Coupling boundary conditions in continuum-particle approach for open systems: theoretical analysis and computational implementation

Dissertation
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

am Fachbereich Mathematik und Informatik
der Freien Universität Berlin

vorgelegt von
Abbas Gholami Poshtehani

Berlin, 2022

Erstgutachter:
Prof. Dr. Luigi Delle Site
Arnimallee 9,
Room 010,
14195 Berlin

Zweitgutachter:
Prof. Dr. Carsten Hartmann
Konard-Wachsmann-Allee 1,
03046 Cottbus,
Hauptgebäude R. 3.52

Tag der Disputation: 26.10.2022

Declaration of authorship

Name: Gholami Poshtehani
First name: Abbas

I declare to the Freie Universität Berlin that I have completed the submitted dissertation independently and without the use of sources and aids other than those indicated. The present thesis is free of plagiarism. I have marked as such all statements that are taken literally or in content from other writings. This dissertation has not been submitted in the same or similar form in any previous doctoral procedure.

I agree to have my thesis examined by a plagiarism examination software.

Date:

Signature:

Dedication

I would like to dedicate this thesis to my parents and family who without their support I would never be in the situation that I am in today. Their efforts and encouragement allowed me to believe in things that seem unachievable to me in the first place. Sadly, they couldn't be with me today to celebrate together, but they have been always in my heart.

I would like to dedicate this thesis to my beloved wife, Maryam, who changed her life direction and stayed beside me. She has always believed in me and supported me whenever I was disappointed. It wasn't possible for me to do this job without her support, help, and ideas.

Acknowledgements

This research has been funded by Deutsche Forschungsgemeinschaft (DFG) through grant CRC 1114 “Scaling Cascade in Complex Systems”, Project Number 235221301, Project C01 “Adaptive coupling of scales in Molecular Dynamics and beyond to Fluid Dynamics”.

I am extremely grateful to my supervisor Prof. Luigi Delle Site who was thoroughly involved in this thesis and was always supportive. Without his guidance, ideas, and creative discussions, it wasn't possible to do this work and plan all steps included in this thesis. I believe that his supervision and effort in motivating me have shaped me into a better researcher and I would be always thankful to him. I would like to acknowledge the invaluable support, discussions, and ideas from Prof. Rupert Klein who was always beside us and made the work more valuable with his brilliant points and ideas throughout all stages. Furthermore, I would like to thank Dr. Felix Höfling for the helpful and constructive discussions during my PhD. I also thank my colleague Roya Ebrahimi Viand for being helpful and taking the time to discuss issues and providing helpful ideas during several steps of this thesis and my former colleague John Whittaker for helping in setting some simulations at the beginning of my PhD.

Abstract

Adaptive Resolution Simulation (AdResS) is a multi-resolution method with open system characteristics for modelling atomistic-level systems. In AdResS, a high-resolution open system is in contact with a reservoir of particles and energy, and the system is recreating the thermodynamics and physics of the full atomistic system of reference. In this thesis, the fundamental characteristics of the AdResS method are studied to provide a better understanding of the statistical mechanics undergoing within open system.

Among the most relevant results, it is worth underlining the equivalence of the grand potential, determined theoretically, with the pressure, calculated numerically for the same volume of the atomistically resolved region. Moreover, such analysis led to a straightforward calculation of the chemical potential of the liquid under investigation for a wide range of thermodynamic conditions. It has been shown that the pressure difference resulting from the abrupt change of resolutions is compensated by the energy provided by the external force (thermodynamic force) in AdResS. Moreover, the chemical potential of AdResS is related to the chemical potential of the full-atomistic simulation of reference by calculating different contributions corresponding to the abrupt change of resolutions.

Next, a fluctuating hydrodynamics (FHD) solver is designed to capture the small-scale fluctuations in the continuum simulations by adding a stochastic flux term to the Navier-Stokes equation of the compressible flow. Then, this continuum solver is coupled to the previously developed AdResS simulator through a small interface region by employing a novel coupling algorithm according to the non-equilibrium AdResS simulation. To this aim, a set of pre-calculated thermodynamic forces is prepared and the information on the continuum side transfers to the particle subdomain by interpolating proper thermodynamic force. The AdResS-FHD coupling system is developed and tested for various cases with different conditions and showed satisfactory agreement with the results of the reference continuum and fully atomistic simulations.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Molecular Dynamics (MD) is a numerical simulation method for modelling the physical motion of particles in various substances. The atoms and molecules are allowed to interact with each other through a well-defined interaction potential which determines the evolution of the system over time [10]. The trajectories of atoms and molecules determine the thermodynamic and physical properties of the system at the macroscale according to the principles of statistical mechanics [10, 87]. The atomic resolution requires significantly more computational resources than macroscale simulation at lower resolutions. Thus, computational costs restrict one to small size and time scales in molecular dynamics at atomistic resolution. MD simulations are popular in materials science, chemistry, and physics with applications in problems with important atomic-level details. MD has been frequently used in e.g., ab-initio prediction of protein and amino acid conformation [72,109], refinement of protein and macromolecule's structure based on experimental constraints from X-ray and NMR [2,90], thin-film growth and ion-subplantation in physics [53, 82], tribological studies in material science [168], pharmacophore development and drug design [29], and the chemistry of catalysis [178], to mention just a few.

MD simulations are computationally expensive as they require recording the positions and velocities of every single particle and calculating their inter-particle forces over time. To reduce the computational costs of the simulations, it is possible to consider some parts of the simulation domain with atomic-level details and MD resolution and the rest which acts as a reservoir environment with lower resolution coarse-grained or even continuum descriptions. The Adaptive Resolution Simulation (AdResS) method was first developed in 2005 to couple different regions with different particle-based resolutions [156] while there exist some prior works to link regions with hybrid schemes of coupling MD to a finite element [24, 162, 175] or quantum-mechanical approaches [32]. In AdResS, particles can change resolution from a full-atom representation to the coarse-grained region on-the-fly. There are two kinds of AdResS approaches and one of them is based on the Hamiltonian of the system (H-AdResS) in which all molecules need both atomistic description and a well-defined mapping point. Thus, in H-AdResS, the Hamiltonian of the system is adjusted while changing resolution from the all-atom region to the coarse-grained resolution [88, 155]. In the other approach, where this work is based, instead of adjusting the Hamiltonian of the system, a weight function is defined for the hybrid region between the all-atom region and the coarse-grained reservoir. The weight function adjusts the inter-particle interactions between particles in different regions according to the all-atom and coarse-grained interactions [114, 132, 156, 160]. However, in the latest version of AdResS, on which this work is based, the coarse-grained region is replaced with a reservoir of non-interacting particles with an abrupt change of resolutions and replacing the weight function with a one-body external force [49]. There are several applications studied by the AdResS method in recent years including Path Integral Molecular Dynamics for liquid water [3, 5], macromolecular liquids [150], ionic liquids [46, 170], hydration of membranes [197], multiphase flow [127], etc. in which the method showed a well-enough resemblance to the reference set-up with coupling different regions

at different resolutions.

In recent decades, various attempts to couple a Molecular Dynamics simulator to continuum hydrodynamics have been done and are reported in the literature [84,147]. In many approaches, a continuum-particle overlapped region exists that acts as a separation boundary between the continuum and particle subdomains with different resolutions. For the overlap region, state variables and flux coupling scheme are introduced to take care of the conservation of bulk mass, momentum and energy [37,51,63,141,147,193]. An overlap region is needed between the MD and continuum parts of the domain to avoid the sharp oscillations in density and pressure between different resolutions similar to the AdResS scheme where the atomistic region is connected to the reservoir region through a small hybrid region [157].

1.1 Outline of the thesis

In this thesis, after the preparation of the AdResS set-up with the calibration of the parameters, the influence of different parameters on the convergence of the simulations and quality of results are systematically studied to allow for a more logical choice of the parameters in later studies. As the first step, the chemical potential of the AdResS set-up and its reference simulation are mathematically related to each other by deriving the proper relation of the grand potential for open systems (a grand canonical ensemble). This explains how the necessary calibrations contribute to the chemical potential of the system during the abrupt change of the resolution in AdResS [73]. Later, as a validation of the assumption in the previous step through the grand potential's balance of the AdResS set-up, the pressure of the system is calculated rigorously for both AdResS and its reference set-ups. It shows how the additional tools of the AdResS scheme (external force) will compensate for the resulting pressure difference due to the change of resolutions [74]. Finally, to further develop the idea of multiscale modelling, a fluctuating hydrodynamics solver is developed to capture the fluctuations at the continuum level. This solver is coupled to the latest version of AdResS with an innovative idea based on the non-equilibrium properties of an open system.

To this aim and according to the discussion above, the current thesis is designed as the following: the theoretical backgrounds of Molecular Dynamics and the AdResS scheme are introduced in the second chapter. In the third chapter, the chemical potential of the system in the AdResS scheme is mathematically related to the chemical potential of the reference set-up. In the fourth chapter, the pressure of the fluid is studied in the AdResS and its reference set-up. In the fifth chapter, the fluctuating hydrodynamic solver is developed and coupled to the AdResS simulation method by introducing a novel coupling algorithm. Finally, the work is summarized in the last chapter and the future outlook for the research is described. In the appendix, the simulation code details for the MD simulation on different packages used in this work and also for the continuum solver are presented.

Chapter 2

Theoretical background

2.1 Molecular Dynamics (MD)

In this chapter, the necessary concepts and theories in the field of Molecular Dynamics and atomistic level related to this work will be introduced. In the MD simulation method, the trajectory of the particles is calculated by numerically solving Newton's equation of motion in which the inter-particle forces are calculated from interatomic potentials or molecular mechanics force field.

The time evolution of molecular dynamics is used for the calculation of the macroscopic thermodynamic properties of the system. A key hypothesis is the ergodic property i.e. the time average agrees with the microcanonical ensemble average (explained later in this chapter) [164]. In molecular dynamics simulations, the information of the N particles at the microscopic level, i.e. the positions and velocities, are translated into macroscopic properties such as energy, temperature, pressure, etc. If we represent the set of coordinates and momenta of N particles with $\mathbf{r}$ and $\mathbf{p}$ respectively, it reads: $\mathbf{r} \equiv \mathbf{r}_1, ..., \mathbf{r}_N$ and $\mathbf{p} \equiv \mathbf{p}_1, ..., \mathbf{p}_N$ where $\mathbf{r}_i$ and $\mathbf{p}_i$ are the coordinates and momenta of i^{th} particle, respectively. The values of these 6N variables define a point in the 6N-dimensional phase space. The instantaneous value of an observable S would be a function of the microstate, i.e. $s(\mathbf{r}, \mathbf{p})$ and its value can be calculated by averaging overtime during a sufficiently long MD simulation.

$$S = \langle s(\mathbf{r}, \mathbf{p}) \rangle_{time} = \lim_{t \to \infty} \frac{1}{t} \int_0^t s(\mathbf{r}, \mathbf{p}, t') dt' \tag{2.1}$$

However, in molecular dynamics, the time is divided into several (n) discrete time steps (τ) which means that the integral in Eq.2.1 will be replaced by a sum,

$$S = \langle s(\mathbf{r}, \mathbf{p}) \rangle = \frac{1}{n} \sum_{i=1}^{n} s(\mathbf{r}_i, \mathbf{p}_i, t_i) \tag{2.2}$$

In the most common form of classical molecular dynamics, the initial positions and velocities of particles will be set at time $t = 0$. Next, the inter-particle forces are calculated based on the predefined inter-particle interaction potentials and accordingly, the particles' acceleration will determine the new positions and velocities by using an integration algorithm. Then, the new observables will be calculated and analyzed after some time (depending on the system under study) to see whether the system has been equilibrated or not; afterwards, the simulation will proceed to the next time step and repeat the previous steps for reaching the new state. This general form of the MD algorithm is illustrated in the flowchart of Fig.2.1.

The study of a system at a macroscopic level using molecular dynamics simulation requires statistical mechanics. At the trajectory level, molecular dynamic simulations generate data such

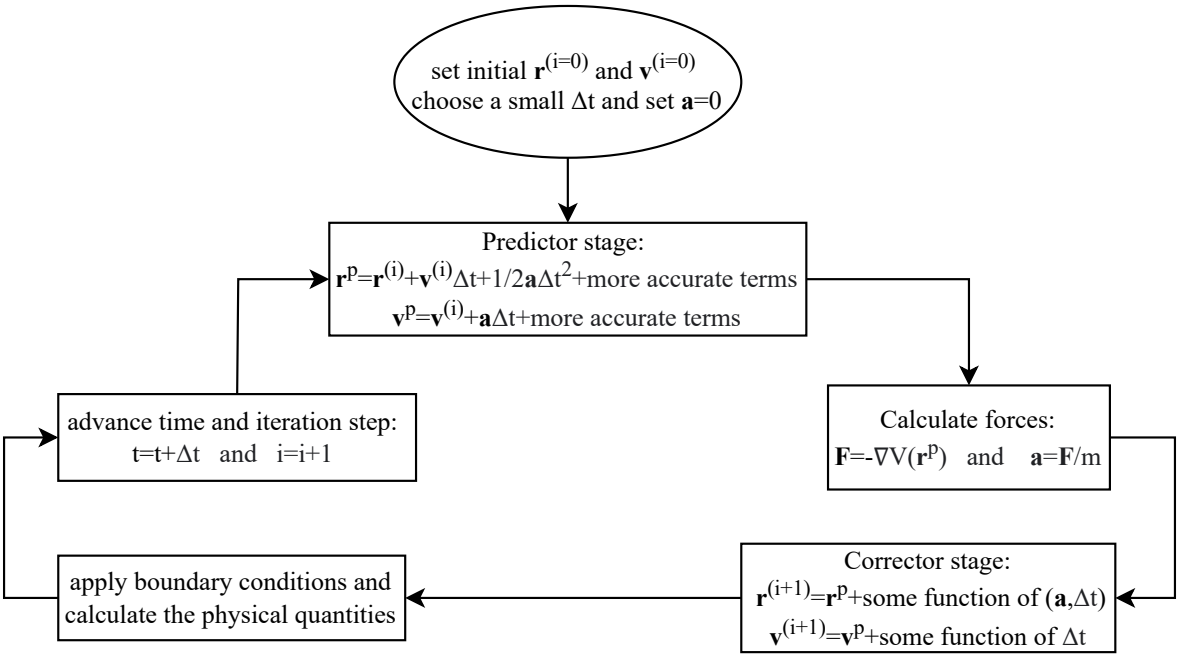


Figure 2.1: A simplified classical molecular dynamic algorithm with a predictor-corrector integrator. The interparticle forces might be calculated by the inter-particle interaction potentials. The type of integrator has a significant effect on the simulation as some of them may have higher-order terms or may use both current and prior step variables.

as atomic position and velocities. Statistical mechanics is required to convert microscopic information into macroscopic attributes such as pressure, energy, thermal properties, etc. In this sense, the thermodynamic state of a system is generally characterized by a limited number of parameters, such as temperature (T), pressure (P), and particle number (N), where the equations of state and other fundamental thermodynamic equations can be used to determine various thermodynamic characteristics.

2.2 Statistical Mechanics

The definition of certain equilibrium and time-dependent distribution functions of fundamental importance in the theory of liquids, as well as a summary of the principles of classical statistical mechanics and a discussion of the link between statistical mechanics and thermodynamics, are provided in this section [87]. In addition, a set of notations that will be used in this thesis are established here.

2.2.1 Statistical ensemble

A collection of points in phase space that meet the criteria of a specific thermodynamic state is referred to an ensemble. An ensemble, in other terms, is a collection of all feasible systems with distinct microscopic states but with the same macroscopic or thermodynamic state. A molecular dynamic simulation generates a series of points in phase space as a function of time; these points all belong to the same ensemble and correspond to the system's various conformations and momenta [77, 110, 120, 165].

In statistical mechanics, there are three main forms of ensembles. The *microcanonical ensemble*

is formed when the system under investigation is isolated, that is, it does not interact with any other systems. The system's energy is constant in this situation. The ensemble is considered a *canonical ensemble* if the system under investigation is in thermal equilibrium with a heat reservoir at temperature T. The system's energy is not constant in this example, but the temperature of the heat bath is (the system's mean temperature is equal to the temperature of the heat bath). The ensemble is known as a *grand canonical ensemble* when the system under discussion is in touch with both heat and particle reservoirs. The system's energy and particle number are not constant in this scenario, but the temperature and chemical potential are.

In all the different ensembles mentioned above, it is crucial to understand the time evolution of the ensemble's phase points by statistical mechanics. In this sense, the Liouville theorem explains the time evolution with regard to the particles' trajectory.

2.2.2 Time evolution and Liouville theorem

Consider a macroscopic system of N identical, point particles of mass m that are contained in a volume V and create a 6N-dimensional phase space. Let $\mathcal{H}$ be the system's Hamiltonian, which can be expressed as

$$\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N) = K_N(\mathbf{p}^N) + V_N(\mathbf{r}^N) + \Phi_N(\mathbf{r}^N) \tag{2.3}$$

with $K_N = \sum_{i=0}^{N} |\mathbf{p}_i|^2/2m_i$ being the kinetic energy, V_N the inter-particle potential energy, and Φ_N the external potential field which in its absence, the system would be spatially uniform and isotropic. Hamilton's equations define the motion of the phase point along its phase trajectory:

$$\dot{\mathbf{r}}_i = \frac{\partial \mathcal{H}}{\partial \mathbf{p}_i}, \qquad \dot{\mathbf{p}}_i = -\frac{\partial \mathcal{H}}{\partial \mathbf{r}_i} \tag{2.4}$$

The distribution of phase points in the ensemble is given by a phase space probability density $f^{[N]}$. The probability that the physical system is in a microscopic condition represented by a phase point residing in the infinitesimal phase space element $d\mathbf{r}^N d\mathbf{p}^N$ at time t is given by the quantity $f^{[N]} d\mathbf{r}^N d\mathbf{p}^N$. According to this definition, probability density $f^{[N]}$ across all phase space integrates to one for every t.

The Liouville equation applies to the local density in the vicinity of a point as the point follows a Hamiltonian flow in the phase space. The Liouville equation can be expressed in two ways.

$$\frac{\partial f^{[N]}}{\partial t} + \sum_{i=1}^{N} \left(\frac{\partial f^{[N]}}{\partial \mathbf{r}_i} \cdot \dot{\mathbf{r}}_i + \frac{\partial f^{[N]}}{\partial \mathbf{p}_i} \cdot \dot{\mathbf{p}}_i \right) = 0 \tag{2.5}$$

or, to put it in another way,

$$\frac{\partial f^{[N]}}{\partial t} = \{\mathcal{H}, f^{[N]}\} \tag{2.6}$$

where the Poisson bracket is denoted by the $\{A, B\}$:

$$\{A, B\} \equiv \sum_{i=1}^{N} \left(\frac{\partial A}{\partial \mathbf{r}_i} \cdot \frac{\partial B}{\partial \mathbf{p}_i} - \frac{\partial A}{\partial \mathbf{p}_i} \cdot \frac{\partial B}{\partial \mathbf{r}_i} \right)$$

Alternatively, using the Liouville operator $\mathcal{L}$, which is defined as $\mathcal{L} \equiv i\{\mathcal{H}, \}$, the Liouville equation takes the following format:

$$\frac{\partial f^{[N]}}{\partial t} = -i\mathcal{L} f^{[N]} \tag{2.7}$$

with the initial specified condition of $f^{[N]}(0)$ and the following formal solution:

$$f^{[N]}(t) = exp(-i\mathcal{L}t)f^{[N]}(0) \tag{2.8}$$

The Liouville equation can be written as $df^{[N]}/dt = 0$. This is known as the Liouville theorem, and it states that the probability density is time-independent. Consider the phase points contained within a phase space element $d\mathbf{r}^N(0)d\mathbf{p}^N(0)$ at time $t = 0$. The element's form will vary over time, but no phase points will enter or exit. As a result of the Liouville theorem, the element's volume must remain constant. Similarly, it can be proved that the temporal evolution of any phase space function $s(\mathbf{r}, \mathbf{p})$ may be described in terms of the Liouville operator:

$$s(\mathbf{r}, \mathbf{p}, t) = exp(i\mathcal{L}t)s(\mathbf{r}, \mathbf{p}) \tag{2.9}$$

The thermodynamic and physical properties of the system can be expressed in terms of the particle trajectories and probability density of the phase space by taking time and ensemble averages.

2.2.3 Time and ensemble averages

Some thermodynamic features of a system can be expressed as averaged functions of the particles' coordinates and momenta. These averages must be time-independent in thermal equilibrium. When a system is isolated from its surroundings, its total energy remains constant, indicating that the Hamiltonian is constant. Let $S(\mathbf{r}^N, \mathbf{p}^N)$ be a function of the phase space variables, and $\langle S \rangle$ be its average value, where the angle brackets denote an undetermined averaging mechanism. Newton's equations of motion, which are a set of 3N second-order differential equations, may be used to calculate the particles' location and velocity (in the absence of an external field) at any time.

$$\mathbf{F}_i = -\nabla_i V_N(\mathbf{r}^N) = m\ddot{\mathbf{r}}_i \tag{2.10}$$

where $\mathbf{F}_i$ is the total force on i^{th} particle. Then, the time average of S can be written as,

$$\langle S \rangle_t = \lim_{t\to\infty} \frac{1}{t} \int_0^t S(\mathbf{r}^N(t'), \mathbf{p}^N(t'))dt' \tag{2.11}$$

The average over an appropriately built ensemble is an alternative to the time-averaging approach stated in Eq.2.11. The coordinates and momenta of the particles fluctuate amongst ensembles, and the ensemble dynamics are represented by the motion of phase points scattered in phase space according to the probability density $f^{[N]}(\mathbf{r}^N, \mathbf{p}^N; t)$. As a result, the equilibrium ensemble average of the function $S(\mathbf{r}^N, \mathbf{p}^N)$ is,

$$\langle S \rangle_e = \iint S(\mathbf{r}^N, \mathbf{p}^N) f_0^{[N]}(\mathbf{r}^N, \mathbf{p}^N) d\mathbf{r}^N d\mathbf{p}^N \tag{2.12}$$

where $f_0^{[N]}$ is the equilibrium probability density.

For a system at equilibrium, the probability distribution function is time independent which means $\partial f^{[N]}(\mathbf{r}^N, \mathbf{p}^N, t)/\partial t = 0$ (see the previous section). Thus, Eq.2.6 reduces to:

$$\{\mathcal{H}, f^{[N]}\} = 0 \tag{2.13}$$

Thus, the probability distribution function is a function of the system's Hamiltonian $\mathcal{H}$,

$$f^{[N]}(\mathbf{r}^N, \mathbf{p}^N) = \frac{1}{\mathcal{Q}_N} F(\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N)) \tag{2.14}$$

where $\mathcal{Q}_N$ is a normalization factor, known as a partition function, defined as:

$$\mathcal{Q}_N = \int F(\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N)) d\mathbf{r}^N d\mathbf{p}^N \tag{2.15}$$

The partition function characterizes the statistical features of a system in equilibrium and quantifies the number of microstates in the phase space that are accessible inside the ensemble, and is specific to the macroscopic parameters used to define the ensemble.

The probability distribution function and the partition function for different ensembles introduced in Sec.2.2.1 are explained in the following.

Microcanonical ensemble(NVE)

In this ensemble, the system is isolated that means for a constant volume, it doesn't exchange energy and matter with its surroundings thus the energy and particles' number remain constant. The equilibrium probability density for this ensemble is

$$f_0^{[N]}(\mathbf{r}^N, \mathbf{p}^N) = \frac{1}{\mathcal{Q}_N} \delta(\mathcal{H} - E) \tag{2.16}$$

where δ is the Dirac δ-function and $\mathcal{Q}_N$ is

$$\mathcal{Q}_N = \frac{1}{N!\, h^{3N}} \int \delta(\mathcal{H} - E) d\mathbf{r}^N d\mathbf{p}^N \tag{2.17}$$

where h is Planck's constant. The factor $1/h^{3N}$ is included in these formulations to guarantee that both $f_0^{[N]} d\mathbf{r}^N d\mathbf{p}^N$ and $\mathcal{Q}_N$ are dimensionless and form-consistent with the relevant quantum statistical mechanics variables, whereas division by $N!$ ensures that all states are counted appropriately.

The probability distribution function in Eq.2.16 selects out those states that have energy E and discards other states. A microcanonical ensemble is therefore equally distributed across the phase space corresponding to a total energy E. If the system is *ergodic*, time and ensemble averages are identical, which means that once a sufficient amount of time has elapsed, the phase trajectory of the system will travel an equal number of times through each phase space specified by Eq.2.16.

Canonical ensemble(NVT)

This ensemble is a closed system that has a constant temperature by exchanging energy with the reservoir while preserving a constant number of particles for a constant volume. The equilibrium distribution function for the canonical ensemble is

$$f_0^{[N]}(\mathbf{r}^N, \mathbf{p}^N) = \frac{1}{\mathcal{Q}_N} exp(-\beta \mathcal{H}) \tag{2.18}$$

where β is the reciprocal of the system's thermodynamic temperature $\beta = 1/k_B T$ and the partition function is:

$$\mathcal{Q}_N = \frac{1}{N!\, h^{3N}} \int exp(-\beta \mathcal{H}) d\mathbf{r}^N d\mathbf{p}^N \tag{2.19}$$

The equilibrium distribution function in Eq.2.18 is referred to Boltzmann distribution. It is noteworthy to mention that in the canonical ensemble, the Hamiltonian is not constant as the system exchanges energy with its surroundings to preserve the temperature.

Grand canonical ensemble(μVT)

This ensemble represents an open system that exchanges both energy and matter with the reservoir to have a constant temperature and chemical potential for a constant volume. The equilibrium probability distribution function for the grand canonical ensemble is

$$f_0^{[N]}(\mathbf{r}^N, \mathbf{p}^N) = \frac{1}{\mathcal{Q}_N} exp(-\beta(\mathcal{H} - \mu N)) \tag{2.20}$$

with μ being the chemical potential. The corresponding partition function is

$$\mathcal{Q}_N = \sum_{N=0}^{\infty} \frac{1}{N!\, h^{3N}} \int exp(-\beta(\mathcal{H} - \mu N)) d\mathbf{r}^N d\mathbf{p}^N \tag{2.21}$$

2.3 Integrators

During a molecular dynamics simulation as specified in the algorithm of MD in Fig.2.1, the positions and velocities of particles need to be updated at each step according to Newton's equation of motion. This updating step requires the force acting on each particle and it can be calculated based on the pre-defined inter-particle potentials ($\mathbf{F}_i(t) = -\nabla_i U(\mathbf{r}^N)$). In the Hamiltonian of the system, Eq.2.3, in the absence of external force fields and with the inter-particle potential of U,

$$\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N) = \sum_{i=0}^{N} \frac{|\mathbf{p}_i|^2}{2m_i} + U(\mathbf{r}^N) \tag{2.22}$$

employing the relations in Eq.2.4 will result

$$\dot{\mathbf{r}}_i = \frac{\partial \mathcal{H}}{\partial \mathbf{p}_i} = \frac{\mathbf{p}_i}{m_i}, \qquad \dot{\mathbf{p}}_i = -\frac{\partial \mathcal{H}}{\partial \mathbf{r}_i} = -\nabla_i U(\mathbf{r}^N) \tag{2.23}$$

These evolution equations determine the system's state for the next step which may be calculated by using some integrator algorithms. In an integration algorithm [188], a discretization of the equations of motion can be obtained by Taylor expansion:

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t + \frac{\mathbf{f}_i(t)}{m_i}\frac{\Delta t^2}{2!} + \dddot{\mathbf{r}}_i(t)\frac{\Delta t^3}{3!} + \mathcal{O}(\Delta t^4) \tag{2.24}$$

$$\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i(t) + \frac{\mathbf{f}_i(t)}{m_i}\Delta t + \ddot{\mathbf{v}}_i(t)\frac{\Delta t^2}{2!} + \dddot{\mathbf{v}}_i(t)\frac{\Delta t^3}{3!} + \mathcal{O}(\Delta t^4) \tag{2.25}$$

where $\mathbf{r}_i$ and $\mathbf{v}_i$ are i^{th} particle's location and velocity, respectively.

Some of the most common integrating schemes are introduced in what follows.

Euler algorithm

The simplest integration scheme based on the Taylor expansion in Eq.2.24 and Eq.2.25 is the Euler integration algorithm which reads

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t + \frac{\mathbf{f}_i(t)}{m_i}\frac{\Delta t^2}{2!} + \mathcal{O}(\Delta t^3) \tag{2.26}$$

$$\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i(t) + \frac{\mathbf{f}_i(t)}{m_i}\Delta t + \mathcal{O}(\Delta t^2) \tag{2.27}$$

The Euler scheme is neither time-reversible nor volume-preserving in the phase space and hence is rather unfavourable. Nevertheless, the Euler scheme can be used to integrate other equations of motion, e.g. the Boltzmann equation [10, 92].

Verlet algorithm

Similar to Eq.2.24 and by expansion of the $\mathbf{r}_i(t-\Delta t)$ according to Taylor expansion

$$\mathbf{r}_i(t-\Delta t) = \mathbf{r}_i(t) - \mathbf{v}_i(t)\Delta t + \frac{\mathbf{f}_i(t)}{m_i}\frac{\Delta t^2}{2!} - \dddot{\mathbf{r}}_i(t)\frac{\Delta t^3}{3!} + \mathcal{O}(\Delta t^4) \tag{2.28}$$

By adding and subtracting the updating equations for positions in Eq.2.24 and Eq.2.28, the following equations for the position and velocity will result.

$$\mathbf{r}_i(t+\Delta t) = 2\mathbf{r}_i(t) - \mathbf{r}_i(t-\Delta t) + \frac{\mathbf{f}_i(t)}{m_i}\Delta t^2 + \mathcal{O}(\Delta t^4) \tag{2.29}$$

$$\mathbf{v}_i(t) = \frac{\mathbf{r}_i(t+\Delta t) - \mathbf{r}_i(t-\Delta t)}{2\Delta t} + \mathcal{O}(\Delta t^3) \tag{2.30}$$

In the Verlet algorithm [188], the velocities are not needed to compute the trajectories; however, they are required to calculate observables like kinetic energy. In the Verlet scheme, the velocities $\mathbf{v}(t)$ are only available once the updated position $\mathbf{r}(t+\Delta t)$ has been calculated, i.e. one time step later. Moreover, the updating of positions according to the Verlet scheme gives rise to numerical imprecision as a small term of order Δt^2 is added to a difference of $\mathcal{O}(1)$-term.

