knowledge this setting is commonly used for temperatures well below the critical temperature of the mixture. Notably, our HMM method applies only for settings where the phase boundary can be expected to be a sharp interface separating the liquid and the vapor regions. Close to the critical temperature this separation property would fail.

The studies in the literature refer solely to the homogeneous case, i.e., they do not span the entire two-phase state space. In [28] it has been shown by extended MD simulations for homogeneous data covering the whole state space, that the predictions of the PC-SAFT theory for the chemical potentials μ_0, μ_1 are in good quantitative agreement with the MD simulations using the Lennard-Jones potentials for argon-methane mixtures. This justifies to use the PC-SAFT chemical potentials for the continuum-scale computations in the HMM which will be proposed in Sect. 4.

Table 2 Lennard-Jones parameters from [41] and species masses in reduced units for the argon-methane mixture, cf. Table 1

Component		σ [Å]	ε/k_B [K]	m [u]
Argon	Ar	3.396 7	117.05	39.948
Methane	CH_4	3.727 5	148.99	16.043

4 An HMM for Isothermal Two-Component, Two-Phase Flow

Following [32] for the single-component flow with phase transitions, we introduce a corresponding HMM that combines the isothermal two-component flow model on the continuum scale (2) with two-component MD simulations described in Sect. 3 for the microscale description of the interface motion.

Here, we focus again on a fluid mixture consisting of argon and methane, see Sect. 2. The component-specific quantities are marked by the indices 0 for argon, and 1 for methane. The two fluid phases are again denoted by the subscripts $-$ and $+$ for the liquid and vapor phases.

First, we formulate on the basis of Algorithm 1 the MD-based microscale interface solver $\mathcal{R}_{\mathrm{MD}}$ which provides an approximate solution for Riemann initial data (Sect. 4.1). Using an MD-based interface solver directly in place is still too computationally expensive. Therefore we suggest the machine-learned surrogate solver $\mathcal{R}_{\boldsymbol{\theta}}$ in Sect. 4.2, which exploits the fact that the interface solver $\mathcal{R}_{\mathrm{MD}}$ can be understood as a nonlinear function mapping finite-dimensional Riemann data to a finite-dimensional space including the phase boundary speed and the state values adjacent to the phase boundary. Finally, in Sect. 4.3 we present the complete HMM for the two-component, two-phase flow. For the sake of simplicity we ignore in the entire section boundary conditions on $\partial\boldsymbol{\Omega}$.

4.1 The Atomistic-Scale Interface Solver $\mathcal{R}_{\mathrm{MD}}$

To reduce the input space of the atomistic-scale interface solver from Algorithm 1 we exploit that the continuum-scale system (1) is invariant with respect to velocity shifts. We define the barycentric velocities

$$\boldsymbol{v}_{\pm} := \frac{\boldsymbol{m}_{0,\pm} + \boldsymbol{m}_{1,\pm}}{\rho_{0,\pm} + \rho_{1,\pm}} \tag{18}$$

and choose $\bar{v} := \boldsymbol{v}_{-}$ as the reference velocity. Furthermore, we introduce the relative velocity for each phase

$$\boldsymbol{v}_{\mathrm{rel},\pm} := \boldsymbol{v}_{0,\pm} - \boldsymbol{v}_{1,\pm}. \tag{19}$$

Assuming we know the densities $\rho_{0,\pm}$, $\rho_{1,\pm}$, we can compute $(\boldsymbol{v}_{0,\pm}, \boldsymbol{v}_{1,\pm})$ from $(\boldsymbol{v}_{\pm}, \boldsymbol{v}_{\mathrm{rel},\pm})$ and vice versa. In the sequel this conversion is done implicitly to simplify the notation.

Algorithm 2 implements the final MD-interface solver $\mathcal{R}_{\mathrm{MD}}: \mathcal{P}_{-} \times \mathcal{P}_{+} \times \mathbb{S}^{d-1} \to \mathcal{P}_{-} \times \mathcal{P}_{+} \times \mathbb{R}$ substituting the exact Riemann solver $\mathcal{R}$.

Algorithm 2 MD-Interface Solver $\mathcal{R}_{\text{MD}}$: Two-Component, Two-Phase Flow

Input: initial states $(\rho_{0,-}, \rho_{1,-}, \mathbf{m}_{0,-}, \mathbf{m}_{1,-}) \in \mathcal{P}_-$, $(\rho_{0,+}, \rho_{1,+}, \mathbf{m}_{0,+}, \mathbf{m}_{1,+}) \in \mathcal{P}_+$, normal direction $\mathbf{n} \in \mathbb{S}^{d-1}$.

Algorithm

(i) Compute the barycentric and relative phase velocities $\mathbf{v}_\pm$, $\mathbf{v}_{\text{rel},\pm}$ according to (18), (19).

(ii) Map $(\rho_{0,\pm}, \rho_{1,\pm}, \mathbf{v}_\pm, \mathbf{v}_{\text{rel},\pm})$ in direction of $\mathbf{n}$ according to the formula in (5),

$$\mathbf{u}_\pm = (\rho_{0,\pm}, \rho_{1,\pm}, v_\pm, v_{\text{rel},\pm}) \coloneqq \mathbf{U}_{\|\mathbf{n},\pm} = (\rho_\pm, \mathbf{v}_\pm \cdot \mathbf{n}, \mathbf{v}_{\text{rel},\pm} \cdot \mathbf{n}), \tag{20}$$

and define

$$\mathbf{U}_{\perp\mathbf{n},\pm} \coloneqq (0, 0, \mathbf{v}_\pm - (\mathbf{v}_\pm \cdot \mathbf{n})\mathbf{n}, \mathbf{v}_{\text{rel},\pm} - (\mathbf{v}_{\text{rel},\pm} \cdot \mathbf{n})\mathbf{n}).$$

(iii) Consider the barycentric -phase velocity as the reference velocity $\overline{v} \coloneqq v_-$ and map it:

$$v_\pm \leftarrow\!\shortmid v_\pm - \overline{v}. \tag{21}$$

(iv) Solve the Riemann problem (5) with Riemann data $(\rho_{0,\pm}, \rho_{1,\pm}, \mathbf{v}_\pm, \mathbf{v}_{\text{rel},\pm})$ using the MD model. This means, we run a two-component MD simulation for the initial continuum states $(\rho_{0,-}, \rho_{1,-}, \mathbf{0}, \mathbf{v}_{\text{rel},-})$ and $(\rho_{0,+}, \rho_{1,+}, \mathbf{v}_+, \mathbf{v}_{\text{rel},+})$, see Algorithm 1. This yields the wave states $(\rho^*_{0,-}, \rho^*_{1,-}, v^*_{0,-}, v^*_{1,-})$, $(\rho^*_{0,+}, \rho^*_{1,+}, v^*_{0,+}, v^*_{1,+})$, and the wave speed s in normal direction $\mathbf{n}$.

(v) Return the reference velocity $\overline{v}$ to the output:

$$\begin{aligned} m^*_{0,\pm} &\leftarrow\!\shortmid m^*_{0,\pm} + \rho^*_{0,\pm}\overline{v}, \\ m^*_{1,\pm} &\leftarrow\!\shortmid m^*_{1,\pm} + \rho^*_{1,\pm}\overline{v}, \\ s &\leftarrow\!\shortmid s + \overline{v}. \end{aligned}$$

(vi) Project the directional states back to the full state:

$$\begin{aligned} \mathbf{U}^*_\pm \coloneqq (\rho^*_{0,\pm}, \rho^*_{1,\pm}, \mathbf{m}^*_{0,\pm}, \mathbf{m}^*_{1,\pm}) &= (\rho^*_{0,\pm}, \rho^*_{1,\pm}, m^*_{0,\pm}\mathbf{n}, m^*_{1,\pm}\mathbf{n}) + \mathbf{U}_{\perp\mathbf{n},\pm} \\ &= P_{\mathbf{n}}((\rho^*_{0,\pm}, \rho^*_{1,\pm}, m^*_{0,\pm}, m^*_{1,\pm})) + \mathbf{U}_{\perp\mathbf{n},\pm} \end{aligned}$$

with $P_{\mathbf{n}}$ defined as in (6).

Result: $\mathcal{R}_{\text{MD}}(\mathbf{U}_-, \mathbf{U}_+; \mathbf{n}) = (\mathbf{U}^*{}_-, \mathbf{U}^*_+, s)$.

Due to the velocity shift described before, we can always achieve that the component $\boldsymbol{m}_{0,-}$ can be assumed to vanish when Step (iv) is executed within Algorithm 2.