Leap-frog algorithm

The leap-frog scheme [10] contains both the positions and velocities from readily available quantities. The updating equations are:

$$\mathbf{v}_i(t+\frac{\Delta t}{2}) = \mathbf{v}_i(t-\frac{\Delta t}{2}) + \frac{\mathbf{f}_i(t)}{m_i}\Delta t \tag{2.31}$$

$$\mathbf{r}_i(t+\Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t+\frac{\Delta t}{2})\Delta t \tag{2.32}$$

In this scheme, the velocities are updated at half-time steps and 'leap' ahead of the positions, and the current velocities can be obtained from

$$\mathbf{v}_i(t) = \frac{1}{2}[\mathbf{v}_i(t-\frac{\Delta t}{2}) + \mathbf{v}_i(t+\frac{\Delta t}{2})] \tag{2.33}$$

Numerical imprecision is minimized in the leap-frog scheme. However, the velocities are still not accessible in an ad-hoc manner [10].

Velocity Verlet algorithm

Velocity Verlet scheme [179], which is algebraically equivalent to the original Verlet scheme, yields the positions, velocities and forces at the same time and is a symplectic integrator, i.e. it preserves the volume in phase space. In this scheme, the positions and velocities are updated according to

$$\mathbf{r}_i(t+\Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t + \frac{\mathbf{f}_i(t)}{m_i}\frac{\Delta t^2}{2!} + \mathcal{O}(\Delta t^3) \tag{2.34}$$

$$\mathbf{v}_i(t+\Delta t) = \mathbf{v}_i(t) + \frac{\mathbf{f}_i(t) + \mathbf{f}_i(t+\Delta t)}{2m_i}\Delta t + \mathcal{O}(\Delta t^3) \tag{2.35}$$

Despite its simplicity, the Velocity Verlet scheme is very stable and has become the most widely used integrator in Molecular Dynamics simulations [179].

2.4 Thermostats

In an ensemble at the constant temperature where the system is allowed to exchange energy with its surroundings in MD, it is required to set the temperature of the domain (or a specific part of it) at a constant value. This constraint can be managed by using a thermostat during the simulation. To add and remove energy from an MD simulation in a more or less realistic fashion, several thermostat algorithms are available. Velocity rescaling, Nosé–Hoover thermostat, Berendsen thermostat, Andersen thermostat, and Langevin dynamics are the most popular ways of controlling temperature by emulating the system in contact with a constant temperature bath [67, 184]. Some of those thermostats are introduced briefly in the following.

The average kinetic energy of the system is related to the temperature by $K = 3Nk_BT/2$, thus, the first idea to keep the temperature is to rescale the particles' velocity in such a way that it generates the desired temperature. The most obvious factor for velocity rescaling is $\sqrt{T_{new}/T_{old}}$ according to the relation between particles' velocity (kinetic energy) and the system's temperature.

Nosé–Hoover thermostat

The Nosé-Hoover thermostat [94] is a temperature control system that "strives" to mimic the canonical phase space distribution. It does this by adding a non-Newtonian factor to the equations of motion to control the kinetic energy. The modified equation of motion is given by:

$$\frac{d\mathbf{v}_i(t)}{dt} = \frac{\mathbf{F}_i(t)}{m_i} - \zeta(t)\mathbf{v}_i(t) \tag{2.36}$$

where ζ is the thermodynamic friction coefficient,

$$\frac{d\zeta(t)}{dt} = \frac{1}{Q}\left[\sum m_i\mathbf{v}_i(t)^2 - (X+1)k_BT\right] \tag{2.37}$$

where X is the number of degrees of freedom, and Q is a parameter with the dimensions of energy $\times$ time^2 that controls the time scale of the temperature variation [94].

Berendsen thermostat

Berendsen thermostat [19] is another temperature-controlling tool for molecular dynamics simulations which uses a weak coupling (γ_i) to an external heat bath of temperature T_0. By using this thermostat, the equation of motion will take the following format,

$$\frac{d\mathbf{v}_i(t)}{dt} = \frac{\mathbf{F}_i(t)}{m_i} + \lambda(\frac{T_0}{T} - 1)\mathbf{v}_i(t) \tag{2.38}$$

which represents a rescaling of velocities per time step from $\mathbf{v}$ to $\lambda\mathbf{v}$ with λ being:

$$\lambda = \sqrt{1 + \frac{\Delta t}{\tau_T}(\frac{T_0}{T} - 1)} \tag{2.39}$$

where τ_T is a time constant that is associated with the strength of coupling.

Andersen thermostat

The Andersen thermostat [11] connects the system to a heat bath by modifying the kinetic energy of the atoms through stochastic forces. The duration between collisions, as well as the number of collisions in a given period, is determined randomly using the PDF of the exponential distribution:

$$P(t) = \nu e^{-\nu t} \tag{2.40}$$

where ν is the stochastic collision frequency. The new momentum of the atom is chosen randomly from a Boltzmann distribution at temperature T after a collision event. In theory, ν can take any value. There is, however, an ideal solution for the stochastic collision frequency:

$$\nu = \frac{2a\kappa V^{1/3}}{3k_B N} = \frac{2a\kappa}{3k_B \rho^{*1/3} N^{2/3}} \tag{2.41}$$

where a is a dimensionless constant and κ is the heat conductivity [11].

Langevin thermostat

In stochastic dynamics, the Langevin equation formulates the Brownian motion of particles in an ideal solvent [169, 203]. The Langevin equation of motion for a particle is in the following form:

$$m_i \frac{d^2\mathbf{r}_i(t)}{dt^2} = \mathbf{F}_i(\mathbf{r}_i(t)) - \gamma_i m_i \frac{d\mathbf{r}_i(t)}{dt} + R_i(t) \tag{2.42}$$

where γ_i is the friction coefficient due to the friction between particles and solvent and $R_i(t)$ is the random force caused by particles' interaction with the solvent. In addition to the inter-particle forces, the equation of motion includes a friction term (γ_i) which reduces the particles' kinetic and also a random fluctuation ($R_i(t)$) which increases the particles' kinetic. The random fluctuation term is chosen in such a way that its mean is zero and is uncorrelated in time. As both the friction and random force terms originate from the interaction of the particles with the solvent, they are related to each other by the following fluctuation-dissipation relation that recovers the canonical ensemble,

$$\langle R(t)R(t')\rangle = 2k_B T m_i \gamma_i \delta(t - t') \tag{2.43}$$

In many MD simulations of a solute-solvent system (e.g. DNA, protein, and nanoparticles in solution) in which the behaviour of the solute is desired, the Langevin equation can be used by assuming atoms being simulated are embedded in a sea of friction particles.

2.5 Force fields and interaction potentials

A force field is a computational tool used in chemistry and molecular modelling to determine the forces between atoms. In molecular dynamics, the force field refers to the functional form and parameters used to find the potential energy of a system of particles. The parameters for a given energy function might be generated via physics and chemistry experiments, quantum mechanics calculations, or a combination of both [67].

The interaction potentials in molecular dynamics can be categorized into non-bonded and bonded interaction potentials between atoms that are separated or bonded to each other. In the following, different contributions to the interaction potentials are explained.

2.5.1 Non-bonded interactions

The non-bonded interactions act between molecules or atoms that are not bonded to each other and are assumed to be pairwise additive which means $\phi(\mathbf{r}) = \sum_{i<j} U_{ij}(r_{ij})$. Consequently, the inter-particle force is $\mathbf{F}_i = -\mathbf{F}_j = -\nabla_i \phi_{ij}$. The non-bonded interaction includes the Van der Waals and electrostatic interactions [87].

Van der Waals interactions

The severe repulsion that arises at a short range and is caused by the overlap of the outer electron shells is the most essential property of the pair potential between atoms or molecules. The short-range order that is typical of the liquid state is created as a result of these intense repulsive forces. Attractive forces at large distances fluctuate more gradually with particle distance and have only a small impact on shaping the structure of the liquid. Instead, they provide a homogeneous background and generate the cohesive energy needed to stabilize the liquid [87].

The simplest model for a fluid is the hard-sphere model with the potential function of:

$$\phi(r) = \begin{cases} \infty, & r < d \\ 0, & r > d \end{cases} \tag{2.44}$$

with d being the hard-sphere diameter. The resulting structure of the fluid with the hard-sphere model does not differ significantly from those resulting from more complicated inter-atomic potentials. However, simulations show that it undergoes a freezing transition at $\rho^* = 0.945$ and the absence of attractive forces makes it a single fluid phase. A simple model that can truly describe the fluid system is the square-well model which includes an attractive potential part by introducing ϵ and γ (= 1.5 typically) which are the depth of the well and a constant for the width of the well, respectively.

$$\phi(r) = \begin{cases} \infty, & r < d \\ -\epsilon, & d < r < \gamma d \\ 0, & r > \gamma d \end{cases} \tag{2.45}$$

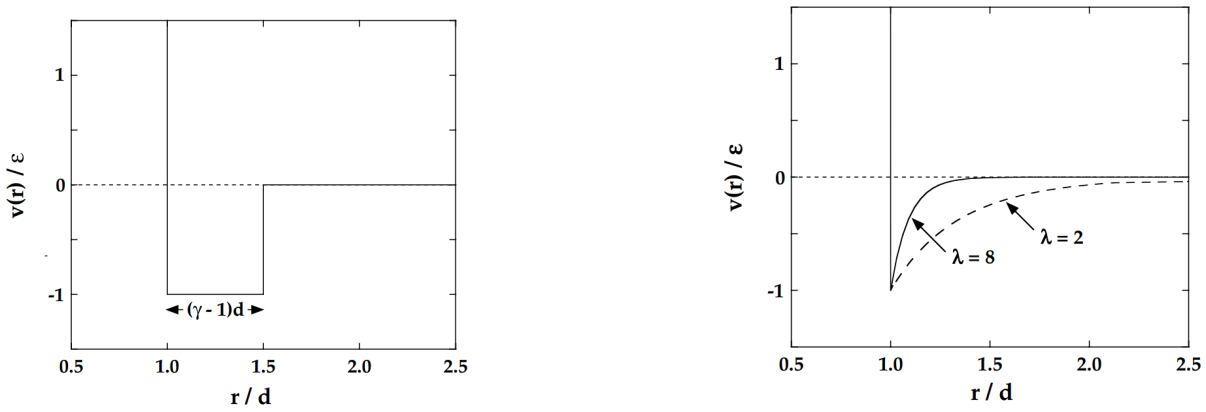
An alternative for the square-well potential function is the hard-core Yukawa potential that theoretically has features of particular interest,

$$\phi(r) = \begin{cases} \infty, & r < d \\ -\frac{\epsilon d}{r} e^{-\lambda(\frac{r}{d}-1)}, & r > d \end{cases} \tag{2.46}$$

where γ determines the inverse range of the attractive tail of the potential. The shape of the potentials for square-well and hard-core models are described in Fig.2.2 [87].

A more realistic potential can be constructed by a detailed quantum mechanics calculation in which at large separations, the attractive contribution varies with r^{-6} and higher-order terms r^{-8} and r^{-12} interactions which are smaller compared to the first term. The short-range interactions are relatively difficult to calculate but they can be approximated by $exp(-r/r_0)$ where r_0 is a range parameter that needs to be modified for the very small r values that should $\phi(r) \to \infty$. Thus, for mathematical convenience, it is usual to represent the short-range repulsion by an inverse power law r^{-n} that n usually is chosen between 9 and 15. With these criteria, the well-known $12-6$ potential of Lennard-Jones is in the following form:

$$\phi(r) = 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right] \tag{2.47}$$



(a) square-well potential

(b) Yukawa potential

Figure 2.2: Simple pair potentials for monatomic systems (a, b). The square-well model (a) is a modified version of the hard-sphere model with an additional attractive term and the Yukawa model (b) has a smooth form for the attractive part which makes it more interesting for theoretical studies. Two examples in case (b) correspond to the different λ values for rare-gas atoms ($\lambda = 2$) and short-range, attractive forces for certain colloidal systems($\lambda = 8$) [87].

This formulation includes two parameters that are the collision diameter σ which is the interparticle distance where $\phi(r) = 0$ and the depth of the potential well at the minimum of interparticle potential (ϵ). Lennard-Jones interactions are considered to be short-range as the inter-particle force resulting from it decays fast by increasing the inter-particle distance; thus, a cut-off radius (r_c) is usually considered in which particles do not interact with each other out of this radius [87].

Electrostatic interactions

The electrostatic interactions that occur between charged particles or molecules are long-range interactions compared to the Lennard-Jones interactions as they decay with r^{-1} (see Eq.2.48). Thus, it is not possible to ignore the long-range interactions similar to what is usually done in Lennard-Jones interactions by considering a cut-off radius.

In the most common methodology for computing electrostatic interactions (Ewald summation [10]), the interaction is divided into short-range and long-range contributions which are calculated in real and Fourier space respectively. In another approach, which is known as Reaction-Field method [182], the interactions are divided into two parts; one inside the cut-off region and one outside. For the electrostatic interactions within the cut-off distance, one may use the well-known Coulomb's law:

$$\phi(r) = \sum_{i=1}^{N} \sum_{j<i}^{N} \frac{q_i q_j}{r} \tag{2.48}$$

where r is the inter-particle distance and q_i is the charge of i^{th} particle. The particles outside the cut-off radius are treated as a cloud that produces a reaction field by assuming that they generate a dielectric medium(ϵ_s). The total energy from the reaction field outside the cut-off radius is:

$$\phi(r) = \sum_{i=1}^{N} \sum_{j \in r_c} \frac{1 - \epsilon_s}{r_c^3 (2\epsilon_s + 1)} \mu_i \mu_j \tag{2.49}$$

where μ is the dipole moment of each molecule [182].

2.5.2 Bonded interactions

Bonded interactions [186] are the interaction of atoms within the same molecule and are not limited to pair interactions but include 3- and 4-body interactions with bond stretching (2-body), bond angle (3-body) and dihedral angle (4-body) interactions. The bond stretching potential between two covalently bonded atoms can be calculated by harmonic potential as follows:

$$V_b(r_{ij}) = \frac{1}{2}k_{ij}^b(r_{ij} - b_{ij})^2 \tag{2.50}$$

where the corresponding force would be:

$$\mathbf{F}_i(\mathbf{r}_{ij}) = k_{ij}^b(r_{ij} - b_{ij})\frac{\mathbf{r}_{ij}}{r_{ij}} \tag{2.51}$$

with k_{ij}^b being the force constant and b_{ij} being the equilibrium bond length.

The vibration of the bond angle between a triple of atoms (i, j, and k) may be represented by a harmonic potential on the triple atoms angle (θ_{ijk}),

$$V_a(\theta_{ijk}) = \frac{1}{2}k_{ijk}^\theta(\theta_{ijk} - \theta_{ijk}^0)^2 \tag{2.52}$$

where k_{ijk}^θ is the force constant and θ_{ijk}^0 is the equilibrium triple bond angle. With Harmonic 3-body potential, the force on each atom may be calculated by the chain rule:

$$\mathbf{F}_i = \frac{\partial V_a(\theta_{ijk})}{\partial \mathbf{r}_i}, \qquad \mathbf{F}_k = \frac{\partial V_a(\theta_{ijk})}{\partial \mathbf{r}_k}, \qquad \mathbf{F}_j = -\mathbf{F}_i - \mathbf{F}_k \tag{2.53}$$

where $\theta_{ijk} = \arccos\frac{\mathbf{r}_{ij}.\mathbf{r}_{kj}}{r_{ij}r_{kj}}$. The labels of i, j, and k are the sequence of atoms in the covalent bond where atom j is in the middle and atoms i and k are on the sides.

For the vibrations of four body bonds which depend on the torsion angle (Θ_{ijkl}), called the dihedral potential, the harmonic potential reads as:

$$V_d(\Theta_{ijkl}) = \frac{1}{2}k_{ijkl}^\Theta(\Theta_{ijkl} - \Theta_{ijkl}^0)^2 \tag{2.54}$$

where k_{ijkl}^Θ is the force constant, Θ_{ijkl} is the angle between the plane where atoms i, j and k are located in and the plane for atoms j, k, and l, and Θ_{ijkl}^0 is the equilibrium angle between those planes [186].

2.6 Boundary conditions

In most molecular dynamic simulations, periodic boundary conditions (PBC) are used to prevent difficulties with boundary effects caused by the limited size and to make the system more resemble an infinite one, at the expense of potential periodicity effects [10].

Because of the particles' presence far away from the surfaces, periodic boundary conditions are used to measure the "bulk" qualities of the system. In most cases, a cubic simulation box is used to house the particles. This cubic box is reproduced in all directions; if a particle exits the core simulation box during a simulation run, one of its image particles enters the central box from the opposite direction. The image particles in adjacent boxes travel in the same direction, as seen in Fig.2.3 for a two-dimensional visualization [177].

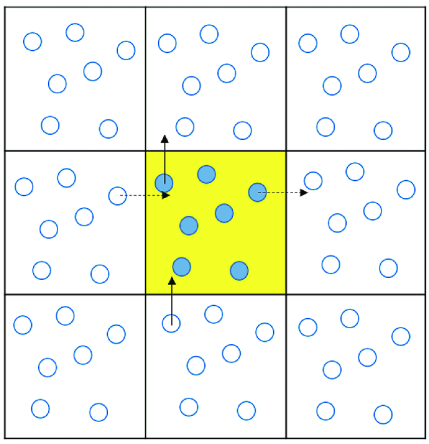


Figure 2.3: Periodic boundary conditions are represented in two dimensions. The paths of the particles in the central simulation box are duplicated in all directions.

Because of its simplicity, a cubic or rectangular box is generally exclusively employed in simulations with periodic boundaries. However, spherical boundary conditions have also been examined, where the three-dimensional surface of the sphere produces a non-Euclidean metric. The application of periodic boundary conditions allows bulk characteristics of systems with a limited number of particles to be simulated.

2.7 Reduced units

In MD simulations, it is common to describe quantities like temperature, density, and pressure in reduced units. This implies that we pick a convenient unit of energy, length, and mass, and then represent all other quantities in terms of these fundamental units [67]. It is common to create a set of dimensionless reduced units for simple liquids made of point-like particles that interact with the same pairwise potential of a generic form $U(r) = \epsilon f(r/\sigma)$. Here, ϵ and σ are constants and f is a smooth and differentiable function. A natural choice for the basic units are the units of length (σ), energy (ϵ), and mass (M, the mass of atoms in the system) and all other parameters can be calculated based on these basic units (see Tab.2.1).

By utilizing these reduced parameters which may be denoted by superscript $*$, the Lennard-Jones potential in its reduced form would be:

$$U^*(r^*) = 4\left[(\frac{1}{r^*})^{12} - (\frac{1}{r^*})^6\right] \tag{2.55}$$

in which the reduced pair potential is $U^* := U/\epsilon$ and the reduced distance is $r^* := r/\sigma$.

The first reason for using reduced units is that many combinations of ρ, T, ϵ, and σ correspond to the same state in reduced units. As an example, Argon at the density of $840[kg/m^3]$ and temperature of $60[K]$ is equivalent to Xenon at the density of $1617[kg/m^3]$ and temperature of $112[K]$ in reduced units which both cases translate to $\rho^* = 0.5$ and $T^* = 0.5$. This means that in real SI units one may run two different simulations for these two cases but in reduced units, one can run the same simulation with the corresponding reduced parameters and then convert the results by suitable coefficients according to Tab.2.1 to the real units. The other reason is that in many molecular simulations one deals with very small or very large numbers which may generate overflow or underflow errors after multiplication during simulation. In reduced units, all quantities of interest are in the order of 1 (say, between 10^{-3} and 10^3) which doesn't raise

Quantity	Symbol	Relation to SI
Length	r^*	$r\sigma^{-1}$
Mass	m^*	mM^{-1}
Time	t^*	$t\sigma^{-1}\sqrt{\epsilon/M}$
Temperature	T^*	$k_BT\epsilon^{-1}$
Energy	E^*	$E\epsilon^{-1}$
Force	F^*	$F\sigma\epsilon^{-1}$
Pressure	P^*	$P\sigma^3\epsilon^{-1}$
Velocity	v^*	$v\sqrt{M/\epsilon}$
Density	ρ^*	$N\sigma^3/V$

Table 2.1: Reduced Lennard-Jones quantities

an error for such scenarios and the result later can be converted into the real units.

In the next section, some physical properties of the molecular systems which are used for the validation of results within this thesis are described.

2.8 Physical properties

There are some physical properties of the fluid under study that have been considered in this research to prove the grand canonical behaviour of the method used here (Adaptive Resolution Simulation). This includes the radial distribution function and the chemical potential of substances that are explained in the following.

2.8.1 Radial Distribution Function

In statistical mechanics, the radial distribution function or pair correlation function ($g(r)$) determines how density (ρ) varies as a function of distance from a reference particle. With this definition, for a homogeneous and isotropic system, the local time-averaged density at a distance of r from a particle located at a certain point would be $\rho g(r)$. In another word, the radial distribution function determines the probability of finding a particle in the radius of r around a reference particle relative to that for an ideal gas.

For a system with N particles in volume V and at temperature T with inter-particle interactions $U(\mathbf{r}_1, ..., \mathbf{r}_N)$, the probability density in space is given by [87]:

$$P^{(N)}(\mathbf{r}_1, ..., \mathbf{r}_N) = \frac{e^{-\beta U}}{\mathcal{Q}_N} \tag{2.56}$$

By fixing the position of $n < N$ particles in $\mathbf{r}_1, ..., \mathbf{r}_n$ with no constraints on remaining $N - n$ particles, one may compute the probability of the reduced configuration by integrating Eq.2.56 over the remaining coordinates $\mathbf{r}_{n+1}, ..., \mathbf{r}_N$:

$$P^{(n)}(\mathbf{r}_1, ..., \mathbf{r}_n) = \frac{1}{\mathcal{Q}_N} \int ... \int e^{-\beta U} d\mathbf{r}_{n+1} ... d\mathbf{r}_N \tag{2.57}$$

By considering the case that any of n particles place at positions $\mathbf{r}_1, ..., \mathbf{r}_n$ in any permutation, the definition of n-particle density would be,

$$\rho^{(n)}(\mathbf{r}_1, ..., \mathbf{r}_n) = \frac{N!}{(N-n)!} P^{(n)}(\mathbf{r}_1, ..., \mathbf{r}_n) \tag{2.58}$$

This equation gives the one-particle density for $n = 1$ which for a homogeneous liquid is independent of the position $\mathbf{r}_1$ and equal to the system's density:

$$\frac{1}{V}\int \rho^{(1)}(\mathbf{r}_1)d\mathbf{r}_1 = \rho^{(1)} = \frac{N}{V} = \rho \tag{2.59}$$

By introducing the correlation function $g^{(n)}$:

$$\rho^{(n)}(\mathbf{r}_1, ..., \mathbf{r}_n) = \rho^n g^{(n)}(\mathbf{r}_1, ..., \mathbf{r}_n) \tag{2.60}$$

Thus, from Eq.2.58 and Eq.2.60, it follows that:

$$g^{(n)}(\mathbf{r}_1, ..., \mathbf{r}_n) = \frac{V^n N!}{N^n(N-n)!}\frac{1}{\mathcal{Q}_N}\int ... \int e^{-\beta U} d\mathbf{r}_{n+1}...d\mathbf{r}_N \tag{2.61}$$

where the second-order correlation function which is defined in the following form is the radial distribution function.

$$g^{(2)}(\mathbf{r}_1, \mathbf{r}_2) = \frac{V^2 N!}{N^2(N-2)!}\frac{1}{\mathcal{Q}_N}\int ... \int e^{-\beta U} d\mathbf{r}_3...d\mathbf{r}_N \tag{2.62}$$

2.8.2 Chemical Potential

In thermodynamics, the chemical potential of species in a chemical mixture is the energy that is added or absorbed from the system due to the change of the particle number for the specific species. Thus, the chemical potential of species in a mixture is defined as the rate of change of the free energy of the system with respect to the change of particle number for given species. In this sense, the fundamental equation of thermodynamics for the open system reads [87]:

$$dU = TdS - PdV + \sum_{i=1}^{n} \mu_i dN_i \tag{2.63}$$

where U, T, S, P, and V are the internal energy, temperature, entropy, pressure, and volume, respectively, with dX being the infinitesimal change in variable X. N_i is the particle number of species i. From Eq.2.63, the chemical potential is:

$$\mu_i = \left(\frac{\partial U}{\partial N_i}\right)_{S,V,N_{j\neq i}} \tag{2.64}$$

In a more convenient formalism, by taking Legendre transformation of Eq.2.63 and definition of Gibbs free energy $G = U + PV - TS$,

$$dG = -SdT - VdP + \sum_{i=1}^{n} \mu_i dN_i \tag{2.65}$$

which results in the following expression for the chemical potential:

$$\mu_i = \left(\frac{\partial G}{\partial N_i}\right)_{T,P,N_{j\neq i}} \tag{2.66}$$

This relation is more convenient for chemical potential calculation rather than Eq.2.64 as keeping the temperature and pressure constant is more feasible compared to the entropy and volume of the system while adding new particles.

It is possible to derive further expressions for the chemical potential by taking further Legendre

transformations from U and using enthalpy $H = U + PV$ and Helmholtz free energy $F = U - TS$ that leads to the following relations for the chemical potential [87]:

$$\mu_i = \left(\frac{\partial H}{\partial N_i}\right)_{S,P,N_{j\neq i}}, \qquad \mu_i = \left(\frac{\partial F}{\partial N_i}\right)_{T,V,N_{j\neq i}} \tag{2.67}$$

All the different forms of the chemical potential are equivalent and may be useful for different physical situations where keeping any thermodynamic property constant is more convenient.

The total chemical potential of a given species can be divided into two parts; one of them is the excess chemical potential (μ^{ex}) and the other is the ideal chemical potential (μ^{id}) that represents the chemical potential of an ideal gas under the same condition (i.e. pressure, temperature, and composition) [67].

$$\mu_i = \mu_i^{id} + \mu_i^{ex} \tag{2.68}$$

In molecular simulations, the chemical potential of the fluid can be calculated, for example, by the Widom insertion method explained in the following.

Widom insertion method

In statistical thermodynamics, the Widom insertion method is used to calculate the material and mixture properties. In general, there are two different approaches for determining the statistical mechanical properties of a system; the first one is to calculate the overall partition function and system's free energy and the second approach is to use the Widom insertion method which yields the chemical potential of the system instead of the system's free energy. The Widom insertion method can be translated as an application of Jarzynski equality [102, 103] as it measures the excess free energy when changing the state of the system from N particles to $N + 1$ particles:

$$\mu^{ex} = \frac{\Delta F_{excess}}{\Delta N} \tag{2.69}$$

where $\Delta N = 1$.

In the Widom insertion method, the insertion parameter B_i is defined in the following format:

$$B_i = \frac{\rho_i}{a_i} = \left\langle exp\left(-\frac{\psi_i}{k_B T}\right)\right\rangle \tag{2.70}$$

where ρ_i is the number density of species i, a_i is its activity, ψ is the interaction energy of the inserted particle with all other particles in the system, and $\langle ... \rangle$ denotes the average over all insertions. In this approach, the particles' locations are fixed and then the new particle will be inserted in different locations and the average will be calculated. Then, the insertion parameter would be related to the chemical potential of the system by the following equation:

$$\begin{aligned}\mu_i = -k_B T \times ln\left(\frac{B_i}{\rho_i \lambda^3}\right) &= k_B T \times ln(\rho_i \lambda^3) - k_B T \times ln\left(\left\langle exp\left(-\frac{\psi_i}{k_B T}\right)\right\rangle\right) \\ &= \mu^{id} + \mu^{ex}\end{aligned} \tag{2.71}$$

In the next section, the principles of the Adaptive Resolution Scheme (AdResS) are explained which is used to couple a full atomistic molecular dynamics subdomain to a lower resolution reservoir.

2.9 Adaptive Resolution Simulation (AdResS)

Many problems in complex soft matter systems are intrinsically multiscale, meaning that microscopic interactions are tightly connected to meso- and macroscopic features. Despite advances in computing power and continuous attempts to improve the efficiency of molecular-dynamics integration techniques, all-atom MD simulations are frequently unable to span the time and length scales required for relaxation in a typical molecular system, such as a polymer solution or melt [105,156]. Moreover, in many problems, there exists a small region of the domain which carries the most interest to study the details. Thus, it brings the idea of considering part of the domain with high-resolution (atomistic details) and the rest at a lower resolution (coarse-grained) [101,135].

In this respective, Adaptive Resolution Simulation (AdResS) is a particle-based multiscale molecular dynamics simulation technique that couples different regions with different resolutions while preserving the physical and thermodynamic properties of the full atomistic simulation of reference in the high-resolution region [156]. In AdResS, part of the domain is considered to have a higher resolution with a fully atomistic description of particles which corresponds to open system with grand canonical properties embedded in a reservoir of particles [160]. This multiscale approach has been refined over time starting from a reservoir of coarse-grained particles up to replacing them with non-interacting particles in its latest version [49].

2.9.1 Background

In general, in Adaptive Resolution simulations particles may change resolution from a full-atom representation to a coarse-grained region on-the-fly according to where they are located. There are two kinds of AdResS approaches and one of them is based on the Hamiltonian of the system (H-AdResS) in which all molecules need both atomistic description and a well-defined mapping point [88, 155]. Here, we use the version where a weight function is defined for the hybrid region between the all-atom region and the coarse-grained reservoir to adjust the inter-particle interactions between particles in different regions according to the all-atom and coarse-grained interactions [114, 132, 156, 160].

In the previous version of AdResS where the all-atom region is connected to the coarse-grained region through a small transition region, particles in different parts of the domain interact with each other via a space-dependent interpolation formula for the force acting between two molecules α and β,

$$F_{\alpha\beta} = w(x_\alpha)w(x_\beta)F^{AT}_{\alpha\beta} + (1 - w(x_\alpha)w(x_\beta))F^{CG}_{\alpha\beta} \tag{2.72}$$

where $F^{AT}_{\alpha\beta}$ and $F^{CG}_{\alpha\beta}$ are derived from the atomistic and coarse-grained potentials, respectively. The interpolation weight function $w(x)$ is defined as,

$$w(x) = \begin{cases} 1 & x < l_{AT} \\ \cos^2\left(\frac{\pi}{2l_\Delta}(x - l_{AT})\right) & l_{AT} < x < l_\Delta \\ 0 & > l_\Delta \end{cases} \tag{2.73}$$

where l_{AT} and l_Δ are the size of atomistic and transition regions [116].

In the current version of AdResS, the coarse-grained region is replaced by a reservoir of non-interacting ideal gas-like particles. The particles in the AT (atomistic region) and transition regions interact with each other with atomistic potentials while particles in the reservoir region (called tracer particles, TR) do not interact with each other nor with the atomistic particles [49]. To reproduce the thermodynamics and statistics of the full atomistic simulation of reference and

compensate for the removed degrees of freedom due to the abrupt change of resolutions, there should be some necessary tools which include a thermostat and an external force that is explained in the next part. Fig.2.4 shows the transition from old AdResS with coarse-grained particles to the new version with tracer particles.

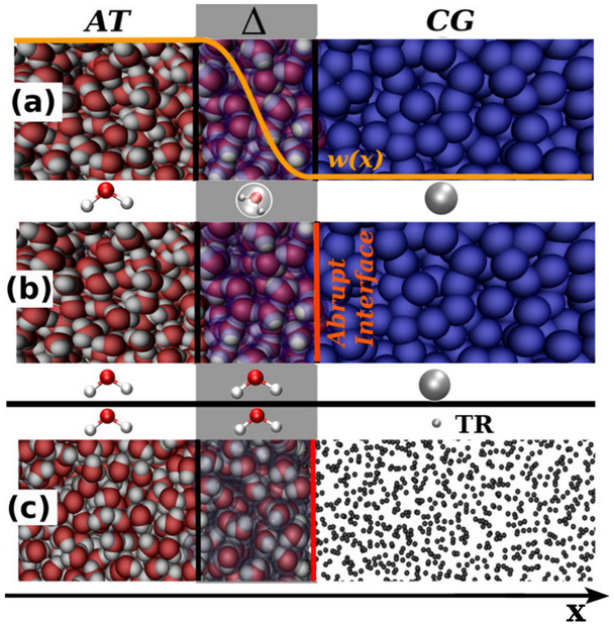


Figure 2.4: The evolution of Adaptive Resolution Set-up from the previous version with coarse-grained particles (panel a) in the low-resolution region and weight function $w(x)$ for interpolation of interaction potentials to the current version with the abrupt change of resolution (panel c) and non-interacting particles in the reservoir region [49].

2.9.2 Set-up and configuration

In AdResS, as mentioned above, the fully-atomistic region is located (usually) in the middle of the simulation domain and is connected to the reservoir of non-interacting particles (tracer region, TR) through a small hybrid region (Δ region), and is supposed to reproduce the thermodynamics and statistics of the fully atomistic simulation of reference. In Fig.2.5, the upper panel (a) shows the set-up of the simulation of reference and the lower panel (b) describes AdResS with red particles having the full atomistic resolution and black points are the dummy particles in the reservoir region. The atomistic and tracer regions are connected through a small region that has some characteristics of atomistic particles (they can interact with each other and particles in the atomistic region with particle-based potentials) and some of the reservoir particles (the thermostat and external force are acting on them).

Changing the resolution of particles from a full-atomistic representation to the tracer region with non-interacting particles and applying the abrupt change of resolutions will remove some degrees of freedom from the system. Thus, it is necessary to introduce some changes to the new system (AdResS) to make it resemble the original system of reference in both thermodynamics and dynamics of the system in the region of interest (AT). These include a thermostat and an external force applied in the reservoir region of AdResS ($TR \cup \Delta$) to preserve the thermodynamics and structure of the reference system. The external force (called thermodynamic force) is calculated self-consistently in an iterative manner during equilibrium runs and details of its

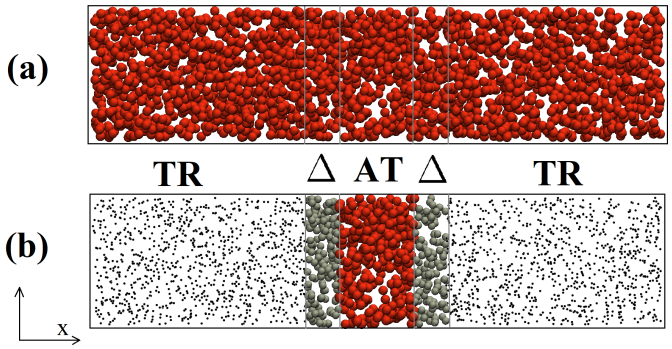


Figure 2.5: simulation set-up for AdResS with non-interacting particles in the reservoir region. Panel (a) shows the simulation set-up of the full atomistic simulation of reference and panel (b) shows the set-up of AdResS. Red particles show the particles with atomistic resolution and black point-like particles are the non-interacting particles of the reservoir region. They are connected by a small transition region with grey particles that have some properties of the atomistic particles and some of the tracer particles [74].

calculation method are explained in the next part. The last required tool is the capping of forces at the border of TR and Δ regions; this is a minor change in the system that does not affect the statistics of the system and is required to escape producing meaningless large interaction forces when dummy particles enter the Δ region on top of each other or at very small interparticle distances [73]. Further notes on the capping force are explained in Chapter 3.

2.9.3 Thermodynamic force

In the simulation domain with the explained decomposition in Fig.2.5, particles tend to move to the regions with lower energy levels in the absence of the external force. Thus, such partitioning will result in a non-uniform density profile in the equilibrium. This will result in an imbalance in the pressure of the system due to the abrupt change of resolution in the constant temperature by applying a thermostat in the reservoir region that is balanced in AdResS with the implementation of an external force (thermodynamic force) [74]. On the other hand, as numerically shown in this thesis, the thermodynamic force is a balancing tool for the chemical potential of the AdResS system according to the reference set-up [73].

In Refs. [68, 195], it has been shown that the thermodynamic force is balancing the resulting pressure difference due to the change of resolution. It has been shown in detail how such external force is expected to compensate for the imbalance resulting from the change of resolutions. With this idea, the following iterative formula for the calculation of the thermodynamic force with an initial zero force was proposed:

$$F_{th}^{i+1}(x) = F_{th}^{i}(x) - \frac{m}{\rho_0^2 \kappa_T} \nabla \rho^i(x) \tag{2.74}$$

where m is the particle's mass, ρ_0 is the target uniform density, and κ_T is the compressibility of the fluid. The derivative of density ($\nabla \rho^i$) is a normal derivative that means $\frac{\partial \rho^i}{\partial n} = \boldsymbol{n} \cdot \nabla \rho^i$ with $\boldsymbol{n}$ being the axes along which the change of resolution is happening. This iterative equation will find the required external force for adjusting the density of the system to the uniform density of the reference system usually within 10-30 steps.

2.9.4 Applications and validations

There are several applications studied by the Adaptive Resolution Scheme in recent years. It has been used for Path Integral Molecular Dynamics for liquid water [3, 5], macromolecular liquids [150], ionic liquids [46, 170], hydration of membranes [197], multiphase flow [127], etc. and in all of them, the method showed a well-enough resemblance to the reference set-up with coupling different regions at different resolutions.

To evaluate the function of AdResS and see how it reproduces the behaviour of the reference set-up, two mandatory tests are the temperature of the system and the density profile over the simulation domain. These two may assure that the adaptive resolution scheme is reproducing the thermodynamics of the original system of reference. Two other benchmarks that are usually calculated during AdResS simulations are the probability of finding particles (Gaussian distribution) and the radial distribution function. Their comparison to the reference set-up will validate that AdResS is keeping the structure and statistics of the original reference full-atomistic simulation.

Chapter 3

Chemical Potential calculation while changing the resolution in AdResS

This is the peer reviewed version of the following article:

A. Gholami, F. Höfling, R. Klein, and L. Delle Site. Thermodynamic relations at the coupling boundary in adaptive resolution simulations for open systems. *Advanced Theory and Simulations,* 4(4):2000303, 2021.

which has been published in final form at `https://doi.org/10.1002/adts.202000303`.

Abstract

The adaptive resolution simulation (AdResS) technique couples regions with different molecular resolutions and allows the exchange of molecules between different regions in an adaptive fashion. The latest development of the technique allows for abruptly coupling the atomistically resolved region with a region of non-interacting point-like particles. The abrupt set-up was derived from having in mind the idea of the atomistically resolved region as an open system embedded in a large reservoir at a given macroscopic state. In this work, starting from the idea of open system, we derive thermodynamic relations for AdResS which justify conceptually and numerically the claim of AdResS as a technique for simulating open systems. In particular, we derive the relation between the chemical potential of the AdResS set-up and that of its reference fully atomistic simulation. This result implies that the grand potential of AdResS can be explicitly written and thus, from a statistical mechanics point of view, the atomistically resolved region of AdResS can be identified with a well-defined open system.

3.1 Introduction

The adaptive resolution simulation (AdResS) technique couples, in a concurrent fashion, regions of space at different molecular resolutions [158, 160, 174]. Recent developments are pushing the method towards a computational realization of an open system embedded in a large reservoir of

particles and energy [6, 20, 45, 47, 48, 56, 113, 122]. The simulation set-up is reduced to the very essential by abruptly coupling an atomistically resolved region to a reservoir of non-interacting point particles [48]. The simplified algorithmic protocol, explained in detail in the next section, has the advantage of high computational efficiency and allows us to write a total interaction potential without making use of artificial, space-dependent, interpolations of atomistic and coarse-grained forces or Hamiltonians [60, 158, 159]. The abrupt coupling between the different regions may give the impression, at a first glance, of being highly artificial; in reality, physical consistency can be achieved by imposing specific numerical conditions. The latter assures that the AdResS simulation reproduces the results of the simulation of an equivalent subsystem in a large fully atomistic system of reference [195, 196].

In this perspective, the natural question arising is whether one can translate the numerical constraints into explicit thermodynamic and statistical mechanics relations occurring at the coupling region. This work demonstrates that arguments relying on physical consistency indeed lead to explicit thermodynamic descriptions of the AdResS set-up that positively pass specific numerical tests. The key result of the paper is the relation between the chemical potential of the atomistically resolved subsystem and the chemical potential of the fully atomistic system of reference. Such a relation, in turn, allows one to define the grand potential of the atomistic region of AdResS in terms of quantities that can be explicitly calculated from numerical simulations. The grand potential expresses the essential thermodynamic and statistical mechanics features of an open system. Thus the possibility of concretely defining the grand potential of AdResS at the microscopic level provides a robust justification for the idea of AdResS as a physically consistent numerical approach to open systems. The derivation of the thermodynamic relation is developed under ideal conditions which do not normally occur in standard simulations; however, numerical tests suggest that the obtained relations can be applied beyond the ideal conditions in which they have been derived.

The results of this chapter enrich the thermodynamic and statistical mechanics foundations of AdResS in its abrupt coupling approach and stimulate future deeper analysis of its several theoretical and numerical implications. The abrupt coupling approach allows for efficient simulations of complex systems such as, e.g., hydrated biological membranes [198]. While the standard atomistic simulation would require a sizable computational effort, the AdResS simulation, due to the drastically reduced number of degrees of freedom, runs at a reduced computational price. It must be noticed that the initial calibration and validation of AdResS requires reference data from fully atomistic simulations. However, for example in solvation studies, the largest portion of the AT region represents bulk liquid and thus calibration and validation of the parameters of AdResS can be done on smaller systems whose size is sufficient to represent only the bulk. The computational price of such test systems, even at a fully atomistic level, is negligible (see e.g. Refs. [4, 100, 198]).

3.2 Basic principles of AdResS with a reservoir of non-interacting particles

Fig.3.1 (a) illustrates the AdResS set-up; this latter consists of partitioning the simulation box into three regions: the region of interest AT, with a fully atomistic resolution, the interface region Δ, with fully atomistic resolution but with additional coupling features to the large reservoir, and TR, the large reservoir of non-interacting particles (tracers). Molecules of the AT region interact with atomistic potentials among themselves and with molecules in Δ, and vice versa, while there is no direct interaction with the tracer particles. Tracers and molecules in the Δ and TR regions are subject to an additional one-body force $F_{th}(x)\,\vec{n}$, named thermodynamic force,

acting along the direction $\vec{n}$ in which the change of resolution takes place; it is a function of the distance x from the atomistic region and $\vec{n}$ is the surface normal of the coupling boundary (Fig.3.1 (c)). Second, a thermostat acts on the Δ and TR regions that compensates for the heat introduced by the change of resolution [56, 195]. In essence, these are the coupling condition between the Δ region and the reservoir TR.

Technically, also a force capping is imposed in the Δ region since point-like particles arriving from the TR region and entering the Δ region may be unphysically close to one other. Due to the abrupt switching of molecular degrees of freedom, close molecules can experience forces between atoms which are artificially large. Admittedly, the force capping is an artificial means by which unphysically large forces are automatically relaxed to the average force occurring in the equivalent fully atomistic simulation. The capping, however, is equivalent to a global modification of the highly repulsive part of the interaction potentials, which has marginal repercussions on the physical properties of the fluid. The exact form of the force capping is given later in this chapter, where we also report numerical tests showing that its effects can be neglected. Recently, in Ref. [181], it has been proposed an alternative approach to circumvent the problem of unphysical large forces at the interface between Δ and TR. It is based on an energy minimization procedure for the insertion of particles from the TR region to the Δ region, as originally suggested in Ref. [113]. Compared to the procedure employed here, the procedure of Ref. [181] is computationally more demanding however it can be useful for liquids composed of large molecules with complex chemical architecture.

In summary, the total potential of the AdResS set-up reads:

$$U_{Ad}(x_N) = U(x_N) + \Phi_\Delta(x_N) + U_{cap}(x_N) \tag{3.1}$$

assuming that at a given instance in time N particles are found in the $AT \cup \Delta$ region with positions $x_N = \{\vec{r}_1, \ldots, \vec{r}_N\}$. Here, $U(x_N)$ represents the total potential from atomistic interactions of particles in $AT \cup \Delta$ among themselves; $\Phi_\Delta(x_N) := \sum_{\vec{r}_j \in \Delta} \phi_{th}(\vec{r}_j)$ collects the contributions due to the potential ϕ_{th} of the thermodynamic force, $\mathbf{F}_{th}(x) = -\nabla_x \phi_{th}(\vec{r})$ with $\phi_{th} = 0$ at the AT/Δ interface [48]. Finally, $U_{cap}(x_N)$ arises from the force capping and is only present in the Δ region.

The effect of $\vec{F}_{th}(r)$ consists in enforcing a homogeneous molecular density in the Δ region equivalent to the molecular density ρ_{at} in the equilibrium of the reference fully atomistic system. In practice, it is calculated self-consistently in an iterative process, starting from $F_{th}^{(0)}(x) = 0$. The update between successive steps k is $F_{th}^{(k+1)}(x) = F_{th}^{(k)}(x) - c\,\nabla\rho_k(x)$, where the density profile $\rho_k(x)$ was calculated from an AdResS simulation using $F_{th}^{(k)}(x)$ and $c > 0$ is a suitable coefficient to control the speed of convergence. The iteration stops when the deviation of $\rho_k(x)$ from a constant profile is within a prescribed tolerance, details are given in the technical details. After $F_{th}(x)$ has been determined, it remains unchanged in the whole AdResS production run without recalibration [68, 195].
The development of the abrupt computational set-up with tracers became possible through the mapping of the algorithm onto a theoretical model of open systems. Such a model fixes a series of conditions that the AdResS simulation must fulfil to be considered valid. Such conditions are sufficient to assure that the physics of the AT region is correct and, in the limit of large TR and AT regions compared to Δ, one has a Grand Canonical-like ensemble for AT (GC-AdResS) [30]. The application of the thermodynamic force in Δ is one of these conditions because it assures that the particle density in Δ is equal to the atomistic target density at equilibrium. Ideally one would want to make sure that the interactions of particles near the boundary of the AT region with their neighbours are statistically isotropic, i.e. independent of whether neighbours have located within the AT or Δ region. Matching the densities between the AT and Δ region is a necessary condition for achieving this.

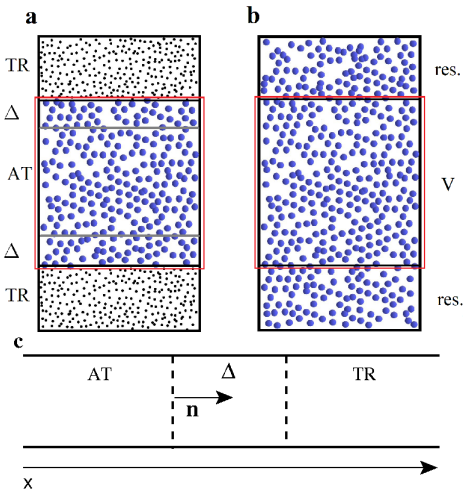


Figure 3.1: Simulation set up in the (a) AdResS system and (b) fully atomistic system. In panel (c) it is shown the x-direction along which the change of resolution occurs and the vector $\mathbf{n}$, normal to the coupling boundaries, that defines the direction of the thermodynamic force. In AdResS particles change resolution when crossing the border between the Δ and TR region (reservoir with tracer particles). In both systems, the red boxes represent the subsystem analyzed in this work. It must be underlined that the AT region is the region of physical interest, and the Δ region is an AdResS-artifact through which the coupling to the non-interacting particle reservoir becomes technically possible. Here we extend the analysis to the Δ region so that its coupling conditions can be rationalized in terms of thermodynamic quantities of the joint $AT \cup \Delta$ region.

Going one step further, it is required that the probability distribution $p(N)$ of the number of particles in the AT region should be the same, within some accuracy, as $p(N)$ of the equivalent subsystem in a fully atomistic simulation. The fulfilment of such a condition assures one that, on average, the exchange of particles between the AT region and the reservoir occurs properly. For a Gaussian distribution, it is sufficient to compare the first two cumulants, which are related to the density and the compressibility, where the latter provides a particularly sensitive test of the boundary conditions [98]. In the language of statistical mechanics, the equivalence of the n-th moments of $p(N)$ between AdResS and the reference system guarantees that the n-th derivatives of the grand potentials of the two systems with respect to the chemical potential agree and vice versa. Additionally, microscopic structural consistency is assured by matching atom-atom radial distribution functions in the AT region. Finally, one could also verify that the interaction energy between molecules in AT and molecules in Δ is negligible compared to the interaction energy amongst the molecules in AT so that the physics of the system is determined only by the interactions between the molecules in AT, that is the physical system of interest.

3.3 Relation between the chemical potential of AdResS and of a fully atomistic system of reference

3.3.1 Principle of equivalence for the grand potential

In an open system, the relevant thermodynamic state potential is the grand potential $\Omega = -PV$, where V and P denote the volume and the pressure of the system. It has the microscopic expression

$$\Omega = -k_B T \ln \left(\sum_{N=0}^{\infty} \mathrm{e}^{\beta \mu N} Q_N \right), \tag{3.2}$$

if the system is equilibrated at the chemical potential μ and the temperature T; as usual $\beta = 1/k_BT$ with k_B Boltzmann's constant. The partition function at a fixed number N of identical particles reads

$$Q_N = \frac{1}{h^{3N}N!}\int_{\mathbb{R}^{3N}}\int_{V^N} e^{-\beta H_N(x_N,p_N)}dx_N dp_N, \tag{3.3}$$

where $p_N = \{\vec{p}_1,\ldots,\vec{p}_N\}$ and $x_N = \{\vec{r}_1,\ldots,\vec{r}_N\}$ are the momenta and positions of the N particles, respectively. The Hamiltonian is the sum of kinetic and interaction potential energies:

$$H_N(x_N,p_N) = \sum_{i=1}^{N}\frac{\vec{p}_i^2}{2m} + U(x_N), \tag{3.4}$$

and m is the particle mass. It must be noticed that here the potential $U(x_N)$ contains interactions only between particles in the system and neglects any potential interaction with the exterior (see e.g. Ref. [97]).

For a subsystem S in a fully atomistic system of reference, whose domain is equivalent to the $S = AT \cup \Delta$ region of the AdResS set-up, let us define the grand potential of the reference system as

$$\langle\Omega_r\rangle = -k_BT\ln\left(\sum_{N=0}^{\infty}\mathrm{e}^{\beta\mu_r N}Q_N^r\right), \tag{3.5}$$

where we denoted the chemical potential of the reference subsystem by μ_r and introduced the effective N-particle partition function

$$Q_N^r := \left\langle\frac{1}{h^{3N}N!}\int_{\mathbb{R}^{3N}}\int_{S^N} e^{-\beta H_N^r(x_N,p_N|x'_M)}dx_N dp_N\right\rangle_{\delta S}. \tag{3.6}$$

Here, the extended Hamiltonian

$$H_N^r(x_N,p_N|x'_M) = \sum_{i=1}^{N}\frac{\vec{p}_i^2}{2m} + U(x_N) + U(x_N,x'_M) \tag{3.7}$$

expresses the fact that the N molecules of the subsystem do not interact only among themselves, but also with M molecules located in a layer δS around the S region. The angular brackets in Eq.3.6 denote an averaging operation over the positions $x'_M \in \delta S$ of these reservoir particles, which, however, are correlated with other particles of the reservoir outside of δS. Mathematically, the probability density $p_{\delta S}(x'_M)$ of the positions in δS is obtained by the marginalization of the phase space density of the universe (subsystem plus reservoir), see also Ref. [47]. Later on, we will specify how one performs the marginalization w.r.t. the outer particles in a simulation where M changes dynamically. If we assume that $U(x)$ is a potential with a sufficiently short interaction range, the volumes obey $|\delta S| \ll |S|$, and thus the integration over the M particles represents a surface effect, we can, in good approximation, identify $\langle\Omega_r\rangle$ with the grand potential of the S region (see also Ref. [97]).

Next, we consider the $S = AT \cup \Delta$ region of the AdResS set-up *without* the thermodynamic force acting in the Δ region, i.e., without the potential energy contribution $\phi_{th}(x)$. Denoting by μ_0 the chemical potential of this subsystem in absence of the thermodynamic force, and the corresponding grand potential Ω_{Ad}^0 is defined as

$$\Omega_{Ad}^0 = -k_BT\ln\left(\sum_{N=0}^{\infty}\mathrm{e}^{\beta\mu_0 N}Q_N^{Ad,0}\right) \tag{3.8}$$

with

$$Q_N^{Ad,0} = \frac{1}{h^{3N}N!}\int_{\mathbb{R}^{3N}}\int_{S^N} e^{-\beta H_N^{Ad,0}(x_N,p_N)}dx_N dp_N \tag{3.9}$$

and the Hamiltonian

$$H_N^{Ad,0}(x_N, p_N) = \sum_{i=1}^{N} \frac{\vec{p}_i^2}{2m} + U(x_N) + U_{cap}(x_N)\,; \tag{3.10}$$

the latter follows from Eq.3.1. As argued above, the number of capping events is negligible and we will neglect the term $U_{cap}(x_N)$ in the following, so that $H_N^{Ad,0}$ reduces to H_N as in Eq.3.4.

The purpose of AdResS is to reproduce the physics of the reference fully atomistic simulation in the AT region. If the AdResS set-up, with Hamiltonian $H_N^{Ad,0}$, was sufficient to this aim one should have: $\langle \Omega_r \rangle = \Omega_{Ad}^0$, but it is easy to numerically verify that this is never the case. However, as described in the previous section, adding a one-particle potential in the Δ region of the AdResS set-up is sufficient to enforce the physical consistency between AdResS and its reference fully atomistic system. In the following subsection, we will interpret the inclusion of the potential of the thermodynamic force in AdResS through the idea of equivalence of the grand potential between AdResS and its fully atomistic system of reference.

3.3.2 Perturbation of the potential energy in the Δ region

Let us anticipate the thermodynamic limit so that the size of AT is arbitrarily large and $|AT| \gg |\Delta|$. Under such conditions, we add a small perturbation to the potential of the Δ region in AdResS. Let us assume that such a perturbation can be designed in such a way that we achieve the wished relation of thermodynamic equivalence between the AdResS set-up and its reference simulation:

$$\langle \Omega_r \rangle = \Omega_{Ad}\,; \tag{3.11}$$

equality holds also for all derivatives of the two grand potentials w.r.t. the variables μ, V, T.

In the actual AdResS numerical simulation, the potential of the thermodynamic force $\phi_{th}(x)$ acting in Δ assures the approximate statistical equivalence between the AdResS simulation and its fully atomistic simulation of reference within AT, at least for the one-particle density and the pair (radial) distribution function. Thus, $\phi_{th}(x)$ is a reasonable approximation to the perturbation needed. In the presence of a perturbation, one can assume that physical quantities of interest in $AT \cup \Delta$ remain, in first approximation, as they were in absence of the perturbation and that the effect of the perturbation can be explicitly derived and added to them. This argument allows us to write in good approximation for the grand potential of the equilibrated AdResS set-up:

$$\Omega_{Ad} = -k_B T \ln \left(\sum_{N=0}^{\infty} \mathrm{e}^{\beta(\mu_0 + \Delta\mu)N} Q_N^{Ad} \right) \tag{3.12}$$

with

$$Q_N^{Ad} = \frac{1}{h^{3N} N!} \int_{\mathbb{R}^{3N}} \int_{S^N} e^{-\beta H_N^{Ad}(x_N, p_N)} dx_N dp_N\,, \tag{3.13}$$

in which $H_N^{Ad}(x_N, p_N) = H_N^{Ad,0}(x_N, p_N) + \Phi_\Delta(x_N)$. Here, we have denoted the chemical potential of the perturbed system by $\mu_0 + \Delta\mu$ and assumed, according to the above definition of perturbation, that the difference $\Delta\mu = \Delta\mu[\phi_{th}]$ originates from the perturbation of the potential energy in Q_N^{Ad}. In order to arrive at an explicit thermodynamic relation between μ_r, μ_0 and $\Delta\mu$, we will derive explicit expressions of Q_N^r and Q_N^{Ad} in the subsection below.

3.3.3 Relation between μ_r, μ_0 and $\Delta\mu$

Following the standard arguments in statistical mechanics [97], the sum over N in the expressions Eq.3.5 and Eq.3.12 for $\langle\Omega_r\rangle$ and Ω_{Ad} represents a major obstacle to the derivation of a direct relation between μ_r, μ_0 and $\Delta\mu$. However, given the conditions of the thermodynamic limit for S we can assume that $p(N)$ is sharply peaked around $\bar{N}$, with $\bar{N}$ being the average number of particles in S. Under such assumption, the sum over N can be approximated by its most relevant term, that is the term of the series corresponding to $\bar{N}$. It follows that the condition $\langle\Omega_r\rangle = \Omega_{Ad}$ [Eq.3.11] implies:

$$-k_BT\ln\left(e^{\beta\mu_r\bar{N}}Q^r_{\bar{N}}\right) = -k_BT\ln\left(e^{\beta(\mu_0+\Delta\mu)\bar{N}}Q^{Ad}_{\bar{N}}\right), \tag{3.14}$$

or equivalently,

$$\beta\mu_r\bar{N} + \ln Q^r_{\bar{N}} = \beta\mu_0\bar{N} + \beta\Delta\mu\bar{N} + \ln Q^{Ad}_{\bar{N}}, \tag{3.15}$$

which becomes exact in the thermodynamic limit. The error of the approximation can be estimated by considering $\overline{N} \pm \Delta N$ as upper and lower bounds on $\overline{N}$ in our calculations, with the standard deviation $\Delta N = \sqrt{\langle N^2\rangle - \langle N\rangle^2}$. One may ask whether the hypothesis of $P(N)$ sharply distributed around $\overline{N}$ is automatically fulfilled in AdResS simulations. In general, this may not be the case when AT is not sufficiently large compared to Δ. In any case, $P(N)$ can be calculated in AdResS while the simulation runs without additional costs and thus one can automatically check the degree of validity of the hypothesis in actual simulations. Moreover, as suggested above, the accuracy of Eq.3.15 can be tested by considering the upper and lower bound of $\overline{N}$. The use of $\overline{N}+\Delta N$ and $\overline{N}-\Delta N$ in Eq.3.15 instead of $\overline{N}$, will lead to a relation between μ_r, μ_0 and $\Delta\mu$ that quantitatively differs from the one where $\overline{N}$ is used. Once a threshold for the desired accuracy of the simulation is given, the difference above can be used as an additional criterion of the validity of the original hypothesis, that is to decide whether or not $P(N)$ is sharp enough. It follows that in an AdResS simulation the fulfilment of the hypothesis that $P(N)$ is sharply distributed around $\overline{N}$ is under the control of the simulator at any time.