Table 3 Density corner points for the two-component (argon-methane) model input data set

$\rho_{0,+}$ [kg · m^{-3}]	$\rho_{1,+}$ [kg · m^{-3}]	$\rho_{0,-}$ [kg · m^{-3}]	$\rho_{1,-}$ [kg · m^{-3}]
1.000 000	0.000 000	1 024.214 739	0.000 000
1.000 000	0.000 000	0.000 000	343.403 446
1.000 000	0.000 000	1 616.252 845	0.000 000
1.000 000	0.000 000	0.000 000	509.380 478
0.000 000	1.000 000	1 024.214 739	0.000 000
0.000 000	1.000 000	0.000 000	343.403 446
0.000 000	1.000 000	1 616.252 845	0.000 000
0.000 000	1.000 000	0.000 000	509.380 478
162.996 622	0.000 000	1 024.214 739	0.000 000
162.996 622	0.000 000	0.000 000	343.403 446
162.996 622	0.000 000	1 616.252 845	0.000 000
162.996 622	0.000 000	0.000 000	509.380 478
0.000 000	30.960 615	1 024.214 739	0.000 000
0.000 000	30.960 615	0.000 000	343.403 446
0.000 000	30.960 615	1 616.252 845	0.000 000
0.000 000	30.960 615	0.000 000	509.380 478

4.2 The Machine-Learned Interface Solver $\mathcal{R}_{\boldsymbol{\theta}}$

To reduce the computational complexity of Algorithm 2, we employ a machine-learned surrogate interface solver $\mathcal{R}_{\boldsymbol{\theta}}$ that approximates $\mathcal{R}_{\mathrm{MD}}$, i.e.,

$$\mathcal{R}_{\boldsymbol{\theta}}(\boldsymbol{U}_-, \boldsymbol{U}_+; \boldsymbol{n})) \approx \mathcal{R}_{\mathrm{MD}}((\boldsymbol{U}_-, \boldsymbol{U}_+; \boldsymbol{n}) \approx \mathcal{R}((\boldsymbol{U}_-, \boldsymbol{U}_+; \boldsymbol{n}). \tag{22}$$

For the full multiscale model, the surrogate model $\mathcal{R}_{\boldsymbol{\theta}}$ replaces the molecular-scale interface solver $\mathcal{R}_{\mathrm{MD}}$ in Step (iv) of Algorithm 2. We note that the implementation of $\mathcal{R}_{\boldsymbol{\theta}}$ relies on the reduced-dimensional state spaces in the rotated Riemann problem (5) and using the invariance with respect to the reference velocity $\boldsymbol{v}_-$ in (18).

The final surrogate solver offers large computational gains: it takes only around 0.10 ms for a single evaluation. The MD simulations (Algorithm 2) on the other hand needs 14 min to 17 min.

4.2.1 Data Set Generation

To prepare the surrogate interface solver, we have to generate a data set D for the input-output relation of the microscale interface solver $\mathcal{R}_{\mathrm{MD}}$. The range of the $\pm$-phase densities $\rho_{0,\pm}$, $\rho_{1,\pm}$ for each component is defined by the convex set and is formed by the points given in Table 3. The barycentric velocity v_+ ranges from $v_{\min} = -750\,\mathrm{m} \cdot \mathrm{s}^{-1}$ to $v_{\max} = 750\,\mathrm{m} \cdot \mathrm{s}^{-1}$. The relative velocities $v_{\mathrm{rel},\pm}$ are bounded by $v_{\mathrm{rel},\min} = -500\,\mathrm{m} \cdot \mathrm{s}^{-1}$ and $v_{\mathrm{rel},\max} = 500\,\mathrm{m} \cdot \mathrm{s}^{-1}$. Taken all together, the resulting convex set forms the input bounding domain B_{in}.

Using distance-maximizing sampling similar to [34], we generate $N_{\mathrm{data}} = 12\,000$ samples in B_{in}, while exploiting the reduced dimensionality, due to rotational invariance and $v_- \equiv 0$. Consequently, we generate the input data set

$$D_{\mathrm{in}} = \{(\rho_{0,-}, \rho_{1,-}, v_{\mathrm{rel},-}, \rho_{0,+}, \rho_{1,+}, v_+, v_{\mathrm{rel},+})_i : i = 1, \cdots, N_{\mathrm{data}}\}.$$

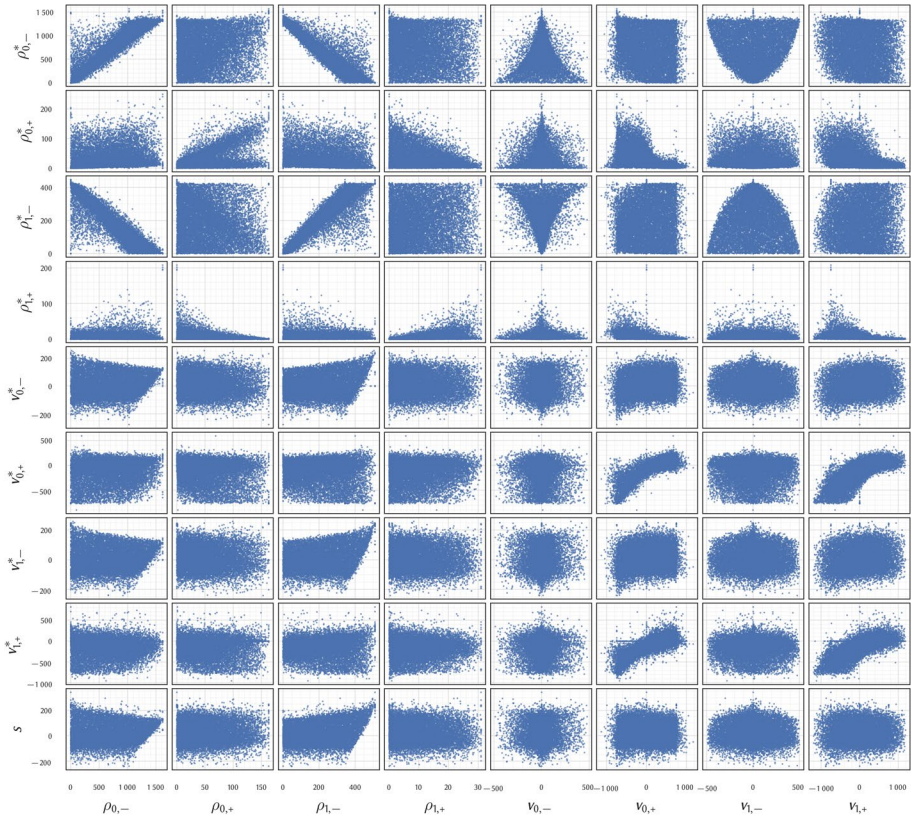

Fig. 4 The data set for the isothermal, two-component, two-phase flow multiscale model. The data set consists of 12 000 data points. The scatter plots show the relation of the input states $\rho_{0,-}$, $\rho_{1,-}$, $v_{0,-}$, $v_{1,-}$, $\rho_{0,+}$, $\rho_{1,+}$, $v_{0,+}$, $v_{1,+}$ to their respective wave states $\rho^*_{0,-}$, $\rho^*_{1,-}$, $v^*_{0,-}$, $v^*_{1,-}$, $\rho^*_{0,+}$, $\rho^*_{1,+}$, $v^*_{0,+}$, $v^*_{1,+}$, and wave speed s. The error-bars show the data range of each separate triplet of MD simulation

For each $(\rho_{0,-}, \rho_{1,-}, v_{\mathrm{rel},-}, \rho_{0,+}, \rho_{1,+}, v_+, v_{\mathrm{rel},+}) \in D_{\mathrm{in}}$ an MD simulation with the MD interface solver (Algorithm 2) is performed. It provides with the corresponding input variables the output

$$(\rho^*_{0,-}, \rho^*_{1,-}, m^*_{0,-}, m^*_{1,-}, \rho^*_{0,+}, \rho^*_{1,+}, m^*_{0,+}, m^*_{1,+}, s), \tag{23}$$

which is gathered into the output data set

$$D_{\mathrm{out}} = \{(\rho^*_{0,-}, \rho^*_{1,-}, m^*_{0,-}, m^*_{1,-}, \rho^*_{0,+}, \rho^*_{1,+}, m^*_{0,+}, m^*_{1,+}, s)_i : i = 1, \cdots, N_{\mathrm{data}}\}. \tag{24}$$

By associating each output data point with its input, we obtain the complete data set

$$D = \Big\{ \Big((\rho_{0,-}, \rho_{1,-}, v_{\mathrm{rel},-}, \rho_{0,+}, \rho_{1,+}, v_+, v_{\mathrm{rel},+}), \\ (\rho^*_{0,-}, \rho^*_{1,-}, m^*_{0,-}, m^*_{1,-}, \rho^*_{0,+}, \rho^*_{1,+}, m^*_{0,+}, m^*_{1,+}, s) \Big)_i : i = 1, \cdots, N_{\mathrm{data}} \Big\}.$$

Note that we run three MD simulations for each input data point. The data set D is visualized in Fig. 4. It is digitally archived at [29]. With the run time of 14–17 min for a single, two-component MD simulation, the generation of the whole data set D took approximately 3 200 h of computing time, which can be easily split among several of machines to decrease the real time until all data points are sampled.

4.2.2 Neural Network Training

To train the neural-network surrogate solver $\mathcal{R}_{\boldsymbol{\theta}}$, we use a standard training procedure, albeit with some model-specific parameters as described in the following. The network is comprised of 5 hidden layers with 60 nodes each. The data set D from Sect. 4.2.1 is split into a training data set D_{train} with 10 800 samples and a validation data set D_{val} with 1 200 samples.

Finally, we note that we use a CRes-neural network, which has been developed in [30] to incorporate conservation of selected state variables into the surrogate interface Riemann solver. For the two-component, two-phase flow we decided to preserve the mass conservation across the interface.

4.3 The Complete HMM

Using the machine-learned interface solver $\mathcal{R}_{\boldsymbol{\theta}}$ we present the final HMM and the interplay of each of its components. To see the integration of the interface solvers in detail, we describe the HMM in the framework of the numerical discretization, a moving-mesh finite-volume method. The method has been introduced for two space dimensions in [7] and open-source code for two and three space-dimensions can be found in [1].