In the next step, we will rewrite the expressions for the $\bar{N}$-particle partition functions $Q^r_{\bar{N}}$ and $Q^{Ad}_{\bar{N}}$. Let us first consider the partition sum of the equilibrated AdResS setup, given in Eq.3.13:

$$Q^{Ad}_{\bar{N}} = \frac{1}{h^{3\bar{N}}\bar{N}!}\int_{\mathbb{R}^{3\bar{N}}}\int_{S^{\bar{N}}} \mathrm{e}^{-\beta H_{\bar{N}}(x_{\bar{N}},p_{\bar{N}})} \times \mathrm{e}^{-\beta\Phi_\Delta(x_{\bar{N}})} dp_{\bar{N}}dx_{\bar{N}}, \tag{3.16}$$

which is nothing else than a quantity proportional to the canonical average of $\mathrm{e}^{-\beta\Phi_\Delta(x_{\bar{N}})}$ w.r.t. the region S, i.e.,

$$Q^{Ad}_{\bar{N}} = Q_{\bar{N}}\langle \mathrm{e}^{-\beta\Phi_\Delta(x_{\bar{N}})}\rangle. \tag{3.17}$$

The evaluation of $Q^r_{\bar{N}}$, defined in Eq.3.6, implies the knowledge of the statistics $p_{\delta S}(x'_M; M)$ of the values of M and the positions x'_M in the shell δS. For the moment let us assume that it is known and the average w.r.t. x'_M can be carried out. For $Q^r_{\bar{N}}$, this average is spelled out in integral form as

$$Q^r_{\bar{N}} = \sum_{M=0}^{\infty}\int_{\delta S^M}\frac{1}{h^{3\bar{N}}\bar{N}!}\int_{\mathbb{R}^{3\bar{N}}}\int_{S^{\bar{N}}} \mathrm{e}^{-\beta H_{\bar{N}}(x_{\bar{N}},p_{\bar{N}})}\times\mathrm{e}^{-\beta U(x_{\bar{N}},x'_M)} dx_{\bar{N}}dp_{\bar{N}}\, p_{\delta S}(x'_M; M)\, dx'_M. \tag{3.18}$$

This expression can also be interpreted as a canonical average of $\langle\exp(-\beta U(x_{\bar{N}}, x'_M))\rangle_{\delta S}$ over the positions and momenta of the $\bar{N}$ particles in S, after dividing by a suitable normalization factor that coincides with the $\bar{N}$-particle partition function $Q_{\bar{N}}$ given in Eq.3.3. We obtain

$$Q^r_{\bar{N}} = Q_{\bar{N}}\left\langle\langle \mathrm{e}^{-\beta U(x_N,x'_M)}\rangle_{\delta S}\right\rangle \tag{3.19}$$

with the double brackets referring, first, to the average over the statistics of M and the positions $x'_M \in \delta S$ and, second, to the canonical average over $x_{\bar{N}} \in S$. In numerical simulations, the

statistics over M are extracted from a sufficiently long simulation run and the corresponding average over the time series takes properly into account the integration over x'_M in Eq.3.18.

Collecting the results of this section and by substituting Eq.3.19 and Eq.3.17 in Eq.3.14, one obtains

$$\mathrm{e}^{\beta\mu_r\bar{N}}\langle \mathrm{e}^{-\beta U(x_{\bar{N}},x'_M)}\rangle = \mathrm{e}^{\beta(\mu_0+\Delta\mu)\bar{N}}\langle \mathrm{e}^{-\beta\Phi_\Delta(x_{\bar{N}})}\rangle \tag{3.20}$$

and, one step further, an explicit formula that links μ_r, μ_0 and $\Delta\mu$:

$$\mu_r - (\mu_0 + \Delta\mu) = \omega_\Delta - \omega_r\,. \tag{3.21}$$

Here, the energies

$$\omega_\Delta := (\beta\bar{N})^{-1}\ln\langle \mathrm{e}^{-\beta\Phi_\Delta(x_{\bar{N}})}\rangle \tag{3.22}$$

and

$$\omega_r := (\beta\bar{N})^{-1}\ln\langle \mathrm{e}^{-\beta U(x_{\bar{N}},x'_M)}\rangle, \tag{3.23}$$

are, respectively, the contribution of the potential of thermodynamic force and the pulled-out interactions of particles in the open system with those in the reservoir.

Interestingly, μ_r, μ_0, ω_Δ, and ω_r can be calculated numerically within fully atomistic and AdResS simulations. In particular, ω_Δ and ω_r contain the terms $\langle \mathrm{e}^{-\beta\Phi_\Delta(x_{\bar{N}})}\rangle$ and $\langle \mathrm{e}^{-\beta U(x_{\bar{N}},x'_M)}\rangle$, which, as for the sampling w.r.t. the x'_M states discussed above, are calculated by sampling x_N over a sufficiently long trajectory and averaging over the time series. Since we have assumed to work under the condition that $p(N)$ is sharply distributed around $\bar{N}$, the dominant configurations in the sampling along the trajectory are those with $\bar{N}$ particles in the subsystem. This means that we can identify with good approximation, $\langle \mathrm{e}^{-\beta\Phi_\Delta(x_{\bar{N}})}\rangle$ and $\langle \mathrm{e}^{-\beta U(x_{\bar{N}},x'_M)}\rangle$ with $\langle \mathrm{e}^{-\beta\Phi_\Delta(x_N)}\rangle$ and $\langle \mathrm{e}^{-\beta U(x_N,x'_M)}\rangle$, respectively, calculated from the simulation.

The possibility of calculating numerically the quantities above implies that indeed the grand potential of the AT region of AdResS, Ω_{Ad}, within the assumptions made, can be explicitly written. In turn, the explicit definition of the grand potential from a microscopic (first principle of statistical mechanics) perspective legitimates the definition of AdResS as a method of open systems that is well-founded on statistical mechanics.

3.4 Technical Details and Validation of AdResS

For validation of AdResS, a variety of LJ fluids with different state points along with a water model at biological conditions have been studied. The LJ fluid particles are of mass m and interact pairwise with the sharply truncated and shifted LJ potential $U(r) = U_{LJ}(r) - U_{LJ}(r_c)$ for $r < r_c$ and $U(r) = 0$ otherwise; the cut-off distance was chosen as $r_c = 2.5\sigma$ and the original LJ potential reads $U_{LJ}(r) = 4\epsilon[(r/\sigma)^{-12} - (r/\sigma)^{-6}]$. The parameters ϵ and σ serve as intrinsic units for energy and length, respectively; the unit of time is set to $\tau := \sqrt{m\sigma^2/\epsilon}$. For the case of water, in addition to the mentioned pair interactions, electrostatic potentials are also included with a cut-off radius of 1.2 nm.

The LJ fluids were kept at the (dimension-reduced) temperature $T^* := k_BT/\epsilon = 1.5$, which is well above the liquid–vapour critical point, and we investigated four different number densities $\rho^* := \rho\sigma^3 \approx 0.20$, 0.25, 0.30, and 0.37, corresponding to particle numbers $N =$ 8k, 10k, 12k, and 15k, where k stands for the SI prefix for 10^3. In the case of water, 58 990 water molecules (i.e. 176 970 atoms) at a biological temperature of 323 K have been considered for simulations.

In the corresponding AdResS set-ups, the same particle numbers were used for the total of LJ and tracer particles. The LJ particles were confined to a cuboid simulation box of size $45\sigma \times 30\sigma \times 30\sigma$ (for the case of water: 33.09 nm × 7.37 nm × 7.37 nm), with periodic boundaries

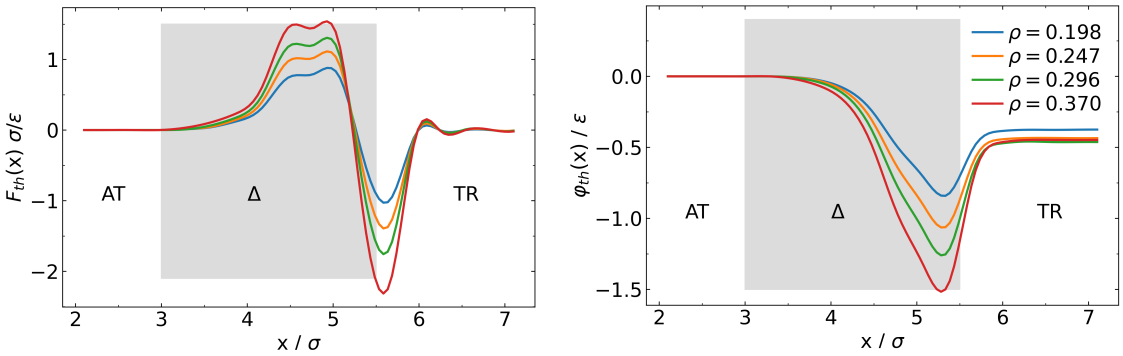


Figure 3.2: Thermodynamic force $F_{th}(x)$ (left) and its potential $\phi_{th}(x)$ (right) used in the AdResS set-ups for Lennard-Jones fluids at the same temperature $T^* = 1.5$ and at four reduced densities. The thermodynamic force is zero in the AT region by construction and vanishes rapidly inside the TR region.

imposed at all faces. For the AdResS set-up, the width of the transition region Δ along the x-axis was set to the cut-off radius, $L_\Delta = r_c$, which provides sufficient space and time for the proper equilibration of particles that entered from the Δ/TR border and changed their resolution abruptly before they reach the AT region of interest. The width of the AT region was chosen as $L_{AT} = 6\sigma$ for LJ cases and $L_{AT} = 10\,\text{nm}$ for water simulation, which is small enough to reduce the computational cost significantly compared to a fully atomistic simulation and large enough to be able to mimic and reproduce the thermodynamics and structure of the fluid under study. The remaining part of the simulation box ($L_{TR} = 34\sigma$ for LJ cases and $L_{TR} = 20.69\,\text{nm}$ for water) is filled with non-interacting particles (tracers). For the fully atomistic simulations serving as the reference, the same geometry of the simulation box was used (Fig.3.1a) and observables were computed only in a subvolume of width L_{AT} along the x-axis, corresponding to the AT region of the AdResS set-up.

All simulations were carried out with the GROMACS software [136] using the stochastic leap-frog integrator with timestep 0.002τ, which acts as a Langevin-type thermostat with the time constant set to 0.05τ. The production runs covered $10^3\tau$ to calculate thermodynamic and statistical properties within the AdResS simulation. The threshold for capping the force on a particle in the Δ region was set to $F_{cap} = 500\epsilon/\sigma$ and was applied separately for each Cartesian component of the force. Excess chemical potentials were computed in standard MD simulations using Widom's method [199], in particular, 10 k test particles were inserted after each interval of 2τ.

In the case of the AdResS set-up and for each density, the thermodynamic force $F_{th}(x)$ was calculated iteratively as described above with the stopping criterion chosen as $\max|\rho(x)-\rho^*|/\rho^* \leq 2\%$; the maximum is taken across the whole simulation box. The thermodynamic force $F_{th}(x)$ was parameterized in terms of a cubic spline interpolation with knot distance 0.3σ. On average, 10–15 iterations were needed for this scheme to converge, and each iteration involved a simulation run over 200τ.

The resulting curves for $F_{th}(x)$ are shown in Fig.3.2 along with the corresponding potentials $\phi_{th}(x)$ obtained from the integration of the force. The main feature of the potentials is a minimum in the Δ region, close to the Δ/TR boundary ($x = 5.5\sigma$), with the depth increasing by a factor of 2 as the density of the fluids is increased from $\rho^* = 0.20$ to $\rho^* = 0.37$. Inside the TR region, the potential converges within a distance of $\approx 1\sigma$ from the Δ/TR boundary to a constant $\phi_{TR} \approx -0.45\epsilon$, i.e., below the value in the AT region. The value of ϕ_{TR} varies only mildly with

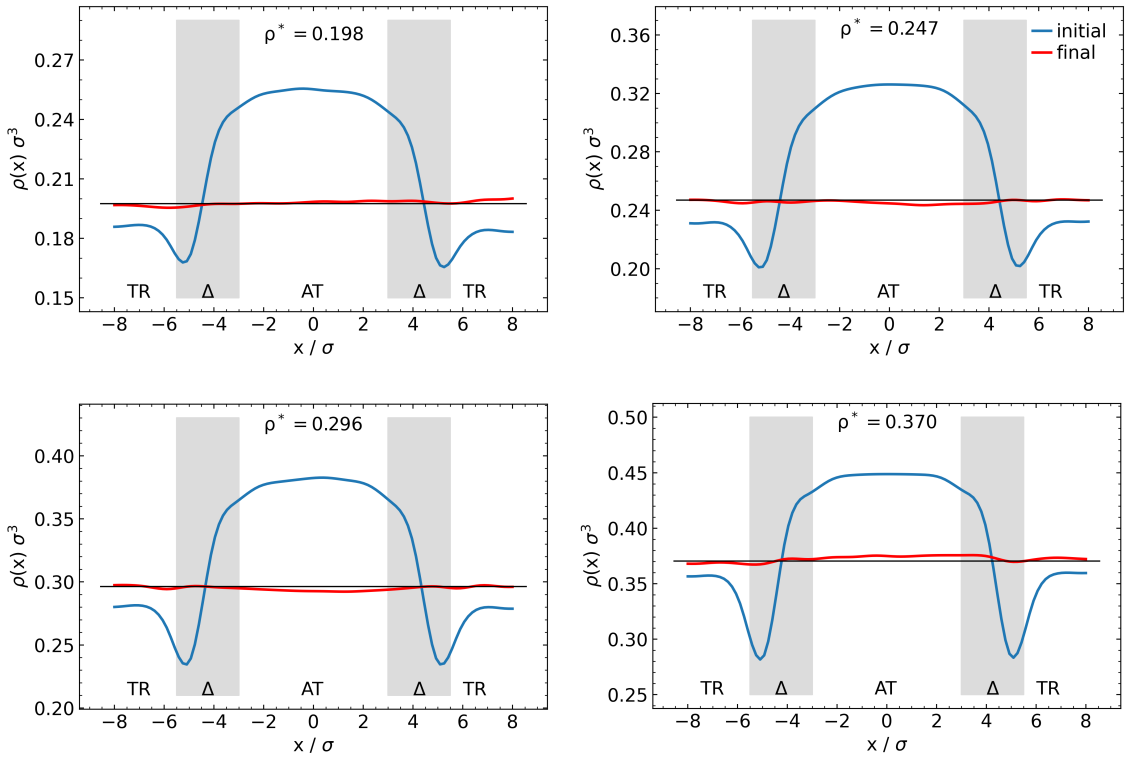


Figure 3.3: Density profiles $\rho(x)$ across the AdResS set-up along the direction of change of resolution, which is chosen as x-axis. Lines show the equilibrium profiles generated at the initial and final steps of the iterative calculation of the thermodynamic force $F_{th}(x)$. The initial choice $F_{th}^{(0)}(x) = 0$ leads to considerable variations in the density (blue), which are forced to a flat profile (red) within a tolerance of 2% relative to the constant equilibrium profile $\rho(x) = \rho^*$ (black) by application of the finally obtained $F_{th}(x)$ (Fig.3.2). The panels show data for Lennard-Jones fluids at the same temperature $T^* = 1.5$ and at four reduced densities as indicated. The transition regions are marked by gray shadings.

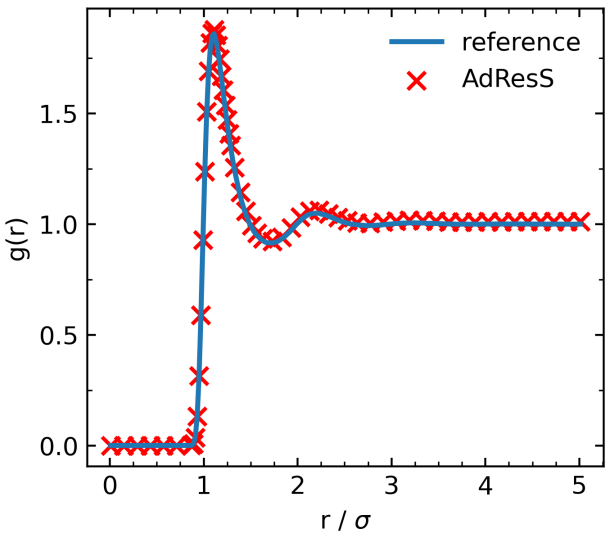


Figure 3.4: Radial distribution function $g(r)$ obtained from the AT region of the AdResS set-up (red symbols) and the corresponding subvolume of the fully atomistic reference (blue line). Data for a Lennard-Jones fluid at temperature $T^* = 1.5$ and number density $\rho^* = 0.37$, using 15 k particles in total.

the density. Note that its sign is opposite to the case of liquid water at room conditions [48]. The physical action of the potential well in $\phi_{th}(x)$ is that tracer particles are pulled into the denser fluid in the Δ region, whereas LJ particles are kept from escaping to the TR region. Effectively, it yields a flat density profile at the equilibrium density ρ^* of the corresponding LJ fluid, i.e., the AdResS set-up reproduces the density of a fully atomistic reference simulation within the prescribed tolerance (Fig.3.3). In the absence of the thermodynamic force, $F_{th}(x) = 0$, the AT and TR regions are unbalanced, generating an excess of particles on one side of the AdResS interface and a depletion on the other. In the specific examples, the density in the center of the AT region, denoted by ρ_0, is increased by 20–30%, which is compensated by a diminution of the amount of tracer particles.

As further checks that the AdResS set-up reproduces the structural and statistical characteristics of the fully atomistic simulation, we compared the radial distribution function $g(r)$ from both approaches, which yield a perfect match (data for $\rho^* = 0.37$ are shown in Fig.3.4). Second, we tested the permeability of the AT/Δ boundary by inspecting the probability distribution $p(N)$ for finding N particles in the AT region and in the corresponding subvolume of the fully atomistic simulation (Fig.3.5). Both distributions superpose closely and resemble a Gaussian; the small shift of the peak positions is related to the allowed tolerance on $\rho(x)$ in the computation of the thermodynamic force. For the density $\rho^*_{at} = 0.37$, we obtained mean values $\langle N \rangle = 2000$ and 2024 for the reference and for AdResS, respectively. Similarly, the standard deviations $\mathrm{std}(N) = 40.6$ and 40.1, being a measure of the compressibility, differ by only 1.3%. We conclude that the AT region of the AdResS set-up used here is a good representation of an open subvolume of a fully atomistic simulation.

3.4.1 The capped energy is negligible

The force capping acting in the Δ region takes care of the divergent interaction potentials, which is technically needed due to the sudden introduction of new interactions upon tracer particles

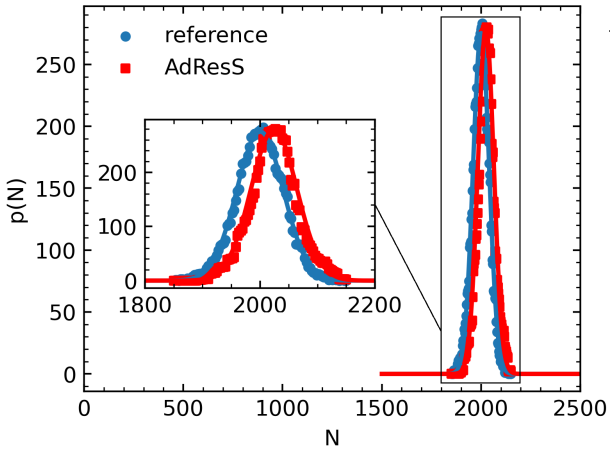


Figure 3.5: Probability distribution $p(N)$ of finding N particles in the region of interest (AT), which is an open system. Comparison of results from the AdResS set-up (red squares) and the fully atomistic reference simulation (blue discs) for a Lennard-Jones fluid at temperature $T^* = 1.5$ and number density $\rho^* = 0.37$. Solid lines are fits to a Gaussian distribution. The inset shows a close-up of the sharp peak seen in the main panel.

entering the atomistically resolved region. Given a certain configuration of molecules in the $AT \cup \Delta$ region, the force capping would renormalize the interaction energy of two molecules, located at the very interface between the Δ region and the tracer region, which have a distance that cannot occur in a fully atomistic simulation. However, this term is negligible compared to the other contributions as evidenced numerically for the LJ fluid at the density $\rho^* = 0.37$, which exhibits the highest frequency of force capping incidences in this study (Fig.3.6). The number of incidences of force cappings rarely exceeds a value of 20 in each MD integration step, which is three orders of magnitude smaller than the total number of pair interactions in the Δ region, estimated to 2×10^4 based on the particle density and the radial distribution function (Fig.3.4). Furthermore, the capping is equivalent to a global modification of the highly repulsive part of the interaction potentials, which has marginal repercussions on the physical properties of the fluid; specifically for the LJ potential and the choice for $F_{cap} = 500\sigma/\epsilon$ used here, the capping corresponds to a modification of the potential for distances shorter than $r_{cap} \approx 0.82\,\sigma$ or potential energies $U(r) \gtrsim 28\,\epsilon$.

3.5 Numerical experiments

Numerical experiments to test Eq.3.21 are performed by molecular dynamics simulations of Lennard-Jones (LJ) fluids for a range of densities so that we gather information for different thermodynamic state points. An additional simulation of liquid water has been carried on to check the applicability of Eq.3.21 for a system with chemically structured molecules, where the passage from the tracer region to the Δ region implies the drastic reintroduction of molecular (atomistic) degrees of freedom. Moreover, liquid water is one of the most relevant examples in molecular simulation and AdResS has been shown to handle such systems in a very satisfactory way (see e.g. Ref. [48]) thus it is an ideal test bed for Eq.3.21.

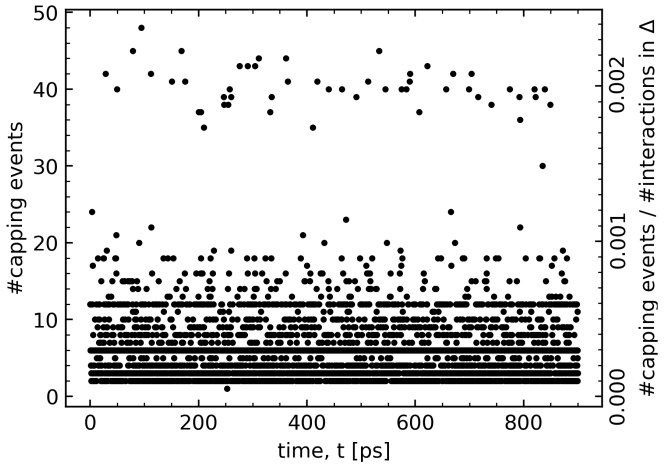


Figure 3.6: Number of incidences of force capping per MD integration step, relative to the total number of pair interactions in the transition region Δ as a function of time, the latter number was estimated to 2×10^4 for the LJ fluid at the density $\rho^* = 0.37$.

3.5.1 Numerical protocol for the calculation of $\Delta\mu$

The total chemical potential of a liquid can be separated into the kinetic and potential contributions $\mu = \mu^{\mathrm{id}} + \mu^{\mathrm{ex}}$. In this relation, μ^{id} originates from the probability distribution of the momenta only, $\propto \exp(-\beta \sum_{i=1}^{N} \vec{p}_i^{\,2}/2m)$, thus it is equivalent to the chemical potential of an ideal gas at the given (uniform) particle density $\rho = N/V$:

$$\mu^{\mathrm{id}} = k_B T \ln(\rho \Lambda^3) \tag{3.24}$$

with the thermal wavelength $\Lambda = h/\sqrt{2\pi m k_B T}$. The contribution μ^{ex} is called excess chemical potential and originates from the position-dependent part of the N-particle phase space density $\propto \mathrm{e}^{-\beta U(x_N)}$ (Ref. [66]). According to the above separation of the chemical potential, Eq.3.21 is rearranged to:

$$\Delta\mu = \mu_r^{\mathrm{ex}} - \mu_0^{\mathrm{ex}} + \gamma^{\mathrm{id}} - \omega_\Delta + \omega_r \tag{3.25}$$

with $\gamma^{\mathrm{id}} = \mu_r^{\mathrm{id}} - \mu_0^{\mathrm{id}} = k_B T \log(\rho_r/\rho_0)$, where ρ_0 is the particle density in the AT region in the initial iteration of the thermodynamic force calculation, that is when no corrections are added to the potential yielding the unbalanced density (Fig.3.3).

All the ingredients needed to explicitly calculate $\Delta\mu$, i.e. the unknown perturbation in the chemical potential generated by the thermodynamic force, are now available. First, μ_r^{ex} can be calculated by, e.g., Widom's test particle insertion [199] in the fully atomistic simulation of the reference system. Second, μ_0^{ex} instead is the chemical potential the system would have in the AT region if AdResS runs without the thermodynamic force in the transition region Δ. It can be determined from also Widom's test particle insertion either in the AT region of the initial AdResS set-up or in a standard MD simulation at the density ρ_0, that is the density of the AT region in the AdResS set-up without any correction measures. The latter occurs as the density in the first iteration run for finding the thermodynamic force (see Fig.3.3), since we are assuming that the AT region is infinitely large.

3.5.2 Numerical results

All terms contributing to Eq.3.25 can be determined from the fully atomistic simulation of reference (μ_r^{ex} and ω_r), the AdResS simulation (γ^{id} and ω_Δ), and a mix of both simulations (μ_0^{ex}) in a straightforward manner as described. Here, after validating the case studies for the AdResS

ρ^*	μ_r^{ex}	μ_0^{ex}	γ^{id}	ω_Δ	ω_r	$\Delta\mu$	μ_r^{ex} acc. Eq.3.27
0.198	-1.255(2)	-1.532(3)	-0.385(3)	0.125(2)	0.222(3)	-0.011(11)	-1.244(11)
0.247	-1.487(3)	-1.789(4)	-0.411(3)	0.160(3)	0.256(3)	-0.013(16)	-1.474(13)
0.296	-1.686(4)	-1.938(5)	-0.384(3)	0.192(3)	0.306(3)	-0.018(18)	-1.668(14)
0.370	-1.912(5)	-2.032(5)	-0.268(3)	0.233(3)	0.365(3)	-0.016(19)	-1.896(14)
water	-24.8(1)	-21.9(1)	-0.203(3)	0.210(4)	3.1(1)	-0.2(3)	-24.6(3)

Table 3.1: Breakdown of the chemical potential relation into AdResS-related contributions [Eq.3.25] for the investigated Lennard-Jones fluids at temperature $T^* = 1.5$ and number density as given in the first column. The values for the density ρ_0 (entering μ_0^{ex} and γ^{id}) and the free energy contribution ω_Δ related to the thermodynamic force were obtained from AdResS simulations (columns 3 to 5), whereas the results for μ_0^{ex} and ω_r (columns 3 and 6) as well as for the reference value for μ_r^{ex} (second column) stem from fully atomistic simulations. The values for μ_r^{ex} in column 8 were calculated according to Eq.3.27. Chemical potentials and free energies are given in units of ϵ for the LJ fluids and units of $\mathrm{kJ\,mol^{-1}}$ for water. Numbers in parentheses give the uncertainty in the last digit(s).

simulation, i.e. investigating their capability for preserving structural and statistical properties of the fluids compared to the reference set-up, we have tested our derivations for four different LJ fluids at different state points (different densities).

Simulation results for each contribution to the excess chemical potential relation stated in Eq.3.25 are reported in Tab.3.1. $\Delta\mu$ can be interpreted as the difference between the chemical potential of the fluid within a fully atomistic simulation and the one computed from an AdResS simulation. Interestingly, one would expect that $\Delta\mu \to 0$ as $|AT| \to \infty$ because the atomistic region would behave as a closed, infinite fully atomistic system with $\mu_0 \to \mu_r$, $\omega_\Delta \to 0$ and $\omega_r \to 0$. The numerical results of the current simulations are for finite systems with sizes typical of routine AdResS simulations and they show that $\Delta\mu \approx 0$ even when ω_Δ and ω_r are not negligible. This is an interesting result because it allows us to state that the numerical experiments over different densities suggest an effective formula:

$$\mu_r - \mu_0 = \omega_\Delta - \omega_r \,, \tag{3.26}$$

or,

$$\mu_r^{\mathrm{ex}} - \mu_0^{\mathrm{ex}} = -\gamma^{\mathrm{id}} + \omega_\Delta - \omega_r \,. \tag{3.27}$$

The relative deviation of μ_r^{ex} calculated from Eq.3.27 w.r.t. the reference value from fully atomistic simulations, is near or below 1% in all cases. In particular, the two values coincide within their specified statistical uncertainties (Fig.3.7). Yet, we note that the reference values systematically lie slightly below the AdResS values, which has a possible source in the neglected contribution due to the capping of unduly large interparticle forces.

We also tested Eq.3.25 for liquid water as a system routinely simulated with AdResS for biological systems such as membranes [198]. This is a far more complex liquid compared to the Lennard-Jones systems and the simulation set-up is far from mimicking the thermodynamic limit, yet we find that the equation still holds. In this case, the dominant correction is ω_r, while $\Delta\mu$ is comparable with γ^{id} and ω_Δ and these terms contribute with less than 1% to the sum in Eq.3.27. The possibility to reconstruct the excess chemical potential μ_r^{ex} with high accuracy from an AdResS simulation provides a first-principles confirmation of the physical consistency of AdResS as an open system.

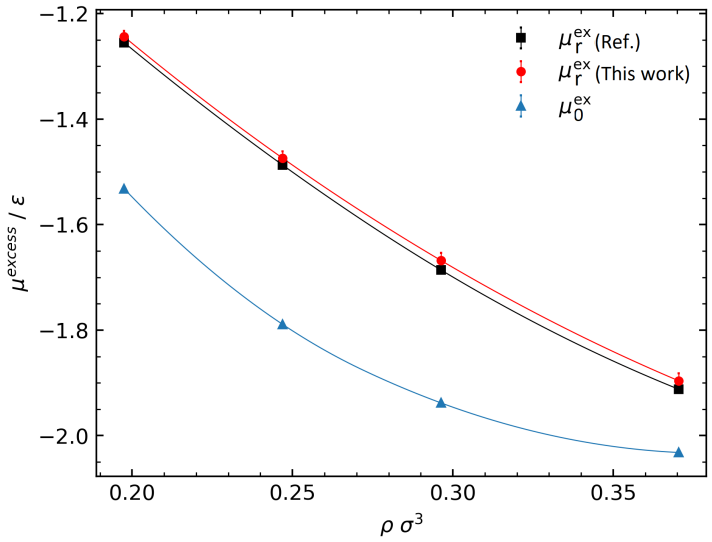


Figure 3.7: Excess chemical potential of LJ fluids at temperature $T^* = 1.5$ as function of the number density ρ. Values obtained from AdResS simulations (red circles) via Eq.3.27 are compared to reference data from Widom's test particle insertion in standard MD simulations (black squares). The quantity μ_0^{ex} (blue triangles) refers to the AT region of the AdResS set-up with the thermodynamic force switched off, which results in the modified density ρ_0 (see Fig.3.3). The data points correspond to columns 1, 2, 3, and 8 of Tab.3.1.

3.6 Conclusions

We have analyzed the coupling region of the AdResS set-up from the microscopic point of view. We have shown the possibility of explicitly writing the grand potential of the atomistically resolved region in terms of quantities that can be determined from simulations. In particular, we have found the relation Eq.3.21 between the chemical potential of AdResS and the chemical potential of its reference fully atomistic simulation. The derivation is done under the ideal condition of the thermodynamic limit for the atomistically resolved region, with the coupling conditions considered as small surface effects. The obtained thermodynamic relation was then tested in several numerical experiments, they show that its actual range of validity extends to finite systems with sizes typical of standard AdResS simulations. Accepting that $\Delta\mu = 0$ holds also for a finite (yet not too small) $AT \cup \Delta$ region implies that the equilibrated AdResS (i.e., with F_{th} switched on) and the subsystem S of the fully atomistic reference simulation are open systems at different chemical potentials, μ_0 and μ_r, that otherwise exhibit the same physical properties.

The numerical confirmation of the validity of the thermodynamic relations in AdResS provides a statistical mechanics validation of the method as a reasonable numerical approximation of an open system embedded in a reservoir of particles and energy. In conclusion, we have shown that although the abrupt coupling may suggest that a high degree of seemingly artificial conditions are required for the technique to work properly, in effect the numerical conditions are consistent with the statistical mechanics' principles of an open system.

Chapter 4

Pressure and Grand Potential balance in Adaptive Resolution Simulation

Abstract

In the previous chapter [73], we have identified a precise relation between the chemical potential of a fully atomistic simulation and the simulation of an open system in the adaptive resolution method (AdResS). The starting point was the equivalence derived from the statistical partition functions between the grand potentials, Ω, of the open system and the equivalent subregion in the fully atomistic simulation of reference. In this chapter, instead, we treat the identity for the grand potential based on the thermodynamic relation $\Omega = -pV$ and investigate the behaviour of the pressure in the coupling region of the adaptive resolution method (AdResS). We confirm the physical consistency of the method for determining the chemical potential described by the previous chapter and strengthen it further by identifying a clear numerical relation between the potential that couples the open system to the reservoir on the one hand and the local pressure of the reference fully atomistic system on the other hand. Such a relation is of crucial importance in the perspective of coupling the AdResS method for open system to the continuum hydrodynamic regime.

4.1 Introduction

In the previous work [73], we have investigated the microscopic origin of several thermodynamic quantities at the coupling boundary of a system of Lennard-Jones (LJ) particles with a reservoir of non-interacting tracers. The adaptive resolution technique (AdResS) [30, 49, 174] was employed, as a technical set-up, for running the numerical simulations. The work aimed to show that the AdResS scheme translates, accurately and efficiently, the statistical mechanics principles of open systems into a convenient numerical simulation tool. A pictorial representation of the AdResS set-up is reported in Fig.4.1 and the relevant details of the method will be reported later on in a specific section. For the current discussion, it is sufficient to consider that the technique allows for the exchange of particles between the atomistically resolved region (AT) and the reservoir region (TR) where particles are not interacting. The exchange occurs through an interface region (Δ) within which a prescribed external potential (potential of the thermodynamic force) and a thermostat enforce the equilibration of the atomistic region to the same thermodynamic state as that of the fully atomistic simulation of reference. The study consisted in comparing the thermodynamic properties of a subsystem of a fully atomistic simulation with those of the equivalent atomistically resolved region in the AdResS set-up, and it concludes the physical consistency of the AdResS scheme with the statistical mechanics model of an open system.

The starting assumption was that the subregion of the fully atomistic simulation(equivalent to the AT region) and the AT region in AdResS are both open regions whose particles follow the Grand Canonical distribution. Since AdResS aims to reproduce the same statistical and thermodynamic properties of the target fully atomistic simulation in the AT region, the equivalence of the particle statistical distributions implies some direct relation between the chemical potentials of the two simulations. Indeed, the study led to the conclusion that the coupling strategy, through the external potential, balances the difference in chemical potential between the fully atomistic and an AdResS simulation without the thermodynamic force. This result justifies, under the Grand Canonical assumption, the role of AdResS as a technical tool to simulate open systems in a physically consistent manner. Although it has been numerically verified that AdResS follows the Grand Canonical distribution (Grand Canonical AdResS) [6, 174, 195], there may be alternative approaches which, without explicitly requiring the Grand Canonical hypothesis, can complement that of Ref.73 and thus further strengthen the role of AdResS as a tool which is consistent with the physical principles of open systems.

In this context, this chapter aims to explore an approach which is complementary to those already considered and involves a thermodynamic quantity, the pressure, without requesting the Grand Canonical hypothesis. The pressure is, with temperature and density, a key thermodynamic quantity in molecular simulation. We show in detail that the coupling strategy of AdResS, through the introduction of an external potential, correctly balances the difference in pressure in the adaptive set-up w.r.t, the fully atomistic value of reference.

4.2 The AdResS Method: Basics

In the AdResS set-up, the simulation box is divided into three regions: the AT region at atomistic resolution (region of physical interest), the coupling region Δ, where particles have an atomistic resolution, but with additional/artificial coupling features to the large reservoir, and TR, the reservoir of non-interacting point-particles known as tracers (see Fig.4.1). Particles can freely move from one region to the other and automatically change their molecular resolution according to the resolution that characterizes the region in which they are instantaneously located.

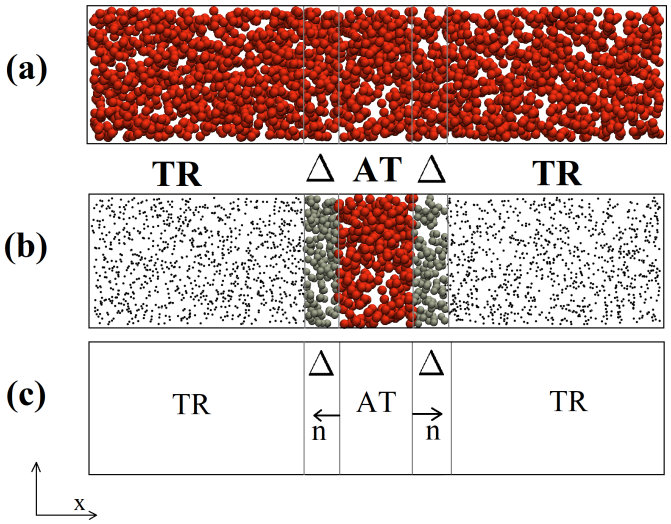


Figure 4.1: Comparison of the AdResS and reference set-ups. Panel (a) shows the reference full atomistic set-up with high resolution through the whole domain. Panel (b) represents the AdResS set-up with the atomistic region AT, the interface region Δ, and the TR reservoir region; here, the i_{th} particle interacts with the j_{th} particle through a pair potential $U_{ij} = U(\vec{q}_j - \vec{q}_i)$. The one-body thermodynamic force, $F_{th}(\vec{q}_i)$, acts on all particles in the $\Delta \cup TR$ region and enforces the desired thermodynamic equilibrium in the region of interest. Panel (c) indicates the direction $\vec{n}$ perpendicular to the coupling surface at the Δ/TR interface along which acts the thermodynamic force.

In terms of interactions, molecules of the AT region have standard atomistic two-body potentials among themselves and with molecules in Δ, and vice versa, but there is no direct interaction with the tracers in TR. Tracers and particles in Δ experience an additional one-body force, called thermodynamic force, along the direction $\vec{n}$ perpendicular to the coupling surface at the Δ/TR interface, $\vec{F}_{th}(\vec{q}) = F_{th}(\vec{q})\vec{n}$ for positions $\vec{q}$. This force, together with the action of a thermostat in these regions, implements an effective coupling to the rest of the universe outside the AT region. The total interaction potential reads: $U_{tot} = U_{tot}^{AT} + \sum_{\vec{q}_j \in \Delta \cup TR} \phi_{th}(\vec{q}_j)$ with the potential $\phi_{th}(\vec{q})$ such that $\vec{F}_{th}(\vec{q}) = -\nabla\phi_{th}(\vec{q})$ and $\phi_{th}(\vec{q}) = 0$ in the AT region, $\vec{q} \in AT$. For the discussion here, it suffices to know that the thermodynamic force is calculated such that the particle density in the atomistic region is equal to a prescribed value of reference. It has been shown [68, 73, 152, 195] that the constraint on the density profile, through the thermodynamic force in $\Delta \cup TR$, induces the thermodynamic equilibrium of the atomistic region w.r.t. conditions of reference of a fully atomistic simulation.

4.3 Pressure calculation in an open system

In the previous chapter, the starting point was the microscopic definition of the AT $\cup$ Δ region in AdResS as an open system with Grand Potential Ω, embedded in the TR region as a reservoir [73]. This Grand Potential is defined in microscopic terms under the hypothesis that AT $\cup$ Δ is characterized by a grand canonical partition function for the particles: $\Omega = -k_B T \ln\left(\sum_{N=0}^{\infty} {}^{\beta\mu N} Q_N\right)$, where μ, T, and Q_N are the chemical potential at equilibrium, the temperature, and the canonical partition function (at a given particle number N), respectively,

and $\beta = 1/k_BT$ with k_B being the Boltzmann's constant. Since we compare a fully atomistic set-up with the AdResS set-up and they are partitioned in space in the same way, in essence, the quantity to check is the pressure. The virial equation (4.1) defines the pressure as the sum of its particles' kinetic and interparticle force contributions in a homogeneous system with no external forces [85, 86, 167]. For a system of volume V, this relation can be expressed as [10, 81]

$$p = \frac{1}{3V}\left(\sum_i m_i \boldsymbol{v}_i \cdot \boldsymbol{v}_i + \sum_i \boldsymbol{r}_i \cdot \boldsymbol{f}_i\right), \tag{4.1}$$

where m_i, $\boldsymbol{r}_i$, and $\boldsymbol{v}_i$ are each particle's mass, position, and velocity respectively, and $\boldsymbol{f}_i$ is the total interparticle force acting on each particle. While Eq.4.1 can be applied to the fully atomistic system, the calculation of the pressure in AdResS is not straightforward. The reason lies in the abrupt change of resolution with sharp boundary effects and the action of an external force field (thermodynamic force).

There are several methods for deriving Eq.4.1, they all use the idea of isotropy and/or homogeneity of the system in their derivations and directly consider the scalar pressure, instead of the stress tensor. The stress tensor should instead be used for inhomogeneous and anisotropic systems [187]. In general, there are two methods for deriving the pressure: (i) through the thermodynamic relation $p = -\partial F/\partial V|_T = k_BT(\frac{\partial}{\partial V} log Q_N(V,T))_T$, with F being the Helmholtz free energy and Q_N being the canonical partition function or its equivalent [36]; (ii) a direct mechanical calculation by summing up the kinetic (momentum carried by particles) and potential (interparticle force $\boldsymbol{f}_{ij}$ acting between pairs of particles) contributions to the pressure (see Fig.4.2). However, while the use of the thermodynamic relation is possible only in the limit of thermodynamic equilibrium for homogeneous systems, the second method can instead be applied in AdResS, using particle trajectories, to calculate the stresses. In inhomogeneous and anisotropic systems, the stress tensor is position and direction-dependent. The most appropriate formal treatment, in this case, consists of writing the inhomogeneity in terms of the stress tensor [89] at the position, $\mathbf{r}$, in space, $\boldsymbol{P}(\boldsymbol{r})$, which can be split into kinetic and potential contributions [187]:

$$\boldsymbol{P}(\boldsymbol{r}) = \boldsymbol{P}^K(\boldsymbol{r}) + \boldsymbol{P}^U(\boldsymbol{r}) \tag{4.2}$$

with components

$$\boldsymbol{P} = \begin{pmatrix} \sigma_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_{zz} \end{pmatrix} \tag{4.3}$$

where the σ_{ii} and τ_{ij} are the normal and shear components of the tensor respectively.

The stress tensor can be defined by the interparticle force acting across a moving test surface along with the simulation domain (see Fig.4.2). The kinetic contribution accounts for the particles' momentum while they cross the test surface and as it depends on each particle's location, it is a single-particle property and can be localized in space. The potential term corresponds to the interaction forces due to the interaction of particles on opposite sides of the surface. This part is not local since it depends on the location of both particles [187] (see also Fig.4.2).

Irving and Kirkwood [99] introduced a new approach for the calculation of the pressure and stress tensor by starting from a statistical mechanical derivation of the equations of hydrodynamics and making a particular selection for the particles that contribute to the inter-particle force. Accordingly, only pairs of particles which satisfy the condition that the line connecting their centers of mass passes through the test surface contribute to the local force. With this definition, they obtained a localized form for the potential contribution of the pressure. For a system with planar symmetry and no-flow condition (like in the AdResS set-up in References [49, 73]), all non-diagonal elements of the stress tensor (Eq.4.3) must be zero on average as

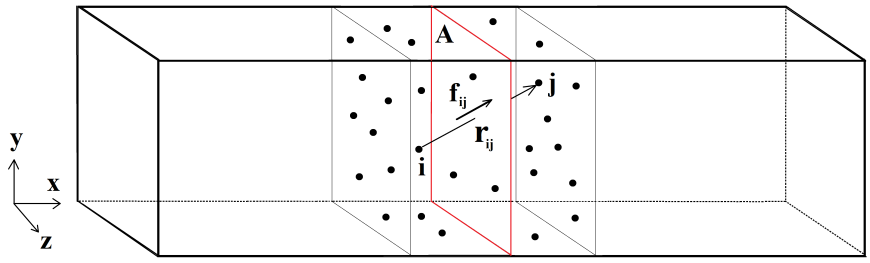


Figure 4.2: Pressure calculation in a volume element in the simulation domain of a molecular system according to the idea of moving test planes. The red surface is located in the middle of the volume element and the stress tensor elements can be calculated by adding the pressure resulting from the interaction force between those particles on the opposite sides of the plane to the kinetic contribution of all particles within the volume element.

there is no shear stress in equilibrium due to the lack of velocity gradient and motion between hypothetical liquid layers [25]. Moreover, the change of resolution is happening along, say, the x-axis, so the normal component of the stress tensor will be $P_N(\boldsymbol{r}) = \sigma_{xx}(\boldsymbol{r})$ and the tangential components are identical due to the symmetry $P_T(\boldsymbol{r}) = \sigma_{yy}(\boldsymbol{r}) = \sigma_{zz}(\boldsymbol{r})$. Finally, the scalar pressure is defined as $p = (\sigma_{xx} + \sigma_{yy} + \sigma_{zz})/3 = (P_N + 2P_T)/3$ [25, 183]. In this context, Irving and Kirkwood proposed the following expressions for the normal and transverse components of the stress tensor [99, 163, 194]:

$$P_N(x) = \rho(x)k_BT - \frac{1}{2A}\langle\sum_{i\neq j}\frac{|x_{ij}|}{r_{ij}}U'(r_{ij})\Theta(\frac{x-x_i}{x_{ij}})\Theta(\frac{x_j-x}{x_{ij}})\rangle, \tag{4.4}$$

$$P_T(x) = \rho(x)k_BT - \frac{1}{4A}\langle\sum_{i\neq j}\frac{y_{ij}^2+z_{ij}^2}{r_{ij}}\frac{U'(r_{ij})}{|x_{ij}|}\Theta(\frac{x-x_i}{x_{ij}})\Theta(\frac{x_j-x}{x_{ij}})\rangle \tag{4.5}$$

where Θ is the Heaviside step function. The first term on the right-hand side of Eq.4.4 and Eq.4.5 is the kinetic contribution which can be calculated by taking into account the local temperature in the small volume element around the test plane and is equivalent to the kinetic contribution in the virial equation (Eq.4.1), i.e. $\frac{1}{3V}\langle\sum_i m_i v_i^2\rangle$. The other terms in Eq.4.4 and Eq.4.5 involve the interaction of pairs of particles and express the fact that when two particles i and j are located on the same side of the surface, the potential contribution of the pressure will be zero and when they are on the opposite sides, the corresponding interparticle force will be considered in the related stress tensor component. The average $\langle ... \rangle$ is the time average that means after reaching equilibrium, the sliding surface moves into the simulations domain at different times and an average value for the normal and transverse components of the stress tensor is calculated at different locations into the simulation domain.

We will use Eq.4.4 and Eq.4.5 to determine the pressure in AdResS and compare the results with those obtained in a fully atomistic simulation by the same relations and also using the virial relation for the homogeneous system (Eq.4.1). The comparison shows the consistency of AdResS as a tool to simulate open systems.

4.4 Numerical Results

In this section, we report the technical details and the numerical results of the simulations. In the following, the AdResS set-up and its technical details are presented, and then the pressure in the domain is calculated based on the discussed methodology. Finally, a relation between

the pressure function and the thermodynamic force needed to balance that pressure difference is shown.

4.4.1 Technical details of the simulation

We use the same technical set-up of Ref. [73] explained in the previous chapter. Below, we briefly summarize the key aspects and invite the interested reader to consult our previous work for specific details. We have considered four Lennard-Jones liquid systems each at a different thermodynamic state point, namely: number densities $\rho^* := \rho\sigma^3 \approx 0.20$, 0.25, 0.30, and 0.37, corresponding to particle numbers $N = 8k$, $10k$, $12k$, and $15k$ at the reduced temperature of $T^* := k_B T/\epsilon = 1.5$ which is well above the liquid-vapour critical point.

A fully atomistic simulation of reference for all test cases has been performed, followed by an adaptive resolution simulation for each state point. In the equilibration run, the corresponding thermodynamic force was determined by the iterative formula [68]:

$$F_{th}^{k+1}(x) = F_{th}^{k}(x) - c(\frac{m}{\kappa_T \rho_0^2})\nabla\rho^k(x), \tag{4.6}$$

with m being the particle mass, κ_T the thermal compressibility, ρ_0 the target density, and c a prefactor for controlling the convergence rate. According to Ref. [68], the above-mentioned external force is derived in such a way that compensates the pressure difference generated drift force resulting from the addition/change of resolution compared to the reference fully atomistic set-up, i.e. $\vec{F}_{th}(x) = \frac{m}{\rho_0}\nabla p(x)$ with $p(x)$ being the pressure of the system as a function of position. In addition, the required external potential relates to the calculated thermodynamic force by $\vec{F}_{th}(x) = -\nabla\phi_{th}(x)$; thus, the added external potential to the system ($\phi_{th}(x)$) is expected to compensate the needed energy to keep the pressure of the system unchanged while progressing toward a multi-resolution domain. This property has been investigated later (see Fig.4.8).

The density profile for each case is shown in Fig.4.3. The AdResS set-up for each case was then validated in the production run with the comparison to the corresponding fully atomistic case of the calculated radial distribution function, $g(r)$, and the probability of finding N particles $p(N)$ in the region of interest (AT) (see Fig.4.4 and Fig.4.5). The criteria of validation of AdResS used above have been shown to ensure the numerical consistency of AdResS as a tool to properly simulate basic structural and statistical properties of the AT region (i.e. the region of interest) [49, 58, 111, 197]. Once the numerical set-up of AdResS has been validated, one can proceed with the calculation of the pressure using the formulas discussed in the previous section. The corresponding results are reported in the next section.

4.4.2 Numerical calculations for the pressure

At first, as a traditional way to calculate the pressure in molecular systems, we have computed the pressure in the fully atomistic simulation of reference, $p_{ref.}$, considering it a homogeneous system and thus using the virial relation (Eq.4.1). The results are shown in Tab.4.1. Next, we have applied the test planes approach introduced above to the fully atomistic system as well. We considered a test plane moving into the simulation domain of the system and compute both potential and kinetic contributions of the normal and tangential components of the stress tensor through a spatial and temporal average (P_N^{at} and P_T^{at} in Tab.4.1). They have been calculated

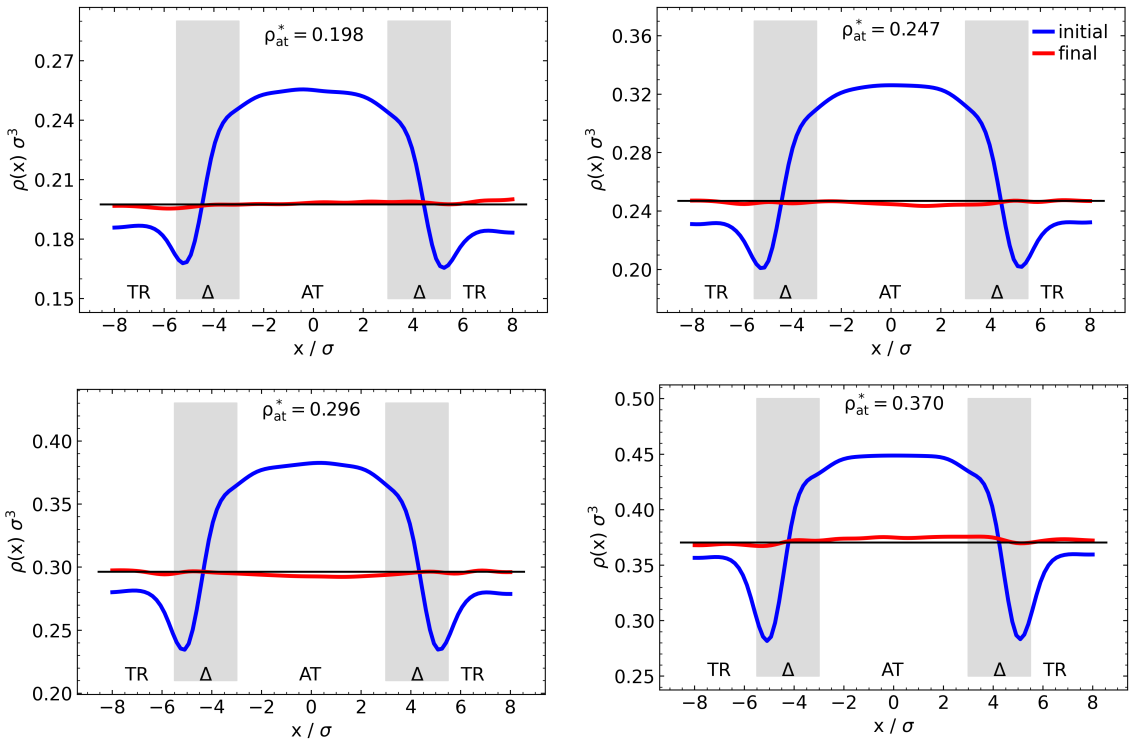


Figure 4.3: Density profiles $\rho(x)$ along the direction of change of resolution for four different cases at reduced densities indicated in the figures and reduced temperature of $T^* = 1.5$. The blue and red curves indicate the density profile in the AdResS set-up before and after the application of thermodynamic force respectively. The proper thermodynamic force is found through an iterative procedure (Eq.4.6) by an initial choice of $F^{(0)}_{th}(x) = 0$ (corresponding to the blue line) and continued till reaching a satisfactory deviation of 2% (corresponding to the red line) from the target constant density (indicated by the black line). The transition regions are marked by grey shadings.

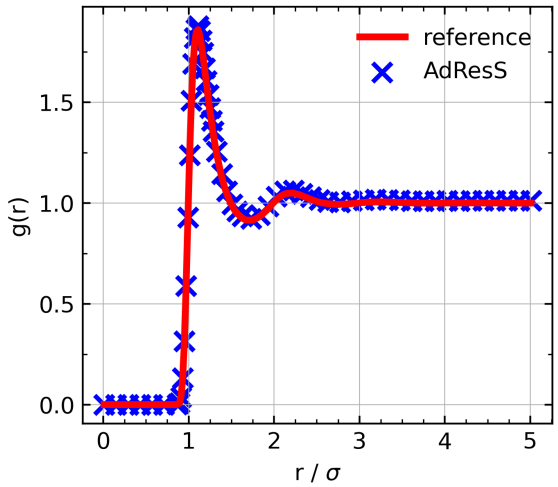


Figure 4.4: Radial distribution function ($g(r)$) for fully atomistic simulation of reference (red line) and AdResS simulation (blue markers). These data correspond to the LJ fluid at the reduced density of $\rho^* = 0.198$ and reduced temperature of $T^* = 1.5$. The same level of agreement was found for the other thermodynamic state points treated and for this reason they are not shown.

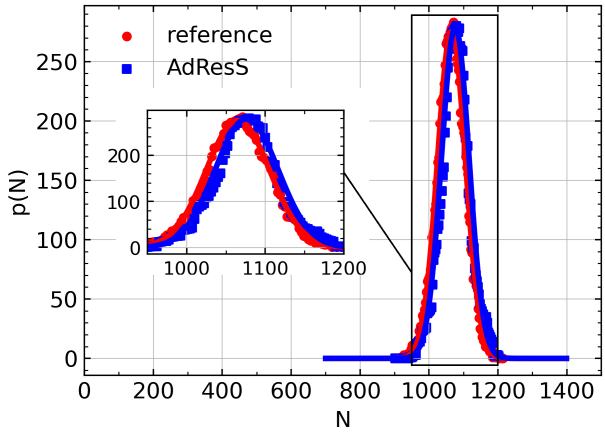


Figure 4.5: Probability of finding N particles in the high-resolution region (AT region) for fully atomistic simulation of reference (red) and AdResS (blue) at the reduced density of $\rho^* = 0.198$ and reduced temperature of $T^* = 1.5$. For each case, a Gaussian distribution is fitted to the calculated data and the close-up of the data around the average particle number in AT region is shown in the inset. The same level of agreement was found for the other thermodynamic state points treated and for this reason they are not shown.

by using trajectory data of particles which are recorded every 10τ during an MD run for the duration of $10^4\tau$ with each time step being equal to 0.002τ. It is noteworthy to mention that we have considered periodic boundary conditions for calculating the interparticle distances in all equations. In addition, only particles within a certain distance from the test planes ($=r_{cut-off}$) have been considered for calculations to implement the effect of cut-off radius, i.e. 2.5σ. Once we have determined the abovementioned quantities for the reference fully atomistic system, we employed the same approach for the AdResS simulation and determined P_N^{ad} and P_T^{ad} (in Tab.4.1).

Table 4.1: Results of pressure calculation based on the plane approach presented in this work. The second column (p_{ref}) is the pressure of the fully atomistic simulation of reference, based on virial relation (Eq.4.1) as a traditional method for calculating pressure in molecular systems. The rest are the scalar pressure (p^{at} and p^{ad}) and stress tensor components (P_N^{at}, P_T^{at}, P_N^{ad}, and P_T^{ad}) in AdResS and fully atomistic simulations which are calculated by Irving-Kirkwood relations (Eq.4.4 and Eq.4.5).

ρ^*	p_{ref}	P_N^{at}	P_N^{ad}	P_T^{at}	P_T^{ad}	p^{at}	p^{ad}
0.198	$0.181_{\pm 0.007}$	$0.183_{\pm 0.007}$	$0.184_{\pm 0.006}$	$0.181_{\pm 0.012}$	$0.183_{\pm 0.015}$	$0.182_{\pm 0.010}$	$0.183_{\pm 0.012}$
0.247	$0.202_{\pm 0.010}$	$0.208_{\pm 0.006}$	$0.207_{\pm 0.007}$	$0.203_{\pm 0.014}$	$0.205_{\pm 0.013}$	$0.205_{\pm 0.011}$	$0.206_{\pm 0.011}$
0.296	$0.220_{\pm 0.013}$	$0.218_{\pm 0.008}$	$0.221_{\pm 0.007}$	$0.224_{\pm 0.015}$	$0.218_{\pm 0.012}$	$0.222_{\pm 0.013}$	$0.219_{\pm 0.010}$
0.370	$0.254_{\pm 0.015}$	$0.251_{\pm 0.010}$	$0.255_{\pm 0.008}$	$0.252_{\pm 0.014}$	$0.256_{\pm 0.014}$	$0.252_{\pm 0.013}$	$0.253_{\pm 0.012}$

As can be seen from Tab.4.1, the method of planes is calculating the pressure satisfactorily. Moreover, the agreement between the values of the fully atomistic simulation and the AdResS simulation in Fig.4.6 confirms, from a straightforward thermodynamic point of view, the equality of the corresponding grand potentials. Thus, the AT region of AdResS is thermodynamically compatible with the equivalent subregion in a fully atomistic simulation.
However, the values calculated of the pressure in Fig.4.6 correspond to the average pressure and the condition of equality of the grand potentials represents only a necessary condition of com-

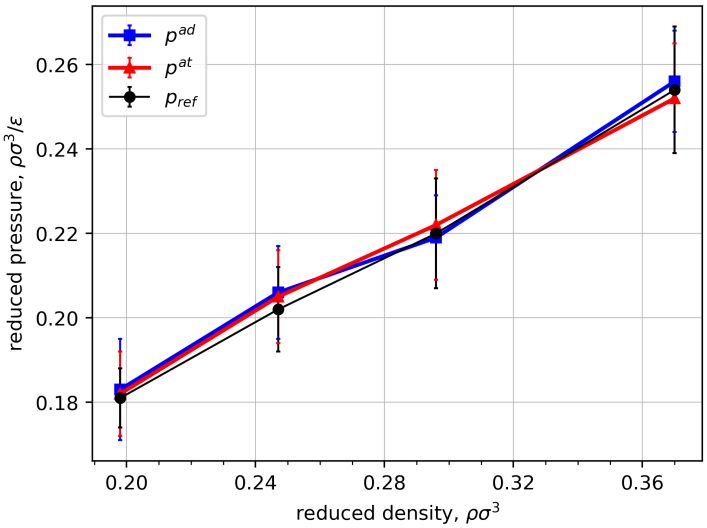


Figure 4.6: The value of scalar pressure in full-atomistic and AdResS simulations at four different thermodynamic state points. These values are calculated based on the virial method for reference set-up (black line) and Irving-Kirkwood relations for reference (red line)) and AdResS (blue line) simulations.

patibility. A more powerful criterion would be a space-dependent check of consistency between the AdResS set-up and the desired thermodynamic equilibrium. This calculation is reported in the section below.