We partition the continuum-scale time interval $[0, t_{\text{end}}]$ according to $0 = t^0 < \dots < t^N = t_{\text{end}}$, and start with an initial mesh $\mathcal{T}(0) = \{T_i^0 : i \in I^0\}$ for $\boldsymbol{\Omega}$ that is assumed to consist of simplices T_i indexed via some index set I^0. Let the mesh parameter $h > 0$ be given by the maximum of all edges' length. The mesh $\mathcal{T}(0)$ is supposed to include a connected set of mesh facets of the simplices, which form the approximation of the initial interface $\boldsymbol{\Gamma}_h(0)$, and thus discrete liquid and vapor bulk domains $\boldsymbol{\Omega}_{h,\pm}(0)$. The initial datum $\boldsymbol{U}_0$ from (3) is projected to the mesh $\mathcal{T}(0)$ giving

$$\boldsymbol{U}_i^0 = \int_{T_i^0} \boldsymbol{U}_0(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x} \qquad (i \in I^0). \tag{25}$$

We assume that we have $\boldsymbol{U}_i^0 \in \mathcal{P}_\pm$ for $T_i^0 \in \boldsymbol{\Omega}_{h,\pm}(0)$.

To introduce the finite-volume time-stepping let us assume that for some discrete time t^n, $n \in \{1, \cdots, N-1\}$, we have the same situation, i.e., a simplicial mesh $\mathcal{T}(t^n) = \{T_i^n : i \in I^n\}$ for $\boldsymbol{\Omega}$, a discrete partition $\boldsymbol{\Omega} = \boldsymbol{\Omega}_{h,-}(t^n) \cup \boldsymbol{\Gamma}_h(t^n) \cup \boldsymbol{\Omega}_{h,+}(t^n)$ with the discrete interface

$$\boldsymbol{\Gamma}_h(t^n) = \{S_{ij}^n : i,j \in I^n, T_i^n \in \boldsymbol{\Omega}_{h,-}(t^n), T_j^n \in \boldsymbol{\Omega}_{h,-}(t^n)\} \tag{26}$$

of connected facets, and a set of cell averages $\{\boldsymbol{U}_i^n\}_{i\in I^n}$ with the property $\boldsymbol{U}_i^n \in \mathcal{P}_\pm$ for $T_i^n \in \boldsymbol{\Omega}_{h,\pm}(t^n)$. Furthermore, for any facet from $\boldsymbol{\Gamma}_h(t^n)$ with normal $\boldsymbol{n}_{ij}^n \in \mathbb{S}^{d-1}$ (pointing to the vapor domain $\boldsymbol{\Omega}_{h,+}(t^n)$), we can compute

$$(\boldsymbol{U}_i^*, \boldsymbol{U}_j^*, s_{ij}^n) = \mathcal{R}_{\boldsymbol{\theta}}(\boldsymbol{U}_i^n, \boldsymbol{U}_j^n; \boldsymbol{n}_{ij}^n).$$

Using the computed speeds s_{ij}^n, the moving-mesh finite-volume method from [7] delivers a mesh $\widetilde{\mathcal{T}}(t^{n+1}) = \{\widetilde{T}_i^{n+1} : i \in \widetilde{I}^{n+1}\}$, a discrete partition $\boldsymbol{\Omega} = \widetilde{\boldsymbol{\Omega}}_{h,-}(t^{n+1}) \cup \widetilde{\boldsymbol{\Gamma}}_h(t^{n+1}) \cup \widetilde{\boldsymbol{\Omega}}_{h,+}(t^{n+1})$, and the family $\{\widetilde{\boldsymbol{U}}_i^{n+1}\}_{i \in I^{n+1}}$ with the properties as in time-step t^n. The new mesh $\widetilde{\mathcal{T}}(t^{n+1})$ evolves from $\mathcal{T}(t^{n+1})$ by affine shifts. To avoid small cells that would deteriorate the time-step, this mesh deformation is followed by a re-meshing leading finally to the new approximations at time t^{n+1}, i.e., the mesh

$$\mathcal{T}(t^{n+1}) = \{T_i^{n+1} : i \in I^{n+1}\},\ \boldsymbol{\Omega} = \boldsymbol{\Omega}_{h,-}(t^{n+1}) \cup \boldsymbol{\Gamma}_h(t^{n+1}) \cup \boldsymbol{\Omega}_{h,-}(t^{n+1}),$$

and the finite-volume approximations

$$\{\boldsymbol{U}_i^{n+1}\}_{i \in I^{n+1}}.$$

A particular point in the moving-mesh finite-volume method is the choice of the numerical flux functions. They can be chosen arbitrarily for facets not in $\boldsymbol{\Gamma}_h(t^n)$ but have to be Godunov fluxes across $\boldsymbol{\Gamma}_h(t^n)$ using the adjacent states $\boldsymbol{U}_i^*, \boldsymbol{U}_j^*$. This avoids the occurrence of states not belonging to $\mathcal{P}_\pm$. For more details and the control of the explicit time-stepping we refer to [7].

We conclude the section with the complete HMM for the two-component, two-phase flow given in algorithmic form.

Algorithm 3 HMM for Two-Component, Two-Phase Flow

Input: Initial mesh $\mathcal{T}(0)$ of $\boldsymbol{\Omega}$ with partition $\boldsymbol{\Omega} = \boldsymbol{\Omega}_{h,-}(0) \cup \boldsymbol{\Gamma}_h(0), \boldsymbol{\Omega}_{h,+}(0)$, the family $\{\mathbf{U}_i^0\}_{i \in I^0}$ from (25) with $\mathbf{U}_i^0 \in \mathcal{P}_\pm$ for $T_i^0 \in \boldsymbol{\Omega}_{h,\pm}(0)$.
Prerequisite: the surrogate interface solver $\mathcal{R}_\theta$, based on $\mathcal{R}_{\mathrm{MD}}$.

Algorithm $n \to n+1$

- Approximate $\boldsymbol{\Omega}_{h,\pm}(0)$ and interface $\boldsymbol{\Gamma}_h(0)$ by mesh
- For each $S_{ij}^n \in \boldsymbol{\Gamma}_h(t^n)$, see (26), compute

$$(\mathbf{U}_i^*, \mathbf{U}_j^*, s_{ij}^n) = \mathcal{R}_\theta(\mathbf{U}_i^n, \mathbf{U}_j^n; \mathbf{n}_{ij}^n). \tag{27}$$

- Run the moving-mesh finite-volume method from [14, 15] with the mesh $\mathcal{T}(t^n)$, the partition $\boldsymbol{\Omega} = \boldsymbol{\Omega}_{h,-}(t^n) \cup \boldsymbol{\Gamma}_h(t^n) \cup \boldsymbol{\Omega}_{h,+}(t^n)$, and the family $\{\mathbf{U}_i^n\}_{i \in I^{n+1}}$.

Result: The mesh $\mathcal{T}(t^n)$, the partition $\boldsymbol{\Omega} = \boldsymbol{\Omega}_{h,-}(t^{n+1}) \cup \boldsymbol{\Gamma}_h(t^{n+1}) \cup \boldsymbol{\Omega}_{h,+}(t^{n+1})$, and the family $\{\mathbf{U}_i^{n+1}\}_{i \in I^{n+1}}$ with $\mathbf{U}_i^{n+1} \in \mathcal{P}_\pm$ for $T_i^{n+1} \in \boldsymbol{\Omega}_{h,\pm}(t^{n+1})$.

Algorithm 3 provides an approximation $\boldsymbol{U}_h : \boldsymbol{\Omega} \times [0, t_{\text{end}}] \to \mathcal{U}$ and an approximation $\boldsymbol{\Gamma}_h(t)$ for the exact interface $\boldsymbol{\Gamma}(t)$, $t \in [0, t_{\text{end}}]$, solving the free boundary value problem for (1). The approximate function $\boldsymbol{U}_h$ and the interface $\boldsymbol{\Gamma}_h$ are defined by

$$\boldsymbol{U}_h(\boldsymbol{x}, t) = \boldsymbol{U}_i^n, \quad \boldsymbol{\Gamma}_h(t) = \boldsymbol{\Gamma}_h(t^n) \quad (\boldsymbol{x} \in T_i^n,\ t \in [t^n, t^{n+1}]),$$

using the quantities computed in Algorithm 3. The machine-learned interface solver $\mathcal{R}_{\boldsymbol{\theta}}$ represents the atomistic microscale whereas the final approximation is valid on the continuum scale only.

Remark 3 (Discretization of source and non-conservative terms) The moving-mesh finite-volume method from [7, 32] uses explicit Euler time-stepping and deals with first-order systems in divergence form only. The discretization of the 0th-order Maxwell-Stefan diffusion terms is handled by simple evaluation of the cell averages.

More complicated is the discretization of the gradients $\nabla \mu_a$ in the right-hand-side term of the continuum-scale system (1).

In one space dimension, we simply apply central finite-differences in the bulk phases. At the cells adjacent to the interface, we use left-/right-sided finite differences, to avoid computing the gradient over the discrete phase boundary.