4.4.3 Relation between the potential of thermodynamic force and pressure

One of the key roles of the thermodynamic force is to calibrate the pressure in the region of interest to produce the same grand potential as that of the corresponding fully atomistic simulation of reference. Since the thermodynamic force is applied to the system only in the Δ region, one may see its effect on the pressure as a function of the position along the axis of change of resolution (x). It is possible to calculate the stress tensor components as a function of x in both full-atomistic and AdResS set-ups by using the relations of Irving-Kirkwood (Eq.4.4 and Eq.4.5) for normal and transverse components which both include kinetic and potential contributions of the pressure. The corresponding functions are shown in Fig.4.7.

As we see in Fig.4.7, the pressure in the AT region and in the equivalent subregion of the fully atomistic simulation are pointwise compatible, within the usual numerical fluctuations. Interestingly, despite the close agreement in the AT region, in the Δ region the difference is rather drastic. To see the effect of thermodynamic force and change of resolution on the resulting pressure difference, we plotted the energy corresponding to the pressure difference (by normalizing the pressure with the local density), which can be interpreted as the required energy to keep the pressure of the system unchanged while adding new resolution to the system, on top of the potential of thermodynamic force, $\phi_{th}(\vec{q})$, that is calculated by integrating the required thermodynamic force for each case (see Fig4.8).

A denser liquid with a larger deviation in density profile (see Fig.4.3) and consequently a larger difference in pressure profile (see Fig.4.7) requires a stronger external potential to reproduce the

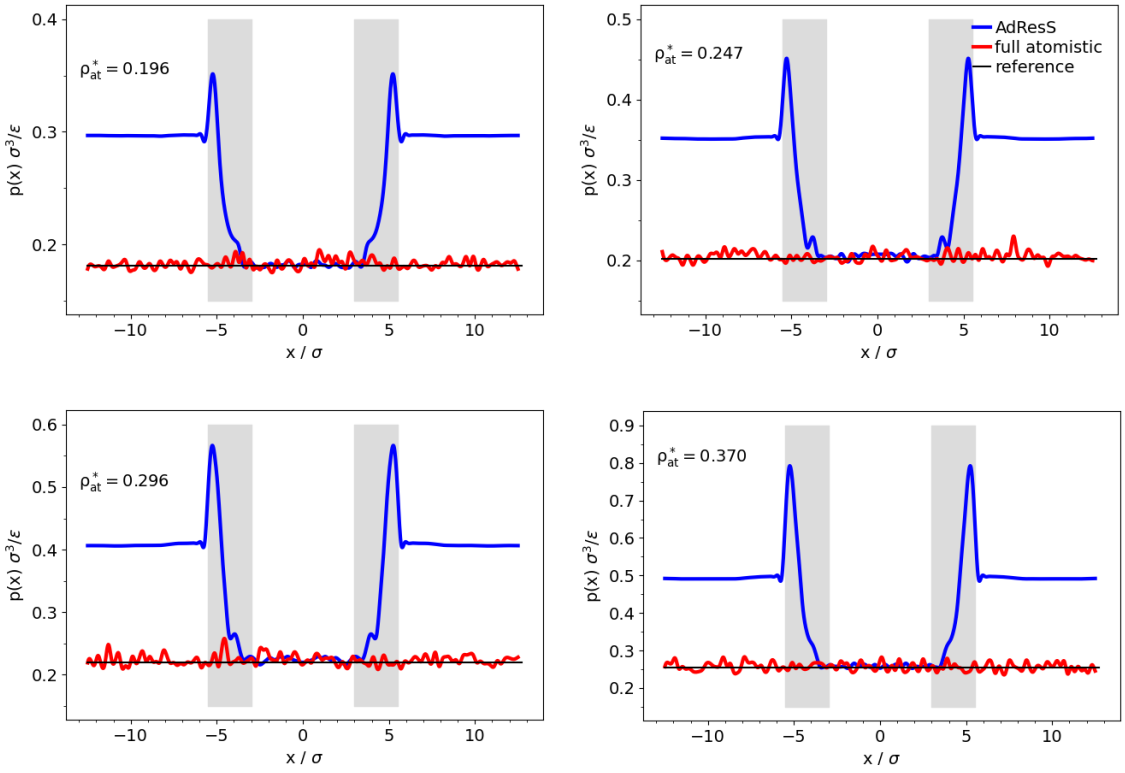


Figure 4.7: The pressure profile for all cases for AdResS and fully atomistic simulation of reference. The black line represents the scalar pressure in the full atomistic simulation of reference whose calculation is based on the virial equation. The red and blue lines represent the pressure in the fully atomistic and the adaptive resolution simulations, respectively. This latter calculation is based on Irving-Kirkwood relations (Eq.4.4 and Eq.4.5). The grey areas show the coupling region Δ and the AT region is located in the middle of the box

same behaviour as the reference set-up and adjust the pressure in the high-resolution region to get the same grand potential. Interestingly, in all cases the energy matches, within its numerical fluctuation (shadowed area), with the curve of the potential of the thermodynamic force. This result is very relevant because it allows the direct pointwise identification of the potential of the thermodynamic force with the energy related to the pressure and thus it assures that the balancing process will always lead to the correct pointwise pressure in the AT region. In turn, such a finding fully complements the results of the previous chapter: *the AT region reproduces the grand potential of the equivalent subregion of the reference simulation either through a microscopic statistical analysis involving directly its partition function or from a straightforward thermodynamic point of view through the calculation of the pressure and its pointwise comparison with the reference system.*

It must be reported that previous work has explored the connection of the pressure with the balancing potential in similar simulation set-ups [61, 153, 154]. An artificial global Hamiltonian was designed and a corresponding semi-empirical statistical ensemble was defined; the ensemble used does not have a well-defined physical meaning, and thus, it does not allow a direct derivation of thermodynamic relations (see detailed discussion in Refs. [30, 44]). The thermodynamic relations proposed in Refs. [61, 153, 154] are rather intuitive and do not offer a clear physical interpretation. In this work, we have gone beyond the artificial global Hamiltonian and defined a physically rigorous Hamiltonian of the open system. The corresponding statistical derivation of its physical quantities is, as consequence, rigorously done in the Grand Canonical ensemble

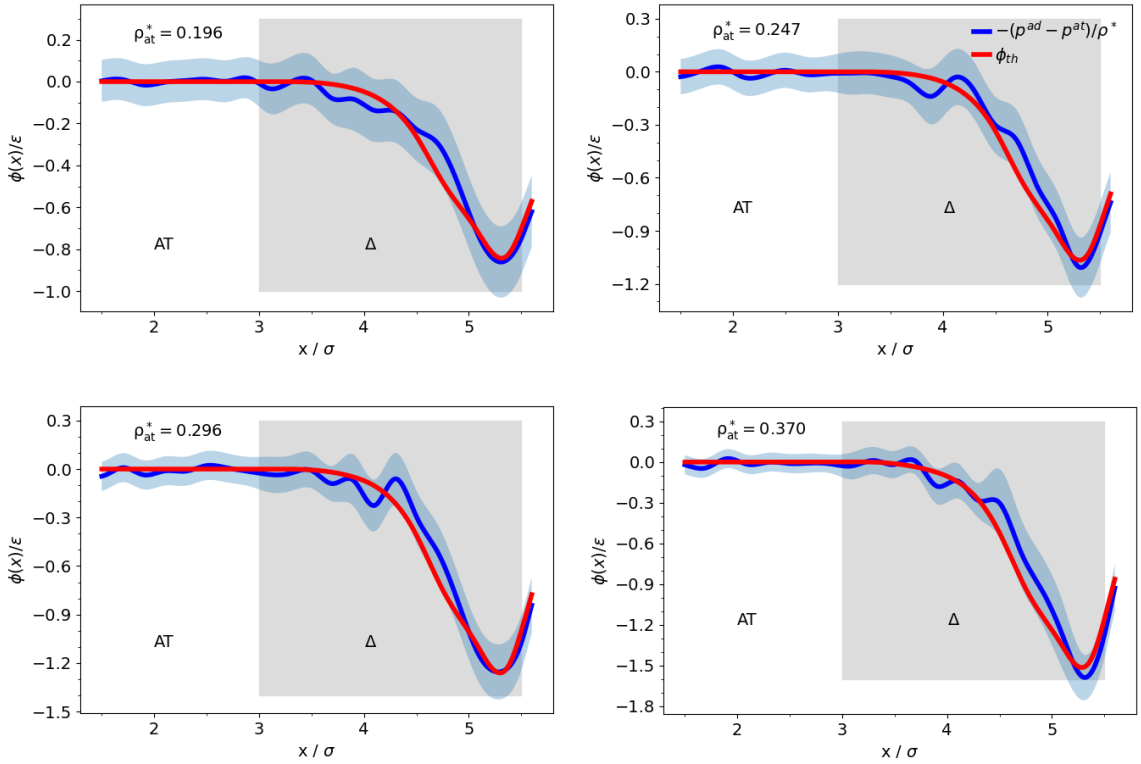


Figure 4.8: Comparison of the required energy to compensate for the pressure difference resulting from the change of resolution (i.e. normalised by the local density) indicated by the blue line and the potential of the thermodynamic force integrated from the calculated thermodynamic force specified by the red line. The shadowed region represents the amount of numerical fluctuation due to the explicit calculation. Instead, the potential of thermodynamic force does not carry numerical fluctuations since once it is determined it is used as a fixed function in the production runs.

for the high-resolution region. Our derivation is then carefully (point-wise) tested with several numerical tests. Thus, the results shown here, together with those of Ref. [73] represent an evolution that contains the approach of Refs. [61, 153, 154] and frames the AdResS techniques within the more general theory of open systems (see also discussion in Ref. [161]).

4.5 Conclusions

The AdResS method has evolved from a numerical algorithm for coupling different resolutions with the main aim of saving computational resources to a more general framework for properly treating open systems embedded in a large environment at well-defined thermodynamic conditions. The passage from a convenient, but empirical, numerical tool [156, 160] to a theoretically well-defined model of open system involves a rigorous mathematical treatment [47] and a computational simplification that allows high transferability of the algorithm from one simulation software to another [49,115]. In between, the theoretical principles and their efficient numerical implementation need to be carefully tested and show consistency w.r.t. to statistical and thermodynamic properties of primary relevance in simulation. The previous work [73] and the current work have the task of showing in detail the physical consistency of the model via its numerical implementation. In this work, we have investigated the behaviour of the stress

tensor and its link to the coupling force (potential) which is one of the main characteristics of the AdResS model. The results show full physical consistency with the physical principle of a proper open system. Furthermore, the knowledge of the link between local pressure and the potential of the thermodynamic force in the Δ region opens access to further conceptual and numerical scenarios. For example, the results of the current study are crucial for designing coupling conditions of the AdResS to hydrodynamics and fluctuating hydrodynamics regulated by field equations (continuum). In this respect, the current paper contributes in a meaningful manner to the development of AdResS as a method of molecular dynamics for open systems.

Chapter 5

Fluctuating Hydrodynamics and coupling the multiscale solver to continuum hydrodynamics

This chapter is accepted to be published in Physical Review Letters and a preprint version is available in Ref. [75].

Abstract

Coupling Adaptive Resolution Simulation (AdResS) with a continuum simulator, which could be applicable to a variety of problems in micro/nano flows, is the next methodological step for simulating open systems. Here, we have implemented a fluctuating hydrodynamics solver to simulate the Landau-Lifshitz Navier-Stokes (LLNS) equations by using a third-order Runge-Kutta numerical discretization method and validated it for different scenarios. Later, a robust coupling algorithm for coupling the current version of AdResS to the developed continuum solver is designed and applied to Lennard-Jones fluids for different test scenarios.

5.1 Introduction

By increasing the interest and number of publications in the field of micro and nano flows, a need has arisen for studying the flow behaviour at micro and nano scales by taking their molecular and atomic level interactions into account [43, 95, 144]. However, Navier-Stokes equations (and other purely PDE approximations) are not able to describe the flow in some problems with small length and/or time scales including cases with large Knudsen number flows and description of regions close to complex geometries or boundary conditions [16, 35, 52, 83, 176, 190].

In statistical mechanics, fluctuations are random deviations from the average state that a system has at equilibrium. All fluctuations increase as the temperature increases and they become smaller at low temperatures. A system at a non-zero temperature does not stay in its equilibrium macroscopic state, but randomly samples all possible states with probabilities given by the Boltzmann distribution. Thermodynamic variables, such as pressure, temperature, or density likewise undergo fluctuation [120]. These fluctuations in some problems with large gradients or special boundaries may affect the stationary response of the system. In such a case, a deterministic continuum-based solver can not capture the fluctuations, thus, one needs to employ a combination of particle-based solver and fluctuating hydrodynamic (FHD) solver depending on

the situation and considering the balance between the precision of the result and computational costs [51].

On the other hand, one can use particle-based simulators (e.g. Molecular Dynamics, Direct Simulation Monte Carlo, and Dissipative Particle Dynamic) to directly capture the effect of fluctuations but this could be computationally expensive as it needs a very large number of particles in some problems [9,14,62]. In many particle-based simulations, most of the computational costs are for those particles far away from the region of interest; while the PDE-based solvers like the Navier-Stokes simulator can simulate the flow at these regions with much lower computational effort. To tackle the aforementioned possible problems, one can employ the Navier-Stokes solver at regions in which such PDE solver is adequate and use the particle-based solvers in (usually) small regions of interest in which the continuum solver fails to describe because of an extreme gradient [192] or closeness to singularities [142] or rarefied regions [83], and finally, couple them through a small buffer region. This treatment for changing resolution through a simulation domain has been explained schematically in Fig.5.1 in which each subdomain provides/gets necessary information from the other one through a transition region.

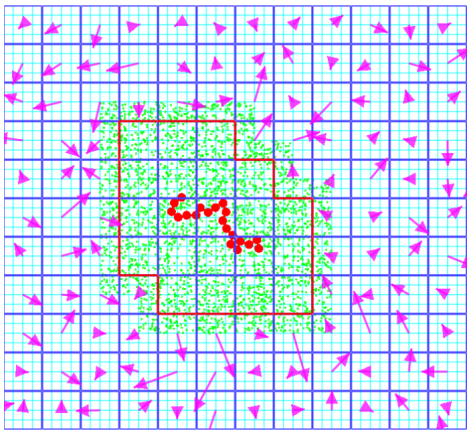


Figure 5.1: A two-dimensional domain including a polymer chain (red circles) in which a small region around the polymer (shown by the red line) is filled with particles (e.g. DSMC particles which have been shown with green circles) and far away regions have been simulated using a hydrodynamic solver. There exists a transition region (the small region outside the red box which has been filled with green particles) to resolve a consistent and stable change of resolutions [51].

In the latest version of Adaptive Resolution Simulation (AdResS), a fully atomistic subdomain is coupled to a reservoir of non-interacting particles through a small buffer region [49]. This configuration of domains reduces the computational costs by doing the all-atom calculation in the relatively small region and treating the rest of the domain as a semi-continuum ideal gas-like fluid [50]. Accordingly, going beyond the current version of AdResS by coupling it to a continuum-based solver can provide more capabilities and benefits for the multiscale solver. This will provide a possibility to simulate some part of the simulation domain with complex situations with a fully atomistic resolution while doing the far-away regions by a continuum-based solver with the possibility of capturing the fluctuations.

5.2 Background

Fluid mechanics problems may consist of several scales in time and space and their applications range from stellar dust dynamics in astrophysics and cloud formation in atmospheric science

to high-speed jet flows in aircraft engines and micro and nanoflows [107]. Computational Fluid Dynamics (CFD) provide a variety of computational methods for solving macroscopic flows. For example, for relatively larger time and space scales the fluid property can be approximated by macroscopic Lagrangian parcels of fluids with Smooth Particle Hydrodynamics (SPH) [78, 128] and Lattice Boltzmann Method (LBM) [23, 28, 76] or represented on Eulerian grid by using Finite Difference (FD), Finite Volume (FV), and Finite Element (FE) methods [126, 151, 185, 189]. However, the continuum PDE-based solvers can not handle situations at microscopic scales [22]. Direct Simulation Monte Carlo (DSMC) [149] or Molecular Dynamics (MD) [166] particle-based solvers can simulate the flow of rarefied gases or dense fluids at the microscopic scale. However, the simulation costs for particle-based solvers are much higher than those for continuum solvers and they are only suitable for microscopic systems [180].

The fluid's state is not constant at the molecular scale and it, even in an equilibrium state, is always changing [52]. Fluctuating Hydrodynamics (FHD) has been an interesting topic in statistical mechanics since the study of light scattering prediction of Rayleigh and Einstein's theory of Brownian motion [148]. Currently, there is a growing interest in studying the fluctuations in fluid mechanics due to the increasing amount of research in nanoscale flows in micro and bio-engineering [7, 12, 93, 108, 146]. Microscopic fluctuations deviate a fluid from its mean state and this property may have a considerable effect on the system's response [16]. As an example of the importance of fluctuations in molecular biology, the molecular motor protein consumes ATP, with a power of 10^{-16} Watts while operating in the background with a power of 10^{-8} Watts [12]. Some examples that fluctuations play a significant role are Brownian molecular motors [12, 133, 146, 201], Kolmogorov flows [17, 18, 130], reaction fronts [138], and droplets breakup [59, 106, 139].

Particle-based numerical simulations inherently include fluctuations as they have irregular dynamics and this can be seen in Molecular Dynamics or Direct Simulation Monte Carlo modellings. However, in the most common format of computational fluid dynamics, partial differential equations are solved and there are no fluctuations in its deterministic format. To include fluctuations in fluid behaviour, Landau and Lifshitz introduced an extended form of the compressible Navier-Stokes equation by adding some white-noise stochastic flux terms to the deterministic compressible flow equations [120]. The Landau-Lifshitz Navier-Stokes (LLNS) equations were originally developed for systems at equilibrium, but later, their validity for non-equilibrium systems has been assessed and verified by molecular simulations [69, 129, 131].

There are several numerical methods for solving LLNS equations. Garcia et al. [71] developed a numerical approach for solving heat equations and one-dimensional LLNS equations. Ladd et al. [117] included stress fluctuations in isothermal Lattice-Boltzmann solver based on a strong theoretical model [1, 54]. Moseler and Lanman [139] for the problem of the breakup of nano-jets added the stochastic stress tensor to LLNS and validated it by Molecular Dynamics results. Sharma and Patankar [173] designed a coupling between fluctuating incompressible solver and suspended Brownian particles. Coveney et al. [35, 40, 79] designed a hybrid scheme for coupling isothermal LLNS solver to Molecular Dynamics simulations. Atzberger et al. [13] developed an immersed boundary method for the fluctuating incompressible Navier-Stokes equations. Voulgarakis and Chu [191] developed a staggered scheme for the isothermal LLNS equations.

Bell et al. [16] developed a central discretization in space and a third-order Runge-Kutta temporal scheme (RK3) for numerically solving the fluctuating hydrodynamics equations. In that work, the evaluation of the numerics for the local variance and spatial correlation structure of equilibrium and non-equilibrium problems have been done. Moreover, Bell et al. [16] proved that the RK3 scheme is an effective discretization approach compared to other schemes for the compressible Navier-Stokes equations.

In the work reported in this thesis, after the preparation of the deterministic Navier-Stokes

solver for compressible flows, a stochastic flux term is added to the equations to include the fluctuations in the deterministic solver. Moreover, for discretizing the partial differential equations, the abovementioned RK3 algorithm is employed.

5.3 Fluctuating Hydrodynamics (FHD)

In statistical mechanics, fluctuations are the random deviations of a system from its average state and these fluctuations become larger at higher temperatures and smaller for temperatures close to absolute zero. The system does not stay at the microscopic equilibrium state but randomly samples all possible states with a Boltzmann distribution probability [8]. Microscopic fluctuations are constantly causing deviations of a fluid from its mean state and these deviations can affect the final response of the system, especially in micro and nano-scale flows. The study of fluctuations in nano-scale fluids is particularly interesting when the fluid is experiencing extreme conditions or close to a hydrodynamic instability [118, 121].

5.3.1 Navier-Stokes equation for compressible flows

The Navier-Stokes partial differential equations are the governing equations of a viscous fluid dynamic and mathematically express the conservation laws for mass, momentum, and energy for a Newtonian fluid. Compared to the hyperbolic Euler equation for inviscid flows, Navier-Stokes equations take the viscous forces into account and is a parabolic equation with better analytical properties and less mathematical structure (not completely integrable) [27, 65].

To derive the Navier-Stokes equations, one has to consider a small volume element of the fluid and write the mass, momentum, and energy balance for the volume element [65]. For the mass balance, this means the total mass coming into or going out of the volume and also the change of the density in the volume should be considered. For the momentum balance, the total force applied to the volume element due to the pressure and viscous forces should be derived and related to the flow velocity by Newton's second law. Finally, the heat produced by the viscous effect and the conductive heat should be taken into consideration for the energy balance equation. By doing so, the following set of equations for the compressible flow are derived (the gravity force is neglected),

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \mathbf{u}) = 0 \tag{5.1}$$

$$\rho(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u}) = -\nabla p + \nabla[\mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^\top)] + \nabla[\lambda(\nabla . \mathbf{u})\mathbf{I}] \tag{5.2}$$

$$\rho \frac{Dh}{Dt} = \frac{Dp}{Dt} + \nabla(k \nabla T) + \Phi \tag{5.3}$$

in the above equations, ρ is the density, $\mathbf{u}$ is the velocity vector, μ and λ are respectively the shear and volume viscosities, p is the pressure, k is the heat conductivity coefficient, T is the temperature, and $h = u + pv$ is the specific enthalpy with u being the specific energy and v being the volume. The value of λ which produces a viscous effect associated with volume change is very difficult to calculate and the term involving λ is often negligible; however, when it is not neglected, the most common approximation is to set $\lambda = -\frac{2}{3}\mu$ [15]. The $\frac{D}{Dt}$ operator is the total derivative with respect to time and is defined as:

$$\frac{D}{Dt} \overset{def}{=} \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla$$

Finally, Φ in the energy balance equation (Eq.5.3) is a function that represents the energy dissipation due to the viscous effects and is calculated as,

$$\Phi = \mu\left(2\left(\frac{\partial u}{\partial x}\right)^2 + 2\left(\frac{\partial v}{\partial y}\right)^2 + 2\left(\frac{\partial w}{\partial z}\right)^2\right) + \mu\left(\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z}\right)^2 + \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right)^2\right) + \lambda(\nabla \mathbf{u})^2$$

Here, u, v, and w are the velocity components in x, y, and z directions.

Solving Navier-Stokes equations (Eq.5.1-5.3) together with the equation of state will determine the flow behaviour in the system for density, velocity, pressure, and temperature fields.

5.3.2 Fluctuating hydrodynamics and LLNS equations

Following Ref. [16], the Landau-Lifshitz Navier-Stokes equations are an extended form of the Navier-Stokes equation for deterministic hydrodynamics modelling with introducing a stochastic flux term [119]:

$$\mathbf{U}_t = -\nabla \mathbf{F} + \nabla \mathbf{D} + \nabla \mathbf{S} \tag{5.4}$$

where $\mathbf{U}$ is the vector of conserved quantities (mass, momentum and energy density),

$$\mathbf{U} = \begin{pmatrix} \rho \\ \mathbf{J} \\ E \end{pmatrix} \tag{5.5}$$

where $\mathbf{J} = \rho \mathbf{u}$ is the momentum density and E is the energy density.The advective ($\mathbf{F}$) and diffusive ($\mathbf{D}$) fluxes are given by,

$$\mathbf{F} = \begin{pmatrix} \rho \mathbf{u} \\ \rho \mathbf{u} \cdot \mathbf{u} \\ (E + P)\mathbf{u} \end{pmatrix} \quad , \quad \mathbf{D} = \begin{pmatrix} 0 \\ \tau \\ \tau \cdot \mathbf{u} + k \nabla T \end{pmatrix} \tag{5.6}$$

with $\mathbf{u}$, P, and T being the fluid velocity, pressure, and temperature, respectively, and $\tau = \eta(\nabla \mathbf{u} + \nabla \mathbf{u}^\top - \frac{2}{3} I \nabla \mathbf{u})$ is the stress tensor in which η and k are the coefficients of viscosity and thermal conductivity, respectively. Except for the mass flux, the other fluxes may have some deviations from their mean values because of spontaneous thermal fluctuations and these fluctuations are introduced by the stochastic flux term, $\mathbf{S}$, in LLNS equations. These noise terms are white in space and time and are formulated using fluctuation-dissipation relations to yield the equilibrium covariances of the fluctuations [16].

$$\mathbf{S} = \begin{pmatrix} 0 \\ \mathcal{S} \\ \mathcal{Q} + \mathbf{u} \cdot \mathcal{S} \end{pmatrix} \tag{5.7}$$

where the stochastic stress tensor ($\mathcal{S}$) and heat flux ($\mathcal{Q}$) have the mean value of zero and following covariances,

$$\langle \mathcal{S}_{ij}(\mathbf{r}, t) \mathcal{S}_{kl}(\mathbf{r}', t') \rangle = 2 k_B \eta T (\delta^K_{ik}\delta^K_{jl} + \delta^K_{il}\delta^K_{jk} - \frac{2}{3}\delta^K_{ij}\delta^K_{kl}) \delta(\mathbf{r} - \mathbf{r}') \delta(t - t') \tag{5.8}$$

$$\langle \mathcal{Q}_i(\mathbf{r}, t) \mathcal{Q}_j(\mathbf{r}', t') \rangle = 2 k_B k T^2 \delta^K_{ij} \delta(\mathbf{r} - \mathbf{r}') \delta(t - t') \tag{5.9}$$

and

$$\langle \mathcal{S}_{ij}(\mathbf{r}, t) \mathcal{Q}_k(\mathbf{r}', t') \rangle = 0 \tag{5.10}$$

where k_B is the Boltzmann constant and δ^K_{ij} is the Kronecker delta,

$$\delta^K_{ij} = \begin{cases} 0, & i \neq j \\ 1, & i = j \end{cases}$$

Here, the full three-dimensional description of Eq.5.4 has been simplified to a 1D equations set as it could be enough for coupling with AdResS. A one-dimensional coupling set-up is simple enough to allow for simulations that assess the general validity of the basic principles on which the technique is based and at the same time it is already sufficient for applications to complex molecular systems such as hydrated biological membranes in a thermal and/or density field or water-ionic liquid mixtures in which a thermal field can drive the phase separation, to cite just a few [145,171,172,197]. There, one can use a two or three-dimensional particle-based environment to couple it to a one-dimensional continuum solver. However, it's quite straightforward to extend the equations and the code from a one-dimensional to a two or three-dimensional solver in future. By this simplification, Eq.5.4 can be written as below,

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ J\\ E\end{pmatrix} = -\frac{\partial}{\partial x}\begin{pmatrix}\rho u\\ \rho u^2+P\\ (E+P)u\end{pmatrix} + \frac{\partial}{\partial x}\begin{pmatrix}0\\ \frac{4}{3}\eta\partial_x u\\ \frac{4}{3}\eta u\partial_x u + k\partial_x T\end{pmatrix} + \frac{\partial}{\partial x}\begin{pmatrix}0\\ s\\ q+us\end{pmatrix} \tag{5.11}$$

where

$$\begin{aligned}\langle s(x,t)s(x',t')\rangle &= \frac{1}{A^2}\int dy\int dy'\int dz\int dz'\langle \mathcal{S}_{xx}(\mathbf{r},t)\mathcal{S}_{xx}(\mathbf{r}',t')\rangle\\ &= \frac{8k_B\eta T}{3A}\delta(x-x')\delta(t-t')\\ \langle q(x,t)q(x',t')\rangle &= \frac{1}{A^2}\int dy\int dy'\int dz\int dz'\langle \mathcal{Q}_{x}(\mathbf{r},t)\mathcal{Q}_{x}(\mathbf{r}',t')\rangle\\ &= \frac{2k_Bk T^2}{A}\delta(x-x')\delta(t-t')\end{aligned}$$

in which A is the surface area of the system in yz-plane.

5.3.3 Numerical discretization

There are several discretization schemes for the LLNS equation, but here we restrict ourselves to finite volume schemes in which all variables are collocated; they are employed as a basis for the hybrid coupling method to a particle subdomain [16, 51]. There are several methods based on CFD schemes that are commonly used for the Navier-Stokes equations and could be extended to LLNS equations like MacCormack [16,71] and Piecewise Parabolic Method [16,31,134]. Here, those discretization methods are briefly introduced; however, we will focus on the variance-preserving third-order Runge-Kutta scheme as it shows to have more accuracy compared to the other mentioned schemes [16]. The method is based on a third-order Runge-Kutta temporal integrator as an ODE integration algorithm to advance the solution combined with a centred discretization of hyperbolic, diffusive and stochastic fluxes to ensure that the algorithm satisfies discrete fluctuation-dissipation balance [52]. Here, the motivation is not only a higher order of accuracy but also its robustness as it reduces the order of stochastic differential equations; also, a simple forward Euler scheme may be unstable as there is no dissipation term in the conservation of momentum equation [16].