In two and three space dimensions, for the n-th time-step, we approximate the gradient $\nabla \mu_a$ in a simplex T_j^n by linear reconstruction. The stencil for the reconstruction includes those neighbors T_k^n of T_j^n that share a surface with T_j^n and belong to the same phase domain as T_j^n. In that way, we avoid mixing the phases during the reconstruction. The reconstruction is done by solving the linear least squares system

$$\boldsymbol{A}^\top \boldsymbol{A} \boldsymbol{\mu}_a' = \boldsymbol{A}^\top \boldsymbol{b}_a \tag{27}$$

with

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{c}_j - \boldsymbol{c}_k \\ \vdots \end{pmatrix} \in \mathbb{R}^{K \times d}, \quad \boldsymbol{b}_a = \begin{pmatrix} \mu_{a,j} - \mu_{a,k} \\ \vdots \end{pmatrix} \in \mathbb{R}^K, \quad T_k^n \text{ in stencil of } T_j^n,$$

where $\boldsymbol{c}_j \in \mathbb{R}^d$ denotes the cell center of T_j^n, $\mu_{a,j} \in \mathbb{R}$ the cell value of the chemical potential, and $K \in \mathbb{N}$ the number of neighbor cells in the stencil. The solution $\boldsymbol{\mu}_a'$ of (27) is used as an approximation of the gradient $\nabla \mu_a$ in cell T_j^n.

5 Numerical Simulations for the HMM

In this section, we present a series of numerical results to validate the multiscale method that has been introduced in Sect. 4 to solve the free boundary value problem for the two-component, two-phase flow model (1).

The model and numerical parameters used for the simulations in this section, if not otherwise stated, are found in Appendix A. We choose the Maxwell-Stefan diffusion coefficient to be $Đ_{01} = 1.0$. We refer to [23] for the computation of specific Maxwell-Stefan diffusion coefficients leading to much smaller values. Numerical tests for simplified settings have shown that the influence of $Đ_{01}$ is minor for the overall dynamics, which justifies our choice.

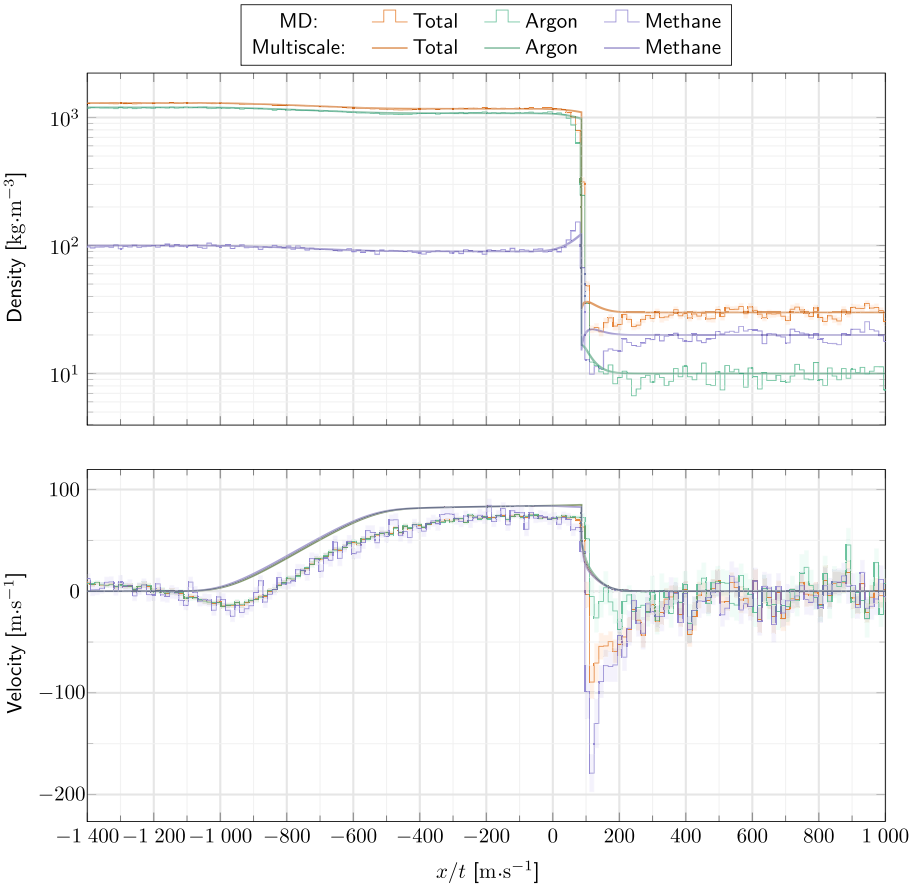

Fig. 5 Multiscale simulation for the isothermal, two-component, two-phase flow model in one space-dimension, overlaid over the corresponding MD simulations

5.1 One-Dimensional Simulation Results

We consider one-dimensional Riemann problems and compare the multiscale simulation results with MD simulations on the entire domain as reference. The section serves as validation for the multi-dimensional droplet simulations in the subsequent Sects. 5.2 and 5.3.

Example 1 (Pressure-driven shock wave) In the first example, we simulate a pressure-driven shock wave originating from an argon-rich liquid. This is set up by the initial data $\rho_{0,-} = 1\,200\,\mathrm{kg}\cdot\mathrm{m}^{-3}$, $\rho_{1,-} = 100\,\mathrm{kg}\cdot\mathrm{m}^{-3}$ on the liquid side, $\rho_{0,+} = 10\,\mathrm{kg}\cdot\mathrm{m}^{-3}$, $\rho_{1,+} = 20\,\mathrm{kg}\cdot\mathrm{m}^{-3}$ on the vapor side, and $\boldsymbol{v}_{0,\pm} = 0\,\mathrm{m}\cdot\mathrm{s}^{-1}$, $\boldsymbol{v}_{1,\pm} = 0\,\mathrm{m}\cdot\mathrm{s}^{-1}$. The simulation results are plotted in Fig. 5, including the corresponding two-component MD simulation results.

In the multiscale solution it can be observed that the liquid phase expands to the right side, while the total density inside the liquid phase decreases. Near the interface, argon moves from the liquid into the vapor phase, whereas methane accumulates inside the liquid, and decreases in the vapor phase.

By comparing the multiscale and the MD solution, we see that the interface position is captured accurately. Furthermore, on the liquid side, the MD simulation and the multiscale model behave qualitatively in the same manner. On the vapor phase side, we see that the argon component of the multiscale simulation is close to the MD simulation. In contrast to that, substantial deviations in the methane component can be observed. We assume that these deviations can be attributed to the isothermal continuum model which does not capture the whole range of the MD simulation. The latter is not perfectly isothermal allowing for small temperature variations. We observed the same effect for the isothermal flow of one component in [32]. Finally, the deviations in the low-density vapor phase might be caused by poor sampling on the MD scale, as few particles of each component are present near the interface. This problem can be solved by increasing the number of particles, as well as the size of the computational domain (see Sect. 3) for the MD simulations.

Example 2 (Colliding vapor wave) In the second example, we simulate a vapor wave that collides with a liquid argon-methane mixture. The corresponding Riemann initial data are $\rho_{0,-} = 440\,\mathrm{kg}\cdot\mathrm{m}^{-3}$, $\rho_{1,-} = 280\,\mathrm{kg}\cdot\mathrm{m}^{-3}$, $v_{0,-} = 0\,\mathrm{m}\cdot\mathrm{s}^{-1}$, $v_{1,-} = 0\,\mathrm{m}\cdot\mathrm{s}^{-1}$ for the liquid phase, and $\rho_{0,+} = 20\,\mathrm{kg}\cdot\mathrm{m}^{-3}$, $\rho_{1,+} = 2\,\mathrm{kg}\cdot\mathrm{m}^{-3}$, $v_{0,+} = -50\,\mathrm{m}\cdot\mathrm{s}^{-1}$, $v_{1,+} = -50\,\mathrm{m}\cdot\mathrm{s}^{-1}$ for the vapor phase. The multiscale simulation results are plotted in Fig. 6, alongside with their respective MD simulation. It can be seen that the vapor wave transmits into the liquid phase, and increases the liquid density slightly. Furthermore, the wave speeds, as well as the interface speed, are captured very well by the multiscale model. As in Example 1, we observe deviations for the velocity in the vapor-phase domain.

5.2 Two-Dimensional Simulation Results

Example 3 (Interaction shock wave/droplet) In this section, we present a two-dimensional, two-component multiscale simulation, where a methane droplet in an argon vapor atmosphere is hit by a shock wave. To this end, we consider the domain $\boldsymbol{\Omega} = [-1.5, 1.5]^2$, which is split into a liquid droplet $\boldsymbol{\Omega}_-(0) = \{\boldsymbol{x} \in \mathbb{R}^2 \colon \|\boldsymbol{x}\|_2^2 < 0.15\}$ and the surrounding vapor domain $\boldsymbol{\Omega}_+(0) = \boldsymbol{\Omega} \backslash \overline{\boldsymbol{\Omega}_-(0)}$. The initial conditions are given by

$$(\rho_0, \boldsymbol{v}_0, \rho_1, \boldsymbol{v}_1)(\boldsymbol{x}, t) = \begin{cases} (180, (0,0), 400, (0,0)), & \boldsymbol{x} \in \boldsymbol{\Omega}_-(0), \\ (20, (0,0), 4, (0,0)), & \boldsymbol{x} \in \boldsymbol{\Omega}_+(0) \text{ and } x \geqslant -0.5, \\ (20, (150,0), 4, (150,0)), & \boldsymbol{x} \in \boldsymbol{\Omega}_+(0) \text{ and } x < -0.5. \end{cases}$$

On the left side, at $x = -1.5$, we apply inflow boundary conditions by setting the ghost cell value to $(\rho_0, \boldsymbol{v}_0, \rho_1, \boldsymbol{v}_1) = (20, (150,0), 4, (150,0))$. On the opposing side, at $x = 1.5$, we use outflow boundary conditions. At the top and bottom, i.e., at $y = \pm 1.5$, reflecting boundary conditions simulate closed walls.