MacCormack Scheme

The MacCormack discretization scheme is a variant of the two-step Lax-Wendroff method for discretizing LLNS equations [71] and is described in the following format with a predictor and corrector step,

$$\mathbf{U}_j^* = \mathbf{U}_j^n + \frac{\Delta t}{\Delta x}\left[-(\mathbf{F}_j^n - \mathbf{F}_{j-1}^n) + (\mathbf{D}_{j+1/2}^n - \mathbf{D}_{j-1/2}^n) + (\mathbf{S}_{j+1/2}^n - \mathbf{S}_{j-1/2}^n)\right]$$

$$\mathbf{U}_j^{**} = \mathbf{U}_j^* + \frac{\Delta t}{\Delta x}\left[-(\mathbf{F}_j^* - \mathbf{F}_{j-1}^*) + (\mathbf{D}_{j+1/2}^* - \mathbf{D}_{j-1/2}^*) + (\mathbf{S}_{j+1/2}^* - \mathbf{S}_{j-1/2}^*)\right]$$

$$\mathbf{U}_j^{n+1} = \frac{1}{2}(\mathbf{U}_j^n + \mathbf{U}_j^{**}) \tag{5.12}$$

in which the values at $j + 1/2$ are a simple finite difference approximation of the cell centre values.

The approximation of stochastic stress tensor ($s_{j+1/2}$) and heat flux ($q_{j+1/2}$) at the edge of the cells are (which is the same for all discretization methods presented later):

$$q_{j+1/2}^n = \sqrt{\frac{k_B}{\Delta t V_c}(k_{j+1}T_{j+1}^2 + k_j T_j^2)}\mathcal{R}(\mu, \sigma^2), \tag{5.13}$$

$$s_{j+1/2}^n = \sqrt{\frac{4k_B}{3\Delta t V_c}(\eta_{j+1}T_{j+1} + \eta_j T_j)}\mathcal{R}(\mu, \sigma^2) \tag{5.14}$$

with V_c being the volume of each continuum cell and $\mathcal{R}(\mu, \sigma^2)$ are independent Gaussian distributed random numbers with zero mean and unit variance.

Bell et al. [16] showed that with the abovementioned stochastic fluxes, the flux's variance reduces to half of its original magnitude in the MacCormack algorithm; thus, they suggested using $\mathbf{S}_{new} = \mathbf{S}\sqrt{2}$ instead in all steps in equations 5.12.

Piecewise Parabolic method

The piecewise parabolic method is a higher-order Godunov scheme that showed satisfactory results for solving LLNS [134]. This scheme consists of a predictor and corrector steps and replacement of $\mathbf{S}$ with $\mathbf{S}\sqrt{2}$ similar to the MacCormack method to preserve the variance [16].

$$\mathbf{U}_j^* = \mathbf{U}_j^n - \frac{\Delta t}{\Delta x}\mathbf{F}_j^n + \frac{\Delta t}{\Delta x}(\mathbf{D}_j^n + \mathbf{S}_{new,j}^n)$$

$$\mathbf{U}_j^{n+1} = \mathbf{U}_j^n - \frac{\Delta t}{\Delta x}\mathbf{F}_j^n + \frac{1}{2}(\frac{\Delta t}{\Delta x})(\mathbf{D}_j^n + \mathbf{S}_{new,j}^n + \mathbf{D}_j^* + \mathbf{S}_{new,j}^*) \tag{5.15}$$

Variance preserving third-order Runge-Kutta method

Bell and collaborators [16] reviewed different discretization methods explained here and developed the new RK3 method to preserve the variance and fluctuations better compared to other methods and showed the improvement in results by using this method. The third-order Runge-Kutta scheme can be written in the following three-stage form,

$$\mathbf{U}_j^{n+1/3} = \mathbf{U}_j^n + (\frac{\Delta t}{\Delta x})\left(\mathcal{F}_{j+1/2}^n - \mathcal{F}_{j-1/2}^n\right)$$

$$\mathbf{U}_j^{n+2/3} = \frac{3}{4}\mathbf{U}_j^n + \frac{1}{4}\mathbf{U}_j^{n+1/3} + \frac{1}{4}(\frac{\Delta t}{\Delta x})\left(\mathcal{F}_{j+1/2}^{n+1/3} - \mathcal{F}_{j-1/2}^{n+1/3}\right)$$

$$\mathbf{U}_j^{n+1} = \frac{1}{3}\mathbf{U}_j^n + \frac{2}{3}\mathbf{U}_j^{n+2/3} + \frac{2}{3}(\frac{\Delta t}{\Delta x})\left(\mathcal{F}_{j+1/2}^{n+2/3} - \mathcal{F}_{j-1/2}^{n+2/3}\right) \tag{5.16}$$

where $\mathcal{F} = -\mathbf{F} + \mathbf{D} + \mathbf{S}$. By combining all three stages,

$$\mathbf{U}_j^{n+1} = \mathbf{U}_j^n + (\frac{\Delta t}{\Delta x})\left[\frac{1}{6}(\mathcal{F}_{j+1/2}^n - \mathcal{F}_{j-1/2}^n) + \frac{1}{6}(\mathcal{F}_{j+1/2}^{n+1/3} - \mathcal{F}_{j-1/2}^{n+1/3}) + \frac{2}{3}(\mathcal{F}_{j+1/2}^{n+2/3} - \mathcal{F}_{j-1/2}^{n+2/3})\right] \tag{5.17}$$

The stochastic components of the flux are independent, identically distributed Gaussian random variables with mean zero and unit variance. Thus, the flux variance in Eq.5.17 at $j+1/2$ is:

$$\begin{aligned}&\langle\delta(\frac{1}{6}(\mathbf{S}^0_{j+1/2})+\frac{1}{6}(\mathbf{S}^{1/3}_{j+1/2})+\frac{2}{3}(\mathbf{S}^2_{j+1/2}/3))^2\rangle\\ =&\frac{1}{6}^2\langle(\delta\mathbf{S}^0_{j+1/2})^2\rangle+\frac{1}{6}^2\langle(\delta\mathbf{S}^{1/3}_{j+1/2})^2\rangle+\frac{2}{3}^2\langle(\delta\mathbf{S}^{2/3}_{j+1/2})^2\rangle=\frac{\sigma^2}{2}\end{aligned}$$

According to the above calculation, the variance of the flux is reduced to half of its original magnitude by using this integrator. In order to preserve the variance, one can replace $\mathbf{S}_{j+1/2}$ by $\sqrt{2}\mathbf{S}_{j+1/2}$ and consequently changes the definition of $\mathcal{F}$ to $\mathcal{F}=-\mathbf{F}+\mathbf{D}+\sqrt{2}\mathbf{S}$.

The above-mentioned procedure and correction do not affect the continuity equation as the stochastic flux does not contribute to the mass balance equation. Bell and collaborators [16] have corrected this effect using a special interpolation scheme by augmenting the variance to compensate for the density reduction due to the temporal averaging. For that purpose, they have used the following expression for J and other conserved quantities to interpolate the cell-edged values from cell-centred values and compute face fluxes.

$$J_{j+1/2}=\alpha_1(J_j+J_{j+1})-\alpha_2(J_{j-1}+J_{j+2}) \tag{5.18}$$

where

$$\alpha_1=(\sqrt{7}+1)/4 \qquad \text{and} \qquad \alpha_2=(\sqrt{7}-1)/4$$

5.4 Numerical results: FHD

In this section, after explaining the technical details of the simulations, we present the results for the application of the described discretization method (Eq.5.16) for solving the fluctuating hydrodynamic equations for a one-dimensional domain. For this purpose, firstly a deterministic Navier-Stokes equation with separated diffusive and advective terms will be solved and later the stochastic fluxes will be added.

5.4.1 Technical details and stability criteria

We have used Argon gas as a prototype of a dilute compressible fluid at the temperature of $300[K]$ with the following thermodynamic properties,

particle mass (m)	6.6335209×10^{-26} $[kg]$
density(ρ)	1.603 $[kg/m^3]$
viscosity (η)	2.23×10^{-5} $[kg/m.s]$
heat conductivity (k)	0.017746 $[W/m.K]$

For all simulations, the viscosity and thermal conductivity of Argon are functions of temperature with the form of $\eta=\eta_0\sqrt{T}$ and $k=k_0\sqrt{T}$. For the dilute gas, the pressure is a function of density and temperature by $P=\rho RT$ and the energy density can be calculated by $E=c_v\rho T+\frac{1}{2}\rho u^2$. For the monatomic gas of Argon, $R=k_B/m$ and $c_v=R/(\gamma-1)$ in which m is the mass of each gas particle and $\gamma=5/3$.

In a numeric simulation of compressible Navier-Stokes equations, the hyperbolic and diffusive stability criteria determine the maximum time step and its relation to the space discretization size,

$$(|u|+c_s)\frac{\Delta t}{\Delta x}\leq 1 \qquad \text{and} \qquad \max\left(\frac{4}{3}\frac{\overline{\eta}}{\overline{\rho}},\frac{\overline{k}}{\overline{\rho}c_v}\right)\frac{\Delta t}{\Delta x^2}\leq\frac{1}{2} \tag{5.19}$$

in which $c_s = \sqrt{\gamma \overline{P}/\overline{\rho}} = \sqrt{\gamma R \overline{T}}$ is the speed of sound while over-line parameters are the values in the equilibrium state. Moreover, for the stability of the advection term, the discretization scheme is suitable for a very small cell's Reynolds numbers ($Re_c << 2$) [16]; this means that to have a very small Reynolds number, $Re_c = \overline{\rho u}\Delta x/\overline{\mu}$, we need to choose a very small cell size. Accordingly, a very small size for the space discretization is selected with $\Delta x = 10[nm]$ which restricted us to choose $\Delta t = 1[ps]$ according to the diffusive and advective stability inequalities in Eq.5.19. These time and space discretizations scales are also helpful for the final goal of coupling the fluctuating hydrodynamics solver to the Adaptive Resolution Simulation set-up.

In all simulations, a periodic boundary condition is imposed; however, any other boundary condition could be easily introduced in the simulation system. With the periodic boundary condition, at each integration step from the three-stage RK3 model (see Eq. 5.16) during each continuum time step, one needs to calculate the left/right boundary values by replacing the right/left side's values.

5.4.2 Deterministic Navier-Stokes

The deterministic solver for Navier-Stokes equations with the RK3 discretization scheme has been tested under different initial conditions and the results are reported in these paragraphs. These simulations have been performed with pure diffusion, pure advection, and the complete Navier-Stokes equations with diffusion and advection for all cases. The results for an acoustic wave with sinus-like initial properties propagation are reported in the next paragraphs.

Here, we initially set the temperature as a sinus-like function around $300[K]$ with an amplitude of $10[K]$ (see Fig.5.2) and consequently set the initial density to have a constant initial pressure over the simulation domain as $P = \rho RT$ for the ideal gas of Argon. Moreover, just in the case of pure diffusion simulation, we initially set the velocity as a sinus-like function around $10[cm/s]$ with an amplitude of $5[cm/s]$ (see Fig.5.2) to see the effect of hydrodynamic diffusion besides the heat conduction.

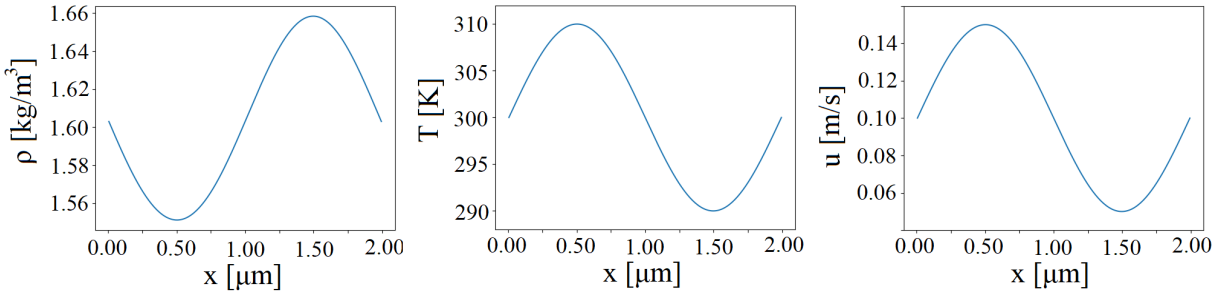


Figure 5.2: initial conditions for generating an acoustic wave with sinus-like density, temperature, and velocity in such a way that the pressure remains constant along the simulation domain with a length of $2[\mu m]$ and $\Delta x = 10[nm]$ and $\Delta t = 1[ps]$.

Diffusion

Here, we have turned off the advection contribution to see how the initial distribution in temperature and velocity diffuses through the simulation domain. In Fig.5.3, the profiles of density, temperature, and velocity have been shown and one can see the diffusion effect over time.

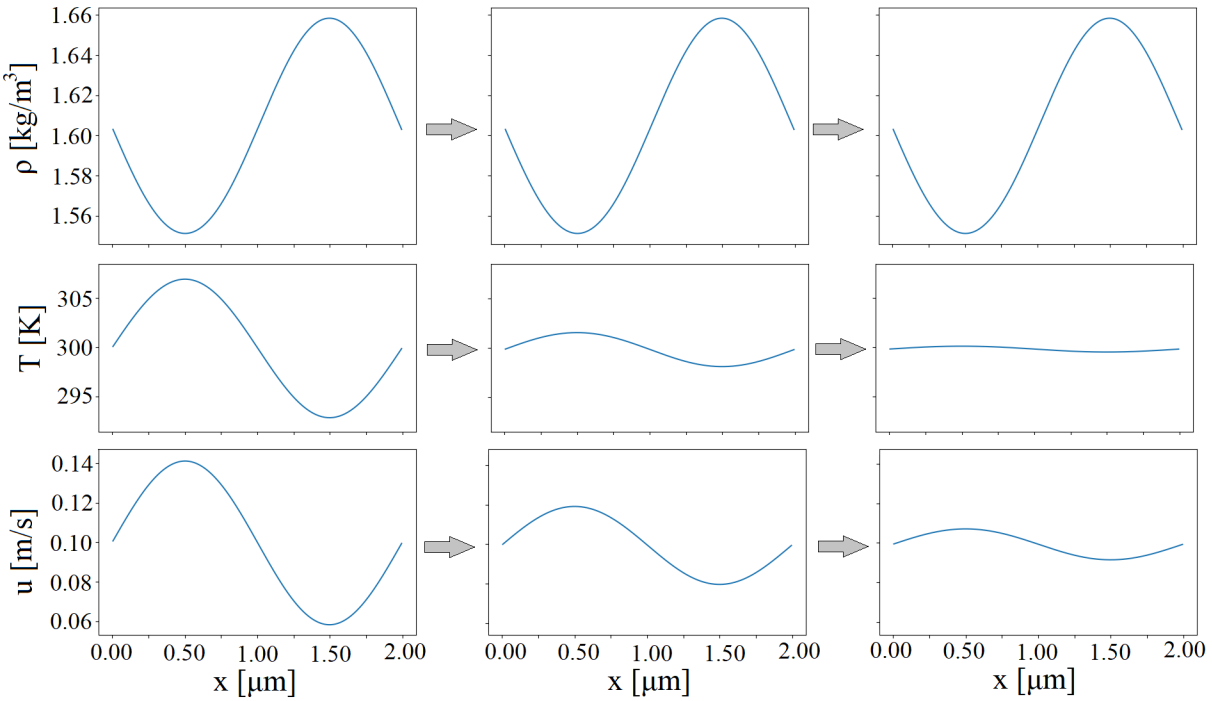


Figure 5.3: The diffusion of initial periodic distribution (see Fig.5.2) in density (first row), temperature (second row), and velocity (third row) profiles in the initial (left column), middle (middle column) and end (right column) stages of simulation time for pure diffusion equation. These results show that the initial perturbation in the state of the fluid is diffusing into the domain while staying the same for the density profile according to Eq.5.11.

According to Fig.5.3, the initial perturbation in temperature and velocity profiles is diffusing into the whole domain and the density profile doesn't change over time as the diffusion equation does not have any contribution to the continuity equation according to Eq.5.11. As can be seen, the temperature is diffusing faster than velocity after a while. This is because the conduction effect for this flow is stronger than the viscous effect; however, both of them would completely diffuse after a long time.

Advection

One can observe the effect of advection by turning off the diffusive fluxes in Eq.5.11 which leads to the Euler equation for inviscid flows. In an advection equation, any initial distribution will advect/move along the simulation domain. In Fig.5.4, the advection of an initial periodic distribution in temperature as well as density profiles is shown.

Advection and diffusion

Here, both diffusive and hyperbolic fluxes have been combined to form the deterministic Navier-Stokes equations, and those fluxes presented in previous subsections are acting on the fluid flow simultaneously. In the following, the results for the progress of an initial periodic distribution in temperature profile (as well as density profile) have been shown. It is necessary to mention that as the diffusive fluxes are very smaller than advective terms, it takes a too long time to see

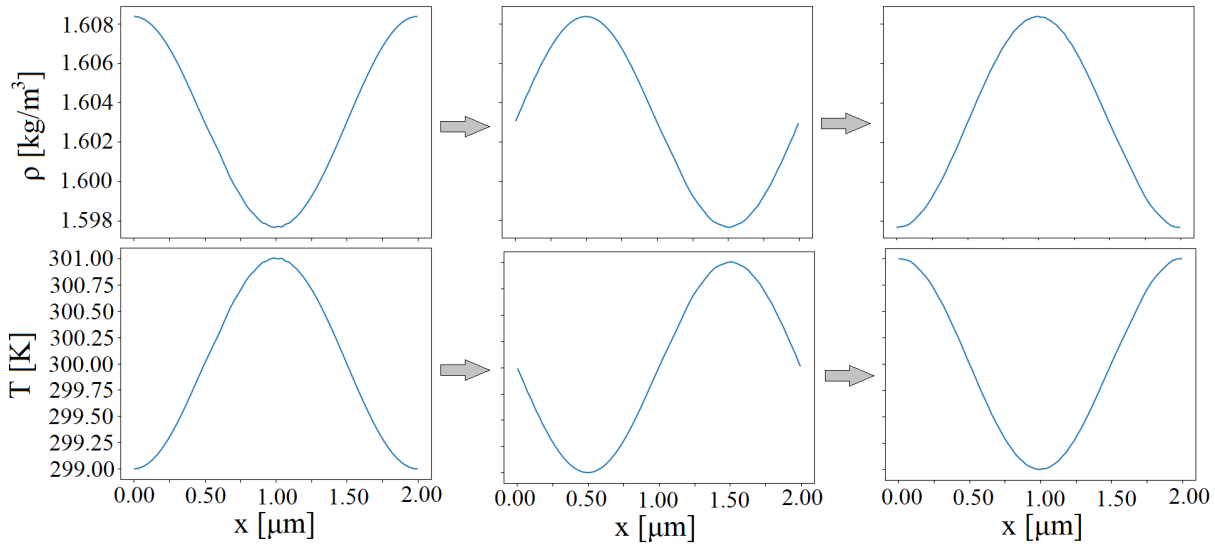


Figure 5.4: Advection of an initial sinusoidal perturbation in density (first row) and temperature (second row) profiles over time. These results show that the initial perturbation in the state of the fluid is advecting into the domain according to Eq.5.11.

both advection and diffusion effects in the simulation domain; so, we artificially enlarged the diffusive fluxes (by increasing the viscosity and heat conduction coefficients by a factor of 5000) to capture its effect in a shorter (real) time.

5.4.3 Fluctuating Navier-Stokes

After preparing the deterministic Navier-Stokes solver for compressible flow and presenting the corresponding results for various initial conditions, the stochastic flux terms will be added to the deterministic partial differential equations (i.e. **S** in Eq.5.4). With fluctuating Navier-Stokes solver, one expects to get similar results for the previously presented initial conditions albeit with fluctuations; however, even the steady-state results may be different for some problems by including those fluctuations (see the piston problem in Ref. [51]).

In the following, the results of the fluctuating hydrodynamics solver for two problems with sinusoidal and sharp jump initial perturbations are presented for the density and temperature profiles.

5.5 Coupling AdResS to FHD

Modelling of many micro and nano-flow systems is computationally very expensive to be simulated by a particle-based solver (e.g. Molecular Dynamics, Dissipative Particle Dynamics, and Direct Simulation Monte-Carlo) but at the same time also continuum-based solvers (PDE-based solvers) are not able to provide enough information and details in the region of interest. An alternative idea is to study the complex region with particle-based solvers and couple it with a

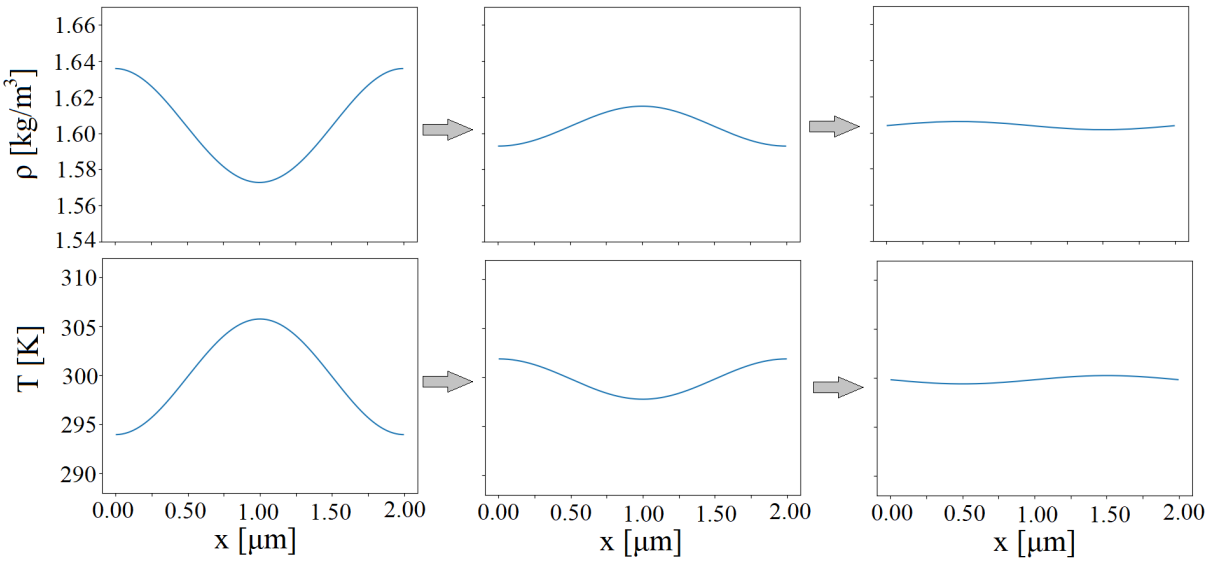


Figure 5.5: propagation of density (first row) and temperature (second row) profiles in a case that advection and diffusion fluxes act simultaneously with the deterministic Navier-Stokes solver on the fluid flow with a constant flow velocity and initial periodic temperature with constant initial pressure. These results show that the initial perturbation in the state of the fluid is advecting and diffusing into the domain according to Eq.5.11.

continuum-based solver which takes care of the rest of the simulation domain far from the region of microscopic details. This will save considerable computational resources while providing adequate information in the region of interest [42].

5.5.1 Background

In recent decades, various attempts to couple a Molecular Dynamics simulator to continuum hydrodynamics have been done and are reported in the literature [84, 147]. In many approaches, a continuum-particle overlap region exists that acts as a separation boundary between the continuum and particle subdomains with different resolutions. For the overlap region, state variables and flux coupling scheme are introduced to take care of the conservation of bulk mass, momentum and energy [37, 51, 63, 141, 147, 193]. An overlap region is needed between the MD and continuum parts of the domain to avoid the sharp oscillations in density and pressure between different resolutions of the same nature as that was needed for the AdResS scheme [157].

For coupling and transferring data between particle and continuum subdomains, one may use a flux-state scheme as it preserves the conservation of variables and could be straightforwardly employed in the simulation. In this scheme, the continuum subdomain provides a state-based boundary condition for the particle subdomain through the insertion of particles into the reservoir region at the boundary of the particle subdomain at each particle time step. During each continuum time step, a certain number of particle time steps are taken and the total flux carried by those particles that cross the interface between particle and continuum subdomains is recorded and would be applied to the continuum subdomain as a boundary condition [51]. Thus, the continuum solver is oblivious to what happens in the particle subdomain; instead, it feels the influence of those conditions through boundary conditions [51, 70, 202].

In *state-flux* coupling scheme, the continuum solver provides the conserved variables ($\mathbf{U}$ in Eq.5.4) in the continuum reservoir macrocells (cells which are considered for discretizing continuum equations) near the particle-continuum interface to the particle subdomain; thus, the

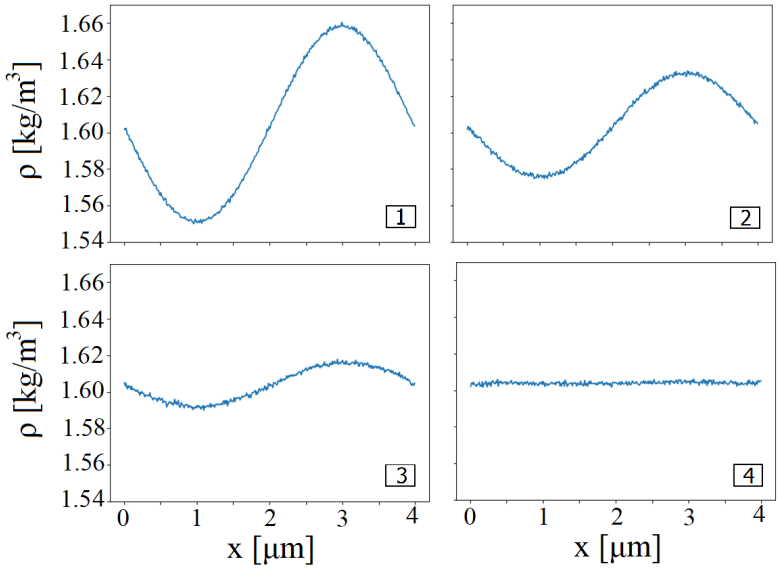


Figure 5.6: Density profile results for the fluctuating hydrodynamics solver with initial sinusoidal perturbation in the density and temperature profiles in constant pressure. The order of plots over time is written at the corner of each frame and the very initial profile without including thermal fluctuations is not presented.

continuum state is imposed as a boundary condition to the particle region. On the other hand, the particle solver provides the flux through the interface to the continuum subdomain; thus, the particle flux is imposed as a boundary condition to the continuum region. It is noteworthy to mention that the information between particle and continuum subdomains is exchanged every certain number of particle (micro) time steps, at the beginning/end of each coupling time steps [51].

This state-flux coupling method is used by Donev et al. [51] to couple a continuum subdomain to a particle subdomain with the Isotropic Direct Simulation Monte Carlo (I-DSMC) method which is a dense fluid generalization of the DSMC algorithm for rarefied gas flows [9] with a structureless nature. However, in the Adaptive Resolution Simulation (AdResS), the aim is to couple a continuum-based solver (fluctuating hydrodynamic solver) with a Molecular Dynamics solver instead of DSMC particles. There are papers in the literature in which an MD solver is employed in the particle subdomain and a flux-flux scheme has been used instead of a state-flux scheme [41, 43, 80, 137, 200] as it is challenging to insert particles into particle reservoir in MD simulations which are needed for the state-flux scheme. In summary, I) for the continuum subdomain, one can solve the fluctuating hydrodynamic equations by considering a **flux** resulting from particle side simulation as a boundary condition at the interface. II) for the particle subdomain, one can do the particle-based simulation by considering a state resulting from the continuum side as a boundary condition at the interface.

For calculating the aforementioned *flux*, one needs to sum up the entire energy, momentum, and interaction energy between particles in the particle subdomain (inside the region bordered with the red line in Fig.5.1) and those in the reservoir (green particles outside the region bordered with the red line in Fig.5.1). The *State* is calculated based on the result of the continuum solver in macrocells in the reservoir region. Then, based on the result of the continuum solver, a certain number of particles (based on the target density) with specific velocity distribution (based on the resulted temperature) would be inserted into the reservoir region [51].

Such coupling approaches with the hybrid scheme have been applied to the coupling of a particle-based solver to a finite element method with deterministic [26, 39, 42, 55, 63, 64, 112, 125, 143, 147]

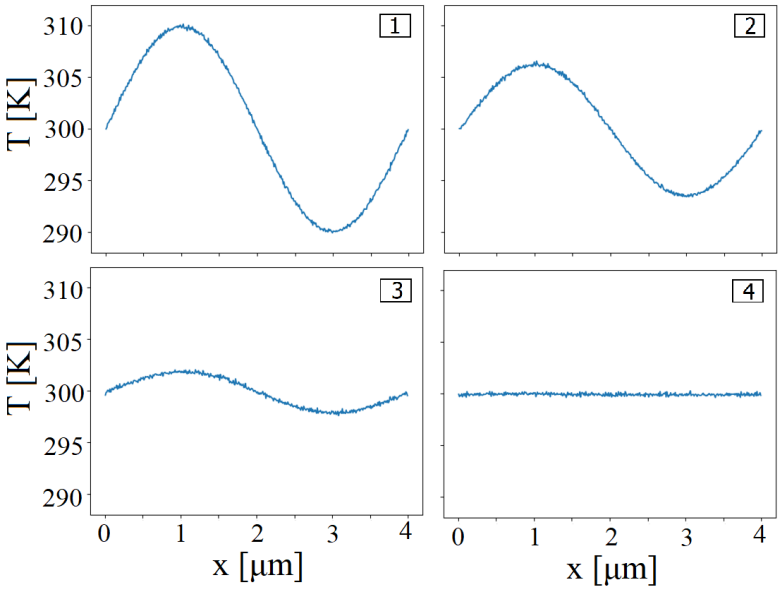


Figure 5.7: Temperature profile results for the fluctuating hydrodynamics solver with initial sinusoidal perturbation in the density and temperature profiles in constant pressure. The order of plots over time is written at the corner of each frame and the very initial profile without including thermal fluctuations is not shown.

or fluctuating Navier-Stokes equations [33, 41] and quantum mechanics model [32, 91] mostly for studying simple liquids such as Lennard-Jones fluid [38] or water [34].