The simulation results are depicted in Figs. 7 and 8. In the first figure, the component-wise densities and velocities are shown, and in the second figure the argon-mole fraction is plotted.

In the beginning, the liquid and vapor phases are not in equilibrium, resulting in small oscillations of the droplet. The oscillations can be clearly observed in the time evolution of the averaged quantities, as illustrated in Fig. 9. Then, the vapor wave hits the droplet, causing a ripple moving through its surface and finally pushing it through the vapor atmosphere. Throughout the simulation, argon accumulates inside the liquid phase, leading to a growth of the droplet; see Fig. 9.

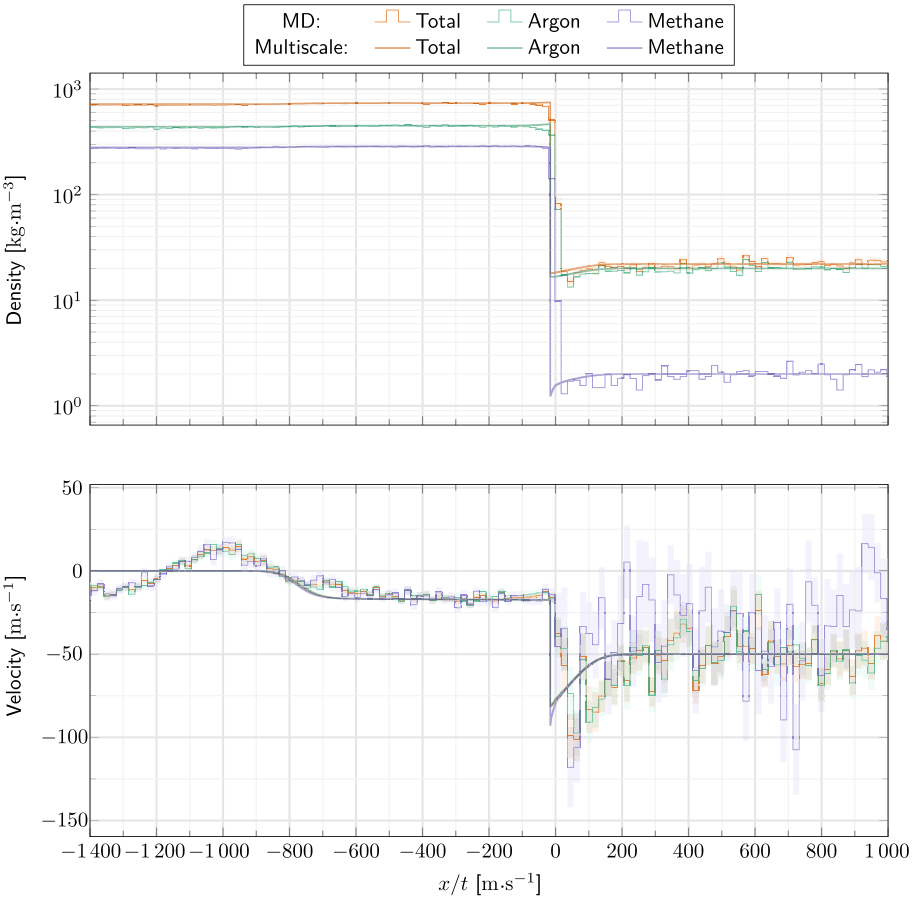

Fig. 6 Multiscale simulation for the isothermal, two-component, two-phase flow model in one space dimension, overlaid over the corresponding MD simulations. We observe a vapor wave colliding with a liquid argon-methane mixture

5.3 Three-Dimensional Simulation Results

Example 4 (Interaction shock wave/droplet) The last example in this section is a three-dimensional version of Example 3, i.e., a simulation of a liquid droplet, that consists mostly of methane, inside an argon-methane vapor atmosphere. The domain $\boldsymbol{\Omega} = [-5, 5]^3$ is split into a liquid droplet $\boldsymbol{\Omega}_-(0) = \{\boldsymbol{x} \in \mathbb{R}^3 : \|\boldsymbol{x}\|_2 < 2\}$ and the surrounding vapor domain $\boldsymbol{\Omega}_+(0) = \boldsymbol{\Omega} \backslash \overline{\boldsymbol{\Omega}_-(0)}$. For the initial data, we set

$$(\rho_0, \boldsymbol{v}_0, \rho_1, \boldsymbol{v}_1)(\boldsymbol{x}, t) = \begin{cases} (180, (0,0,0), 400, (0,0,0)), & \boldsymbol{x} \in \boldsymbol{\Omega}_-(0), \\ (20, (0,0,0), 4, (0,0,0)), & \boldsymbol{x} \in \boldsymbol{\Omega}_+(0) \text{ and } x \geqslant -3, \\ (20, (200,0,0), 4, (200,0,0)), & \boldsymbol{x} \in \boldsymbol{\Omega}_+(0) \text{ and } x < -3. \end{cases}$$

Inflow boundary conditions are applied at $x = -5$, by setting the ghost cell values to $(\rho_0, \boldsymbol{v}_0, \rho_1, \boldsymbol{v}_1) = (20, (200, 0, 0), 4, (200, 0, 0))$. Opposed to that side, at $x = 5$, outflow

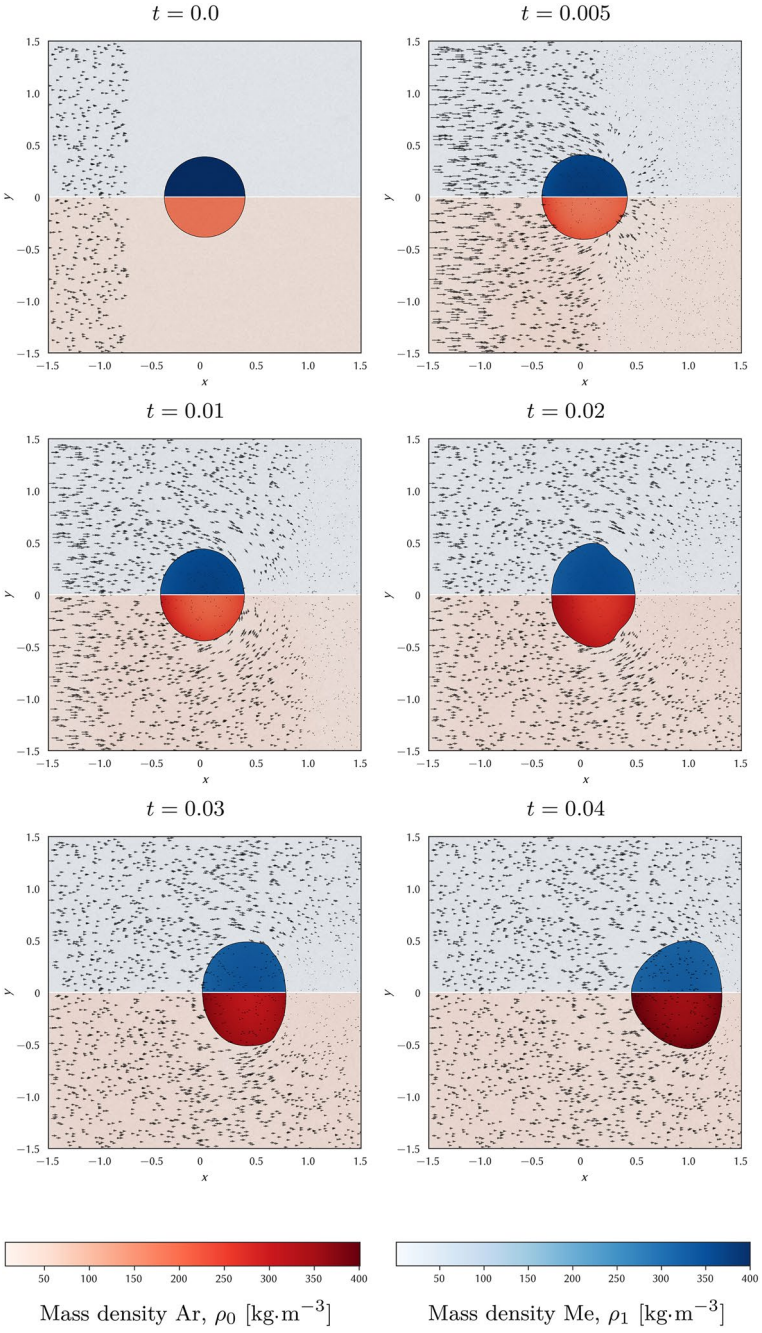

Fig. 7 Two-dimensional multiscale simulations. Each sub-figure depicts the densities ρ_0, ρ_1 and velocities $\boldsymbol{v}_0$, $\boldsymbol{v}_1$ of each component at various time steps. The upper part of each sub-figure shows ρ_1, $\boldsymbol{v}_1$ for methane, and the lower part ρ_0, $\boldsymbol{v}_0$ for argon

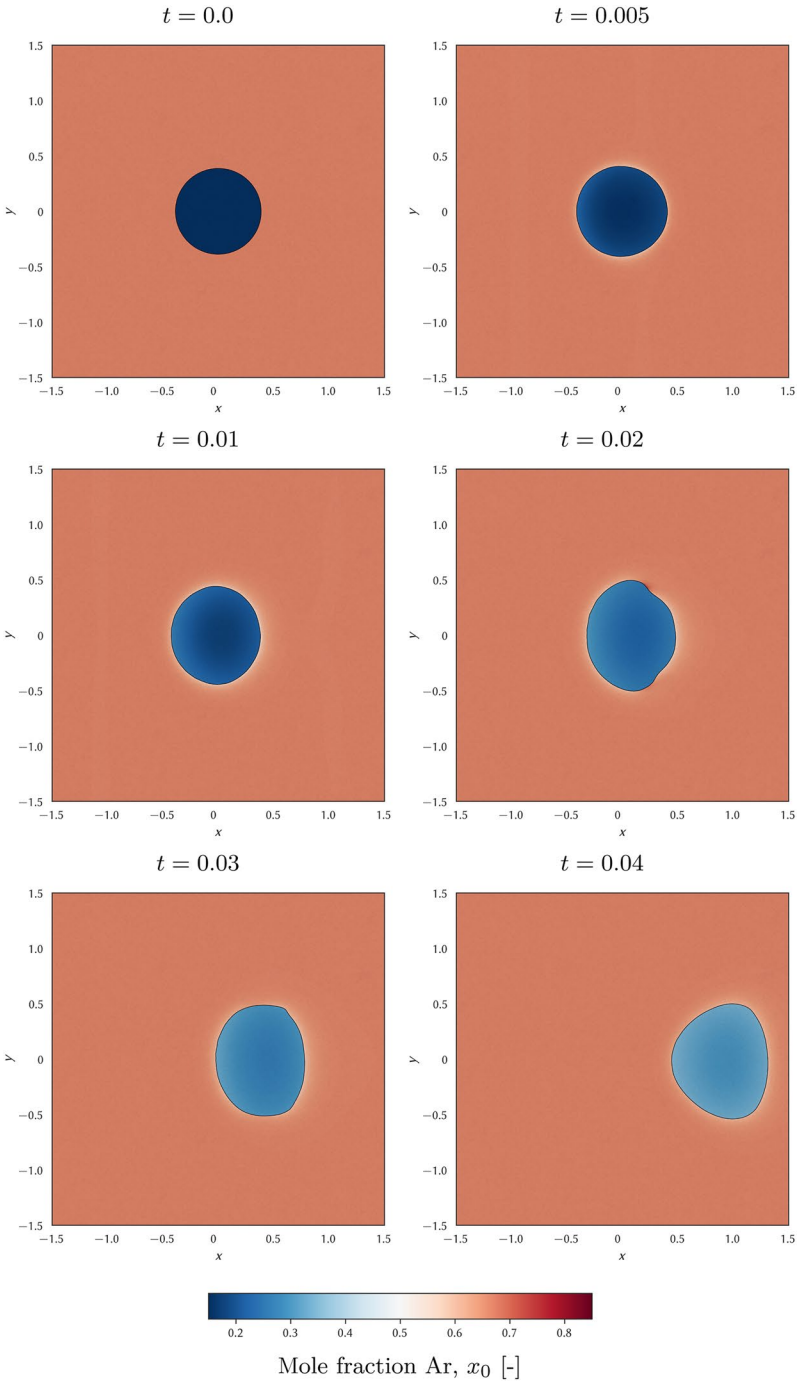

Fig. 8 Two-dimensional multiscale simulation. Each sub-figure depicts the mole fraction x_0 of argon at various time steps

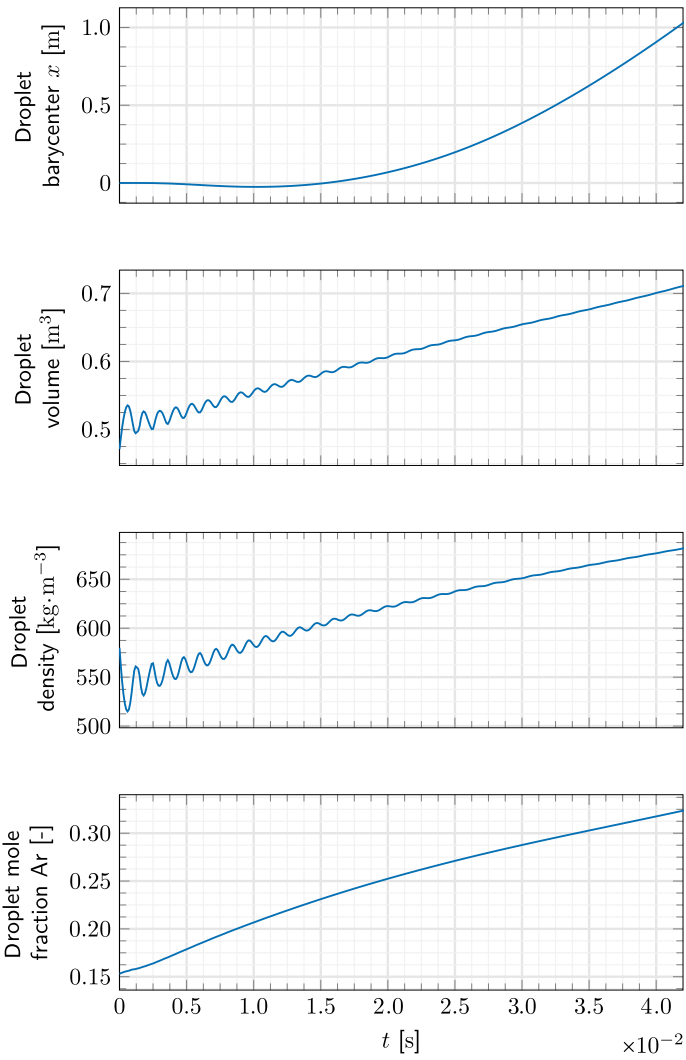

Fig. 9 Time evolution of phase-averaged quantities for the two-component two-phase flow multiscale simulation in two space-dimensions

boundary conditions are applied. On every other side of the domain, we implement reflecting boundary conditions.

For the multiscale simulation results, we refer to Figs. 10 and 11. Both figures show the three-dimensional solution visualized on the plane through $z = 0$. In Fig. 10 the component-wise densities and velocities are depicted. We display also the (projected) mesh to demonstrate that the interface is resolved on the discrete level. In Fig. 11, the phase boundary is shown alongside the barycentric velocity magnitude.

We observe that the droplet is hit by a shock wave, which results in a ripple that runs over the liquid surface. Additionally, the momentum from the vapor causes the droplet to move to

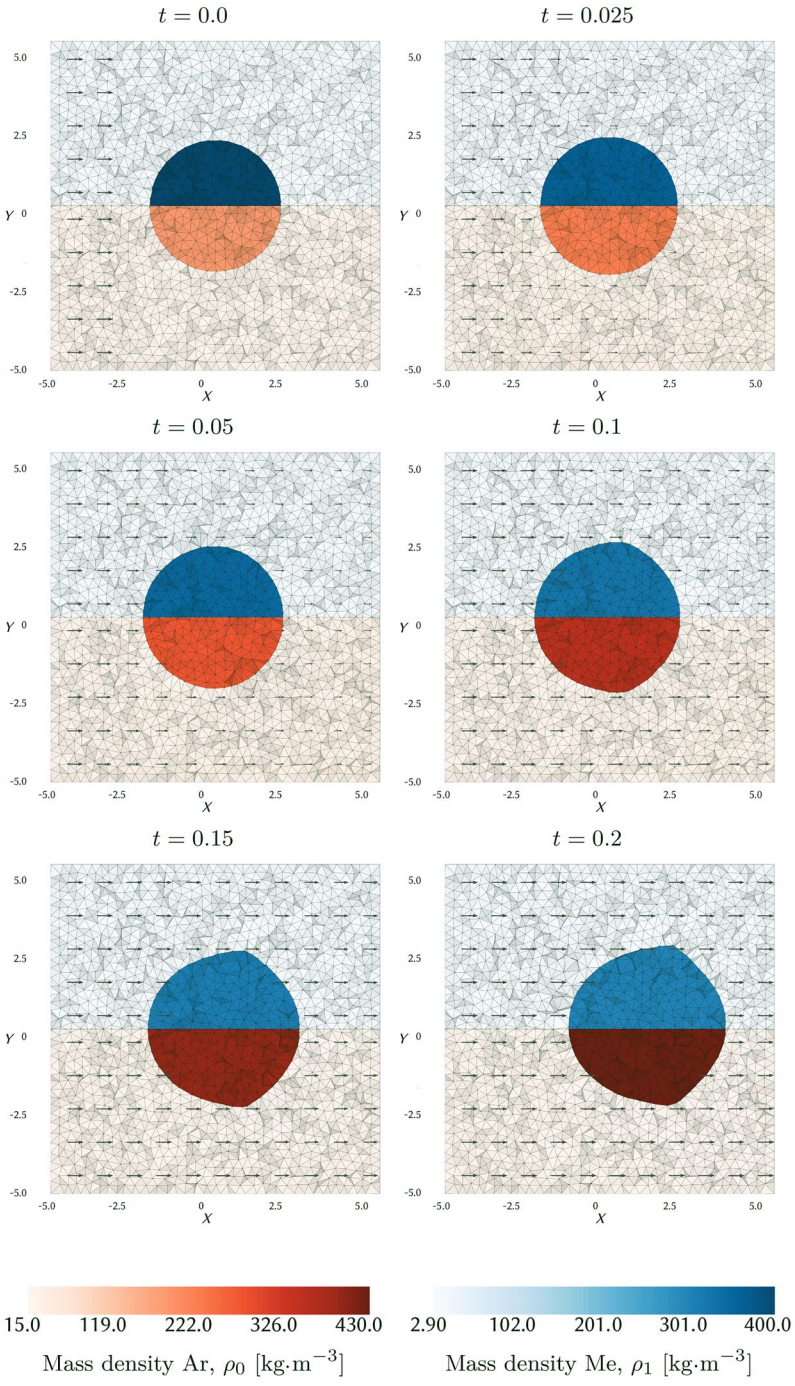

Fig. 10 Three-dimensional multiscale simulation of the two-component two-phase flow model. The upper part of each sub-figure shows ρ_1, $\boldsymbol{v}_1$ for methane, and the lower part ρ_0, $\boldsymbol{v}_0$ for argon, at various time steps

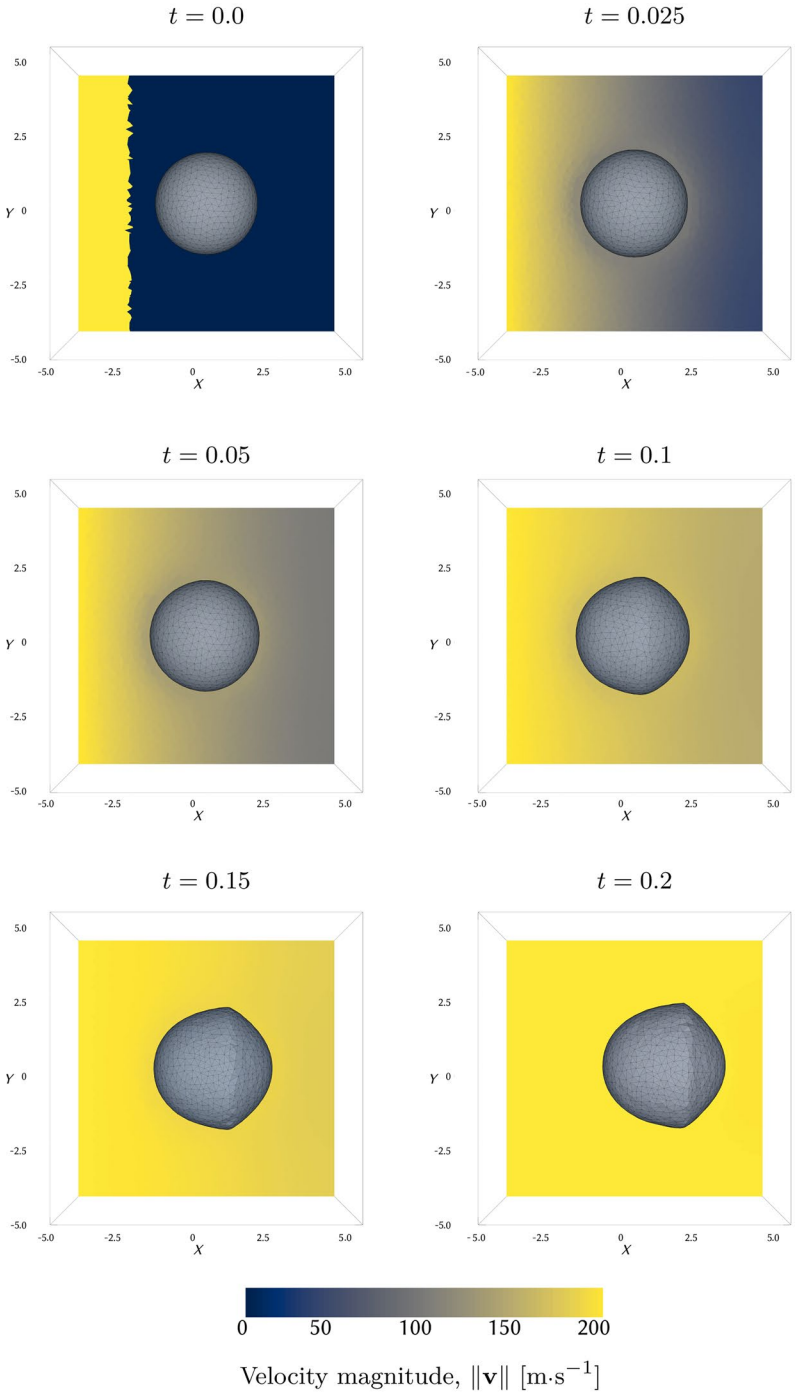

Fig. 11 Three-dimensional multiscale simulation. Each sub-figure depicts the barycentric velocity magnitude $\|\boldsymbol{v}\|_2$ and the phase boundary at various time steps

the right side. Considering the two fluid components, we see that the methane concentration inside the liquid droplet decreases slowly, whereas the amount of liquid argon slowly increases.

Next, we provide an overview over the simulation run times. A single time step of the simulation takes on average 2.7 s. This splits approximately into 1.0 s for the moving mesh operations (see Sect. 4.3), and 1.7 s for the finite volume part of the simulation. The mesh consists of 2.7×10^5 cells, with 2.0×10^3 interface facets averaged over all 100 000 time steps. In total, the whole simulation takes 75 h on a single desktop computer.

Note that in this simulation, the need for a surrogate solver becomes apparent. If we had not employed a surrogate, we would need to perform an MD simulation (taking 14 min to 17 min) for every interface facet at every single time step. For a single time step, that would accumulate to approximately 500 h of computational time (albeit parallelizable). Compared to that, the surrogate solver needs 0.102 ms for a single evaluation. The offline phase of the surrogate solver takes approximately 3 200 h plus training. The majority of this workload (generating the training data set) is however embarrassingly parallel. Consequently, even if we include the offline phase, using a surrogate solver pays off after a few time steps.

6 Conclusions

The main result of our study is a proof-of-concept validation for an HMM that has been designed to model the dynamics of the compressible two-component flow with the liquid-vapor phase transition. The behavior of the numerical solutions on the continuum scale matches the MD simulations, with even quantitatively correct results in the liquid phase domains. Improving the results in the vapor phase seems to require more computational effort in the sampling procedure and the thermalization process. It is noteworthy that we do not consider a generic fluid mixture, but use a real EOS for argon-methane mixtures. Another significant outcome is that the HMM is able to simulate two- and three-dimensional droplets in case of complex interactions. At the same time, both small- and large-scale deformations of the (sharp) phase boundary can be rendered by the moving-mesh finite-volume method during the numerical simulations.

Up to our knowledge, this is the first time that compressible mixtures of real fluids with resolved sharp interfaces are simulated using an MD-based and neural network-accelerated multiscale model. In particular, we are able to simulate phase transitions of fluid mixtures-without the need to prescribe some (ad hoc) closure relations at the interface.

We think that the HMM in combination with an ML surrogate modeling paves the way to handle more complex flow scenarios. Surface tension has not been taken into account, despite its importance for two-phase evolution. Here, it is possible to account for curvature effects on the continuum scale by adding a geometric term to the momentum equation [36]. However, the surface tension coefficient might then require additional MD simulations like for the EOS. On the expense of more MD simulations, we expect that the extension to mixtures with more than two components is straightforward. A conceptual advantage of the underlying moving-mesh finite-volume method is that tailored numerical schemes can be applied in the liquid and the vapor phases. This would allow for benefitting from, e.g., implicit time-stepping in the low-Mach liquid domain and, more importantly for mixtures, to account for large Maxwell-Stefan coefficients.

Appendix A Parameter Tables

See Tables A1, A2, A3, and A4.

Table A1 Parameter table for the atomistic interface solver

Molecular Dynamics	
Cutoff-radius r_{cutoff}	2.5
Total number of particles N	32 768
Number of time steps N_{end}	5×10^2
Time step $\Delta\tau$	5×10^{-4}
Time sampling ratio $\alpha_{\text{t-smpl}}$	0.2

Table A2 Parameters for Sect. 5.1

Simulations for $d = 1$	
End time t_{end}	0.003
Time step Δt	10^7
Domain $\mathbf{\Omega}$	$[-5, 5]$
Base cell width Δx	2×10^{-3}
Δx_{min}	$1/2\Delta x$
Δx_{max}	$3/2\Delta x$
Maxwell-Stefan diffusion coefficient $Đ_{01}$	1.0

Table A3 Parameters for Sect. 5.2

Simulations for $d = 2$	
Time step Δt	5×10^{-7}
Domain $\mathbf{\Omega}$	$[-1.5, 1.5]^2$
Base edge length Δx	1.25×10^{-2}
Δx_{min}	$1/2\Delta x$
Δx_{max}	$3/2\Delta x$

Table A4 Parameters for Sect. 5.3

Simulations for $d = 3$	
Time step Δt	2×10^{-6}
Domain $\mathbf{\Omega}$	$[-5, 5]^3$
Base edge length Δx	0.25
Δx_{min}	$1/2\Delta x$
Δx_{max}	$3/2\Delta x$
Maxwell-Stefan diffusion coefficient $Đ_{01}$	1.0

Funding Open Access funding enabled and organized by Projekt DEAL. When preparing this manuscript, the authors have kept the COPE guidelines on how to deal with potential acts of misconduct. The research leading to these results received funding from Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through the project SFB–TRR 75 with the project number 84292822, and the DFG under Germany's Excellence Strategy-EXC 2075 with the project number 390740016.

Data Availability The data set for the machine learned surrogate model is digitally archieved in [29].

Compliance with Ethical Standards

Conflict of Interest On behalf of all authors, the corresponding author states that there is no conflict of interest. The authors have no relevant financial or non-financial interests to disclose.

References

1. Alkämper, M., Magiera, J.: Interface preserving moving mesh (Code). DaRUS (2021). https://doi.org/10.18419/darus-1671
2. Allen, M.P., Tildesley, D.J.: Computer Simulation of Liquids, 2nd edn. Oxford University Press Inc, Oxford (2017)
3. Anderson, D.M., McFadden, G.B., Wheeler, A.A.: Diffuse-interface methods in fluid mechanics. Annu. Rev. Fluid Mech. **30**(1), 139–165 (1998)
4. Andrianov, N., Warnecke, G.: The Riemann problem for the Baer-Nunziatio model of two-phase flows. J. Comput. Phys. **195**, 434–464 (2004)
5. Berthelot, D.: Sur le mélange des gaz. Comptes rendus hebdomadaires des séances de l'Académie des Sciences **126**, 1703–1706 (1898)
6. Bothe, D., Dreyer, W.: Continuum thermodynamics of chemically reacting fluid mixtures. Acta Mech. **226**(6), 1757–1805 (2023)
7. Chalons, C., Rohde, C., Wiebe, M.: A finite volume method for undercompressive shock waves in two space dimensions. Math. Model. Numer. Anal. **51**, 1987–2015 (2017)
8. Colombo, R.M., Priuli, F.S.: Characterization of Riemann solvers for the two phase p-system. Commun. Partial Differ. Equ. **28**(7/8), 1371–1389 (2003)
9. Dreyer, W., Giesselmann, J., Kraus, C.: A compressible mixture model with phase transition. Physica D **273/274**, 1–13 (2014)
10. Engquist, B., Li, X., Ren, W., Vanden-Eijnden, E.: Heterogeneous multiscale methods: a review. Commun. Comput. Phys. **2**(3), 367–450 (2007)
11. Faccanoni, G., Kokh, S., Allaire, G.: Modelling and simulation of liquid-vapor phase transition in compressible flows based on thermodynamical equilibrium. ESAIM Math. Model. Numer. Anal. **46**(5), 1029–1054 (2012)
12. Fechter, S., Munz, C.-D., Rohde, C., Zeiler, C.: A sharp interface method for compressible liquid-vapor flow with phase transition and surface tension. J. Comput. Phys. **336**, 347–374 (2017)
13. Frezzotti, A., Barbante, P.: Simulation of shock induced vapor condensation flows in the Lennard-Jones fluid by microscopic and continuum models. Phys. Fluids **32**(12), 122106 (2020)
14. Ghazi, H., James, F., Mathis, H.: A nonisothermal thermodynamical model of liquid-vapor interaction with metastability. Discrete Contin. Dyn. Syst. Ser. B **26**(5), 2371–2409 (2021)
15. Gross, J., Sadowski, G.: Perturbed-Chain SAFT: an equation of state based on a perturbation theory for chain molecules. Ind. Eng. Chem. Res. **40**(4), 1244–1260 (2001)
16. Hairer, E., Lubich, C., Wanner, G.: Geometric numerical integration illustrated by the Störmer-Verlet method. Acta Numer. **12**, 399–450 (2003)
17. Han, E., Hantke, M., Warnecke, G.: Criteria for nonuniqueness of Riemann solutions to compressible duct flows. Z. fur Angew. Math. Phys. Mech. **93**(6/7), 465–475 (2013)
18. Hantke, M., Matern, C., Warnecke, G., Yaghi, H.: A new method to discretize a model for isothermal flow with a multi-component equation of state. J. Comput. Appl. **422**, 114876 (2023)
19. Hantke, M., Müller, S.: Analysis and simulation of a new multi-component two-phase flow model with phase transitions and chemical reactions. Q. Appl. Math. **76**(2), 253–287 (2018)
20. Hantke, M., Müller, S., Richter, P.: Closure conditions for non-equilibrium multi-component models. Contin. Mech. Thermodyn. **28**, 1157–1189 (2016)

21. Hantke, M., Thein, F.: On the impossibility of first-order phase transitions in systems modeled by the full Euler equations. Entropy **21**(11), 1039 (2019)
22. Hitz, T., Jöns, S., Heinen, M., Vrabec, J., Munz, C.-D.: Comparison of macro- and microscopic solutions of the Riemann problem II. Two-phase shock tube. J. Comput. Phys. **429**, 110027 (2020)
23. Janzen, T.: On diffusion coefficients of multicomponent liquid mixtures predicted by equilibrium molecular dynamics simulation. Doctoral thesis, Technische Universität Berlin, Berlin (2019)
24. Keim, J., Munz, C.-D., Rohde, C.: A relaxation model for the non-isothermal Navier-Stokes-Korteweg equations in confined domains. J. Comput. Phys. **474**, 111830 (2023)
25. Krishna, R., Wesselingh, J.A.: The Maxwell-Stefan approach to mass transfer. Chem. Eng. Sci. **52**(6), 861–911 (1997)
26. Lorentz, H.A.: Ueber die Anwendung des Satzes vom Virial in der kinetischen Theorie der Gase. Annalen der Physik **248**(1), 127–136 (1881)
27. Ma, P.C., Lv, Y., Ihme, M.: An entropy-stable hybrid scheme for simulations of transcritical real-fluid flows. J. Comput. Phys. **340**, 330–357 (2017)
28. Magiera, J.: A molecular-continuum multiscale solver for liquid-vapor flow: modeling and numerical simulation. PhD thesis, University of Stuttgart, Stuttgart (2021)
29. Magiera, J.: Data sets for a molecular-continuum multiscale solver for liquid-vapor flow: modeling and numerical simulation. DaRUS (2021). https://doi.org/10.18419/darus-1258
30. Magiera, J., Ray, D., Hesthaven, J.S., Rohde, C.: Constraint-aware neural networks for Riemann problems. J. Comput. Phys. **409**, 109345 (2020)
31. Magiera, J., Rohde, C.: A particle-based multiscale solver for compressible liquid-vapor flow. In: Klingenberg, C., Westdickenberg, M. (eds.) Theory, Numerics and Applications of Hyperbolic Problems II, pp. 291–304. Springer, Cham (2018)
32. Magiera, J., Rohde, C.: A molecular-continuum multiscale model for inviscid liquid-vapor flow with sharp interfaces. J. Comput. Phys. **469**, 111551 (2022)
33. Magiera, J., Rohde, C.: Analysis and numerics of sharp and diffuse interface models for droplet dynamics. In: Schulte, K., Tropea, C., Weigand, B. (eds.) Droplet Dynamics Under Extreme Ambient Conditions. Fluid Mechanics and Its Applications, pp. 67–86. Springer, Cham (2022)
34. Mitchell, D.P.: Spectrally optimal sampling for distribution ray tracing. In: Proceedings of the 18th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH 91, pp. 157–164. ACM, New York (1991)
35. Ponte, M., Streett, W.B., Miller, R.C., Staveley, L.A.K.: An experimental study of the equation of state of liquid (argon + methane), and the effect of pressure on their excess thermodynamic functions. J. Chem. Thermodyn. **13**(8), 767–781 (1981)
36. Rohde, C., Zeiler, C.: On Riemann solvers and kinetic relations for isothermal two-phase flows with surface tension. Z. fur Angew. Math. Phys. **69**(3), 76 (2018)
37. Saurel, R., Abgrall, R.: A multiphase Godunov method for compressible multifluid and multiphase flows. J. Comput. Phys. **150**(2), 425–467 (2023)
38. Saurel, R., Petitpas, F., Abgrall, R.: Modelling phase transition in metastable liquids: application to cavitating and flashing flows. J. Fluid Mech. **607**, 313–350 (2008)
39. Shen, Y., Ren, Y., Ding, H.: A 3D conservative sharp interface method for simulation of compressible two-phase flows. J. Comput. Phys. **403**, 109107–10919 (2020)
40. Truskinovsky, L.: Kinks versus shocks. In: Shock Induced Transitions and Phase Structures in General Media. IMA Volumes in Mathematics and Its Applications, vol. 52, pp. 185–229. Springer, New York (1993)
41. Vrabec, J., Fischer, J.: Vapour liquid equilibria of mixtures from the NpT plus test particle method. Mol. Phys. **85**(4), 781–792 (1995)
42. Vrabec, J., Lotfi, A., Fischer, J.: Vapour liquid equilibria of Lennard-Jones model mixtures from the NpT plus test particle method. Fluid Phase Equilib. **112**(2), 173–197 (1995)
43. Zein, A., Hantke, M., Warnecke, G.: Modeling phase transition for compressible two-phase flows applied to metastable liquids. J. Comput. Phys. **229**(8), 2964–2998 (2010)