Hettithanthrige et al. [200] discussed various approaches in a hybrid coupling scheme for coupling a continuum domain to an atomistic subdomain with Direct Simulation Monte-Carlo scheme. Delgado-Buscalioni et al. studied the hybrid scheme with a flux-flux interface coupling for MD-continuum set-up with unsteady flows at constant temperature tested by shear and acoustic waves [33, 41, 42]. Donev et al. [51] continued a previously developed continuum-particle coupling method [202] for coupling a fluctuating hydrodynamic solver [16] to Isotropic Direct Simulation Monte-Carlo (I-DSMC) particle-based solver by using the aforementioned flux-state hybrid coupling scheme. Later, Delgado-Buscalioni et al. presented a triple-scale simulation of molecular liquids by coupling AdResS to the deterministic continuum solver which covers a length scale from molecular dynamics to coarse-grained and beyond to continuum hydrodynamic. This coupling scheme was developed for a certain temperature and by insertion and deletion of particles in the reservoir region of AdResS which acts as a hybrid region for coupling to implement the boundary conditions [43]. In recent years, Korotkin and Karabasov developed a particle-continuum coupling for hybrid simulations of liquids based on a two-phase flow analogy with coupling AdResS to a fluctuating hydrodynamic solver by assuming the absence of macroscopic temperature gradients [96].

In this work, a new version of the flux-state hybrid coupling scheme presented before is developed to make the particle and continuum-based solvers interact without the artificial insertion/deletion of particles for conservation benefits. The developed coupling algorithm is presented in the next part and later the necessary technical details for the simulation set-up are explained, and finally, the results of the coupling system for a variety of test scenarios are presented.

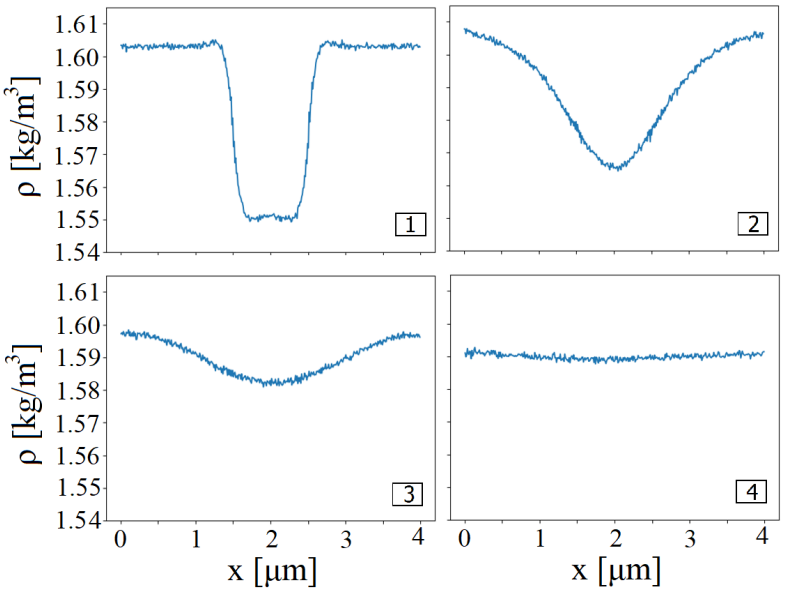


Figure 5.8: Density profile results for the fluctuating hydrodynamics solver with an initial sharp jump in the density and temperature profiles in constant pressure. The order of plots over time is written at the corner of each frame and the very initial profile without including thermal fluctuations is not presented.

5.5.2 Methodology and algorithm

In the state-flux hybrid coupling scheme utilized by Donev et al. [51], the output of the continuum solver at the neighbour cells of the particle subdomain acts as a state for the boundary condition of the particle subdomain. This means some particles with specific positions and velocities will be inserted into the reservoir region of the AdResS to reproduce the density, momentum, and energy of the corresponding continuum cells. On the other hand, the total mass, momentum, and energy of particles crossing from the particle-continuum interface will be calculated during particle simulation and imposed as a boundary condition to the continuum solver at the interface.

To couple the Adaptive Resolution simulation to the fluctuating hydrodynamic solver, we have defined the interface at the border between the AT and Δ regions of the AdResS set-up, indicated by a red line in Fig.5.10. Here, to avoid the artificial insertion/deletion of particles in the reservoir region of AdResS (TR region in Fig.5.10), we employed a new scheme with the use of a dictionary of thermodynamic forces which acts as a tool to transfer the continuum information into the particle subdomain. In AdResS, the thermodynamic force is calculated iteratively and acts as an external force to preserve the properties of the fully atomistic simulation of reference. In this new approach, a list of thermodynamic forces for different densities and temperatures is prepared as a two-variable function that can predict the required thermodynamic force by interpolation as a function of the length in the direction of the change of resolution. Thus, the idea is to implement the resulted state from the continuum solver at the neighbour cells of the particle subdomain (shaded cells in Fig.5.10) by the corresponding thermodynamic force for that state to the particle-based solver (AdResS). Moreover, it is required to set the reservoir temperature according to the calculated temperature of the neighbour cells in continuum simulation during each coupling time step. On the other hand, the calculation of the fluxes to pass to the continuum solver as a boundary condition is similar to the references, which means the total mass, momentum, and energy of particles crossing the particle-continuum interface will be imposed as a boundary condition to the continuum solver at the interface.

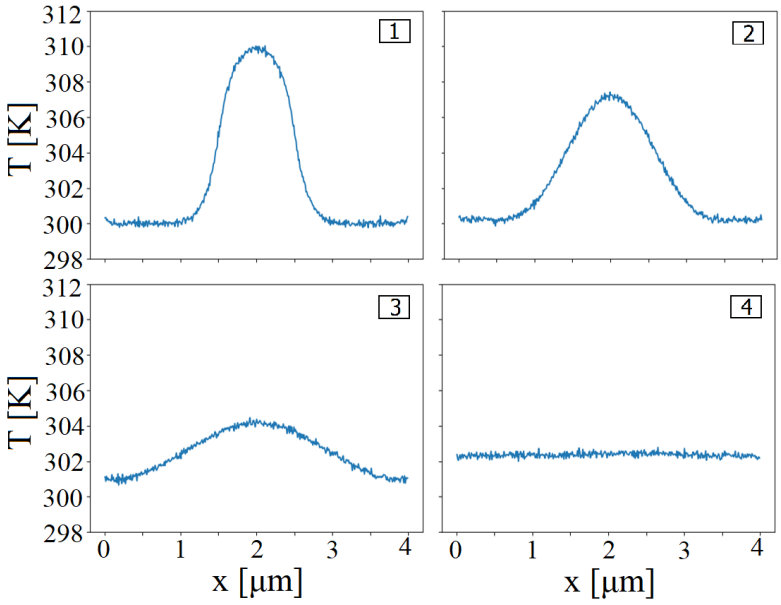


Figure 5.9: Temperature profile results for the fluctuating hydrodynamics solver with an initial sharp jump in the density and temperature profiles in constant pressure. The order of plots over time is written at the corner of each frame and the very initial profile without including thermal fluctuations is not shown.

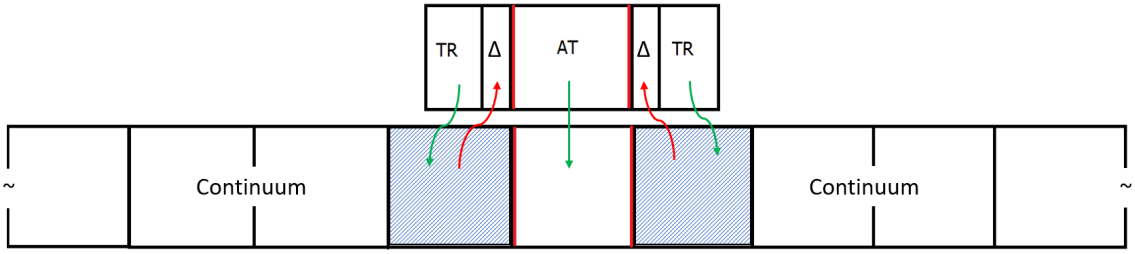


Figure 5.10: The schematic figure of the coupling AdResS to the fluctuating hydrodynamics solver. The particle subdomain is located in the middle of the domain and the interface of the continuum and particle subdomains is indicated by the red line at the left and right borders of the AT and Δ regions of AdResS. The red and green arrows indicate the direction of information transfer according to the newly developed state-flux coupling algorithm and the shaded continuum cells represent the neighbour cells of the particle subdomain at the left and right sides which overlap with the reservoir region of AdResS to ensure the smooth coupling of the solvers.

The coupling simulation starts with an initial array of density, velocity, and temperature for the whole domain as it was a pure continuum domain. Then, based on the density and temperature of the neighbour macrocells, a new thermodynamic force for the left and right sides of the AdResS simulation will be calculated. In the next step, AdResS simulation will be performed by setting the left and right reservoirs' temperature according to the temperature of the continuum neighbour cells. Then, the quantities of interest (density, velocity, and temperature) in the region of interest (AT region of AdResS domain) together with the mass, momentum, and energy fluxes at the interface are calculated during AdResS simulation. Finally, with the new values at the middle of the continuum domain, the fluctuating hydrodynamic solver will advance for a certain number of continuum time steps which are considered as a single coupling time step.

The new values for the density, velocity, and temperature (and also pressure and other dependent parameters) will allow us to re-calculate the required thermodynamic force and advance the particle-based solver and repeat the coupling procedure as long as the results converge. This coupling algorithm is illustrated in Fig.5.11.

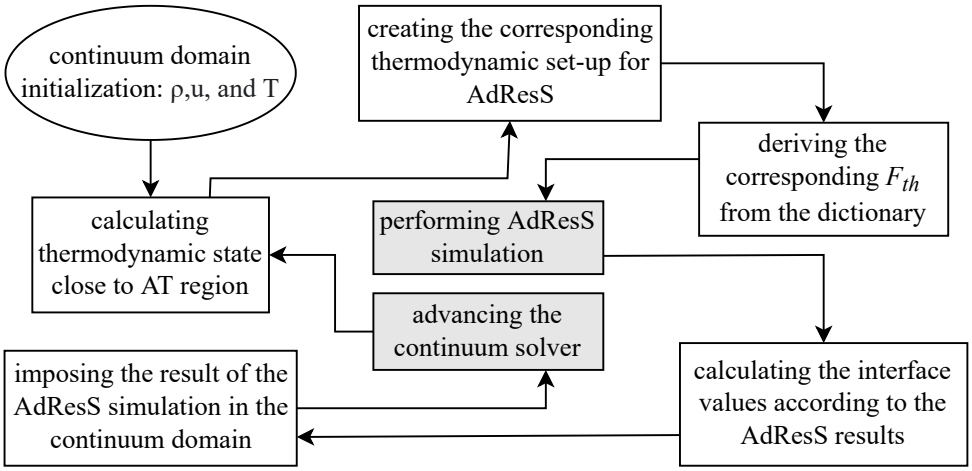


Figure 5.11: The coupling algorithm for coupling AdResS to fluctuating hydrodynamic.

5.6 Numerical results: coupling AdResS to FHD

The coupling of AdResS to the previously presented fluctuation hydrodynamic solver has been done for the configuration illustrated in Fig.5.12 and by the algorithm explained in the previous section and pictorially illustrated in Fig.5.10 and Fig.5.11. In the following, the technical details of the simulation with the necessary information on fluid properties will be explained. Next, results for various test scenarios will be presented.

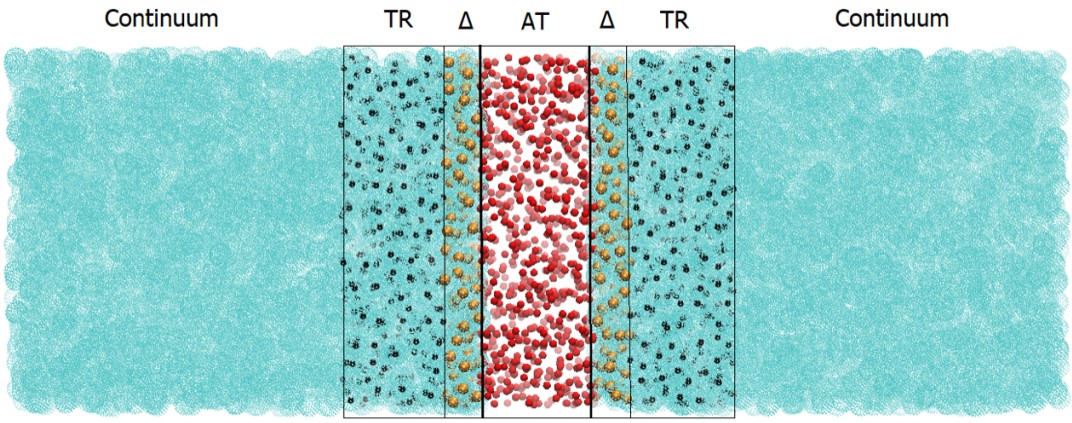


Figure 5.12: Atomistic subdomain (red particles) embedded in a continuum hydrodynamic domain (blue region) through an interface region (indicated with orange and black points corresponding to Δ and TR regions of AdResS) which overlaps with the blue region. Each solver (namely continuum and molecular dynamics) provides an interface boundary for the other one. This information is exchanged among the regions during the simulation.

5.6.1 Technical details

To couple the molecular resolution to the continuum hydrodynamics, we have considered Argon liquid as a single atomic simple fluid with the relative simplicity of the spherical two-body pair potential and the amenability to the basic theory. The simulation domain is decomposed as illustrated in Fig.5.12 into particle and continuum subdomains and their interface is considered to be at the border of the AT and Δ regions of AdResS. The size of the AdResS box is $75\sigma \times 15\sigma \times 15\sigma$ in which the length of AT region is 25σ and the tracer and Δ regions' size are 21σ and 4σ, respectively. The mass of each Argon particle is $6.6335209 \times 10^{-26} kg$, the σ value for Argon is $0.34[nm]$ and the ϵ value is $120k_B$ with k_B being the Boltzmann constant. For calculating the density profile as a function of length by Fourier transformation, the box is divided into 750 slices along the axis of change of resolution. In all simulations, the MD time step is 0.002 in reduced units which is equivalent to almost $4.3[fs]$ ($t = t^* \sqrt{m\sigma^2/\epsilon}$) and each AdResS simulation consists of 2.5×10^6 steps which 30% of them are considered for equilibration and the rest for data production.

The AT region of AdResS is designed to be equivalent to several continuum cells and in all reported results it includes 10 continuum cells which means that the space discretization size (dx) for the continuum domain would be $AT/10 = 2.5\sigma = 0.85[nm]$ while the whole domain size is $10AT = 250\sigma = 85[nm]$. The stability conditions presented in Eq.5.19 restrict us to choosing a very small time step for the continuum solver which in our case is $0.1[ps]$.

Fluid state

In the preparation of the fluctuating hydrodynamic solver for the compressible flow, the fluid under study was considered a dilute gas with ideal gas properties and the equation of state $P = \rho RT$ (and consequently straightforward relations for thermodynamic properties) which is used within numerics for solving Navier-Stokes equations. Fig.5.13 shows the phase diagram for the Lennard-Jones fluid in the reduced unit in the $\rho - T$ diagram. Choosing a dilute gas means having a relatively low temperature while being at very low densities. However, it is not possible to do the AdResS simulations for such low-density fluids as a key point in AdResS is to calculate the density profile and approximate the required thermodynamic force based on the deviation of the resulting density from the target one. This is because in dilute gases the particles are less than the required number of particles for filling the domain and producing a uniform density profile without unphysical fluctuations.

On the other hand, being in the supercritical region ensures us to be in the safe region without phase change and coexistence region of liquid and gas but it requires finding a suitable equation of state (and other thermodynamic properties) for the fluctuating hydrodynamic numerical solver. A supercritical fluid is any substance at a higher temperature and pressure than its critical point in which a distinction between gas and liquid phase does not exist anymore and has both gas and liquid-like properties. It is gas-like as it is a compressible fluid that can fill its container and is liquid-like as it has comparable density and solvating power.

Finally, we have considered a supercritical state far from the critical point at the reduced temperature of 2.5 and reduced density of 0.572 which corresponds to a reduced pressure of 2.37. These reduced values for the selected state correspond to the following SI units for Argon: $\rho = 964.82[kg/m^3]$, $T = 300[K]$, and $P = 100[MPa]$. This density means having 9647 Argon particles in the AdResS simulation domain for the specified box size of $75\sigma \times 15\sigma \times 15\sigma$.

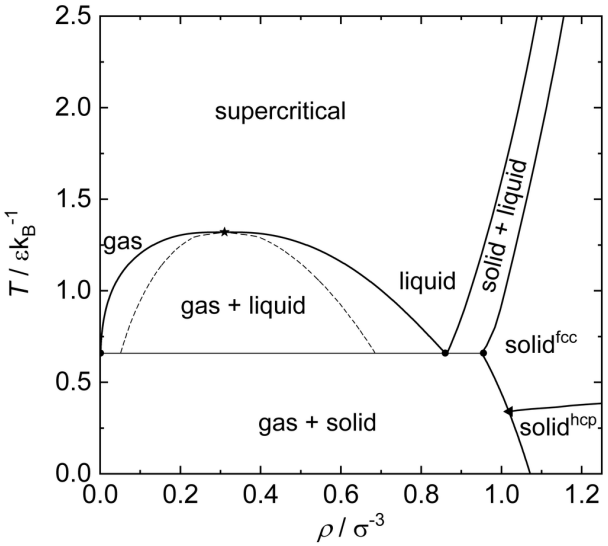


Figure 5.13: Phase diagram of Lennard-Jones substances in $\rho - T$ diagram [87].

Equation of state and thermodynamic properties

The equation of state for the Lennard-Jones fluid can be determined by doing several MD simulations and finding a correlation between pressure as a function of density and temperature. There are several methods for deriving the equation of state of Lennard-Jones which in general can be categorized into two groups: those with a theoretical basis (semi-theoretical) and those with a purely empirical basis, each of them applies to some ranges of densities and temperatures [104]. The Modified Benedict-Webb-Rubin equation of state used in this work is the one used by Nicolas et al. [140] and explained in detail in Ref. [104] and contains 32 linear and one non-linear parameter. It starts from writing the Helmholtz free energy, separating its residual and ideal parts, and deriving its pressure relation. It results in the following relation for the pressure in reduced units:

$$P^* = \rho^* T^* + \sum_{i=1}^{8} a_i \rho^{*(i+1)} + F \sum_{i=1}^{6} B_i \rho^{*(2i+1)} \tag{5.20}$$

where the coefficients a_i and b_i are functions of temperature only and are represented in Tab.5.1 which contain 32 linear parameters in the Modified Benedict-Webb-Rubin equation of state (presented in Tab.5.2). In Eq.5.20, $F = exp(-\lambda\rho^{*2})$ and λ is a nonlinear adjustable parameter and is reported to be equal to 3.

Thus, by using the Eq.5.20 it is possible to calculate the reduced pressure at each density and temperature around the state that the simulation is being performed and convert it to the SI unit by $P = P^*\epsilon/\sigma^3$.

The thermodynamic properties of the fluid depend on the temperature and density of the fluid. This includes the heat capacity (c_v), viscosity (μ), and thermal conductivity (k) of the fluid that for Argon can be found at different state points in the source data at the National Institute of Standards and Technology (NIST) [124]. In this work, we have found and stored the values of heat capacity, viscosity, and thermal conductivity of Argon at 3 different temperatures and 11 different densities around the selected state ($\rho = 964.82[kg/m^3]$ and $T = 300[K]$) within a range of $\pm\%15$ and interpolated those values for any other situations.

i	a_i	b_i
1	$x_1T^* + x_2\sqrt{T^*} + x_3 + x_4/T^* + x_5/T^{*2}$	$x_{20}/T^{*2} + x_{21}/T^{*3}$
2	$x_6T^* + x_7 + x_8/T^* + x_9/T^{*2}$	$x_{22}/T^{*2} + x_{23}/T^{*4}$
3	$x_{10}T^* + x_{11} + x_{12}/T^*$	$x_{24}/T^{*2} + x_{25}/T^{*3}$
4	x_{13}	$x_{26}/T^{*2} + x_{27}/T^{*4}$
5	$x_{14}/T^* + x_{15}/T^{*2}$	$x_{28}/T^{*2} + x_{29}/T^{*3}$
6	x_{16}/T^*	$x_{30}/T^{*2} + x_{31}/T^{*3} + x_{32}/T^{*4}$
7	$x_{17}/T^* + x_{18}/T^{*2}$	
8	x_{19}/T^{*2}	

Table 5.1: The a_i and b_i temperature-dependent coefficients for the pressure equation of Lennard-Jones fluid.

i	x_i	i	x_i	i	x_i	i	x_i
1	0.86309	9	2798.29177	17	63.98608	25	−113.16076
2	2.97622	10	−0.04839	18	16.03994	26	−8867.77154
3	−8.40223	11	0.99633	19	68.05917	27	−39.86983
4	0.10541	12	−36.98000	20	−2791.29358	28	−4689.27030
5	−0.85646	13	20.84012	21	−6.245128	29	259.35353
6	1.58276	14	83.05402	22	−8116.83610	30	−2694.52359
7	0.76394	15	−957.47997	23	14.88736	31	−721.84876
8	1.75317	16	−147.77462	24	−10593.46755	32	172.18021

Table 5.2: The parameters for the Modified Benedict-Webb-Rubin equation of state regressed in Ref. [104].

Non-equilibrium problem in AdResS

While coupling the fluctuating hydrodynamic solver to the AdResS subdomain, it is possible to have different densities and temperatures at the left and right sides of the particle subdomain in the neighbour cells in Fig.5.10 which turns the simulation into a non-equilibrium-like problem within the AdResS domain. Recently, Ebrahimi Viand et al. [57] performed AdResS simulations for a non-equilibrium problem with different temperatures and densities at the left and right reservoirs. They studied the analogy of AdResS and Bergmann-Lebowitz model of open systems [21, 123] and performed the AdResS simulation for the non-equilibrium problem by applying the thermodynamic forces for AdResS simulation of the equilibrium problems corresponding to each reservoir.

In their analogy, when the AdResS system is in contact with two different reservoirs with different thermodynamic conditions, according to the Bergmann-Lebowitz model, once the system is in contact with both reservoirs at the same time, the combined effect translates to the combined thermodynamic forces for each side. In our case, this means that for running a non-equilibrium problem with different thermodynamics (density and temperature) at the left and right sides of the AdResS simulation, one can calculate two different thermodynamic forces, one for the left and one for the right sides of the domain at equilibrium conditions corresponding to each reservoir's thermodynamic condition. Next, add each of those resulting thermodynamic forces one to the left and one to the right sides simultaneously, while setting the thermostat of the left and right sides according to the temperature of the reservoirs. This will result in the non-equilibrium scenario in the AdResS set-up with two different thermodynamic state points on the left and right sides.

5.6.2 Thermodynamic force dictionary

By applying the idea of non-equilibrium simulation with AdResS [57], it is possible to implement a non-equilibrium problem in AdResS by using pre-calculated thermodynamic forces of the equilibrium situations with different reservoirs. This brings the idea of preparing a list of thermodynamic forces for a range of densities and temperatures around the state that we are interested in and using the list for obtaining the proper thermodynamic force for any other situation by interpolation.

As this work intends to develop a coupling set-up for problems with varying densities and temperatures, the AdResS simulator should be prepared for those conditions. This means that it is required to prepare a list of thermodynamic forces called the "dictionary of thermodynamic forces" for different thermodynamic equilibrium situations around the state of the fluid under study. Thus, with the selected state of Argon at the density of $964.82[kg/m^3]$ and temperature of $300[K]$, here a set of AdResS simulations for three different temperatures in the range of $\pm\%15$ and eleven different densities in the range of $\pm\%15$ around the target state have been done and the thermodynamic forces for the corresponding equilibrium cases are calculated. The potential of thermodynamic force for the abovementioned list of densities and temperatures have been calculated and presented in Fig.5.14.

According to Fig.5.14, a system with higher density which means a higher number of particles needs a larger amount of energy to be equilibrated and reproduce the results of the full atomistic simulation of reference. On the other hand, it needs more equilibration energy (energy provided by thermodynamic force) at higher temperatures to generate the flat density at equilibrium. In other words, as shown in the previous chapter [74], the role of thermodynamic force is to flatten the pressure profile by its definition and with higher temperatures, one expects to have higher pressure differences in the AdResS domain before applying a thermodynamic force which needs to be compensated with higher external potential.

5.6.3 AdResS validation

In the procedure of coupling AdResS to fluctuating hydrodynamics simulator, it is essential to ensure that the multi-resolution particle-based solver is properly modelling the fluid under study in different situations. This means that the adaptive resolution scheme is supposed to reproduce the results of the fully atomistic system with the same parameters in and out of equilibrium within an acceptable range of error.

Equilibrium

In the case of the thermal and hydrodynamic equilibrium where there is no pressure and temperature difference between different subregions of the particle subdomain, one expects to get the same equilibrium results for the AdResS and full atomistic simulation of reference. Apart from the flat density and temperature profiles for both simulations (that means a flat pressure as well) which has been shown in all previous studies [73, 74], one expects to get the same physical behaviour. The quantities that describe the physical state of the system are the particles' probability distribution in the region of interest ($p(N)$) and the radial distribution function ($g(r)$). As an instance for the current study, the comparison of particles' number probability distribution, radial distribution function, and density profile at equilibrium for the Argon fluid is shown in

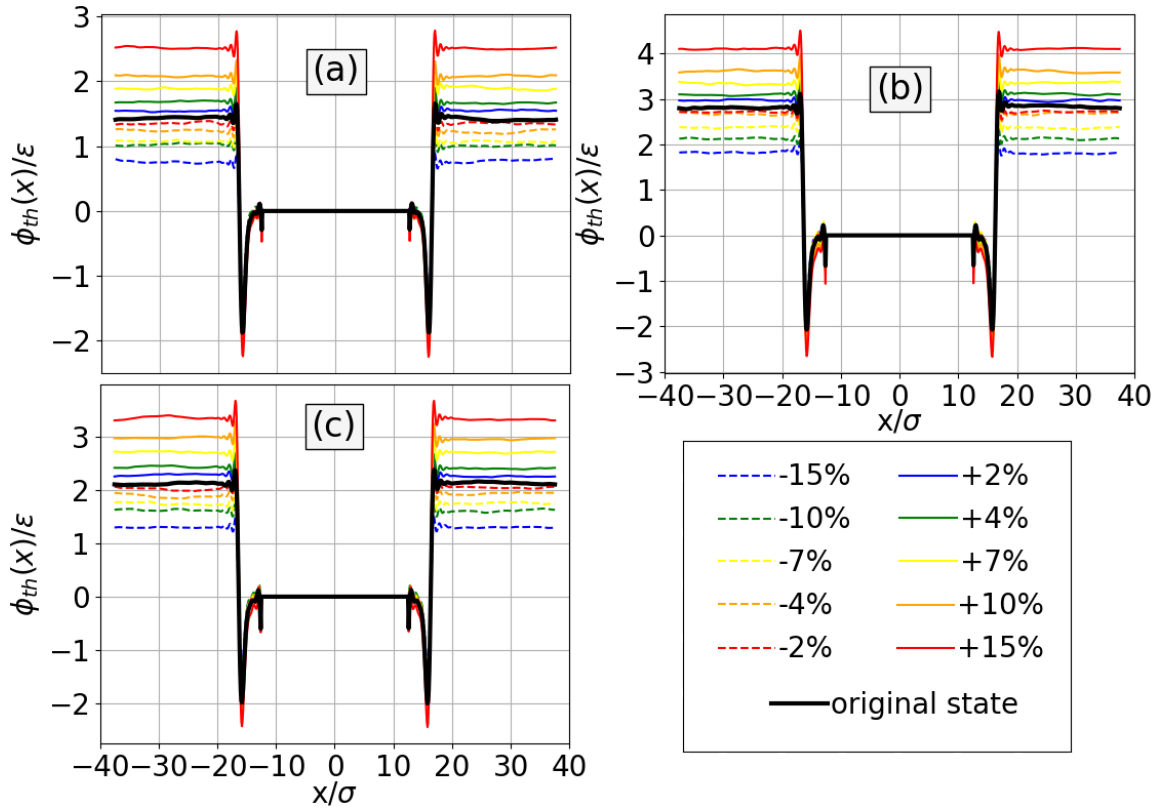


Figure 5.14: The potential of thermodynamic force for 3 different temperatures of 255$[K]$(a), 345$[K]$(b), and 300$[K]$(c) in 11 different densities that have been calculated by iterative manner for AdResS simulations at equilibrium. The range of densities and temperatures covers $\pm$%15 around the target state ($\rho^* = 0.57$ and $T^* = 2.5$). The black solid line shows the potential of thermodynamic force for the target density at each temperature. The coloured solid and dashed lines represent the cases with a density higher and lower than the target state, respectively. In each set, densities increase in the order of colours such as blue, green, yellow, orange, and red.

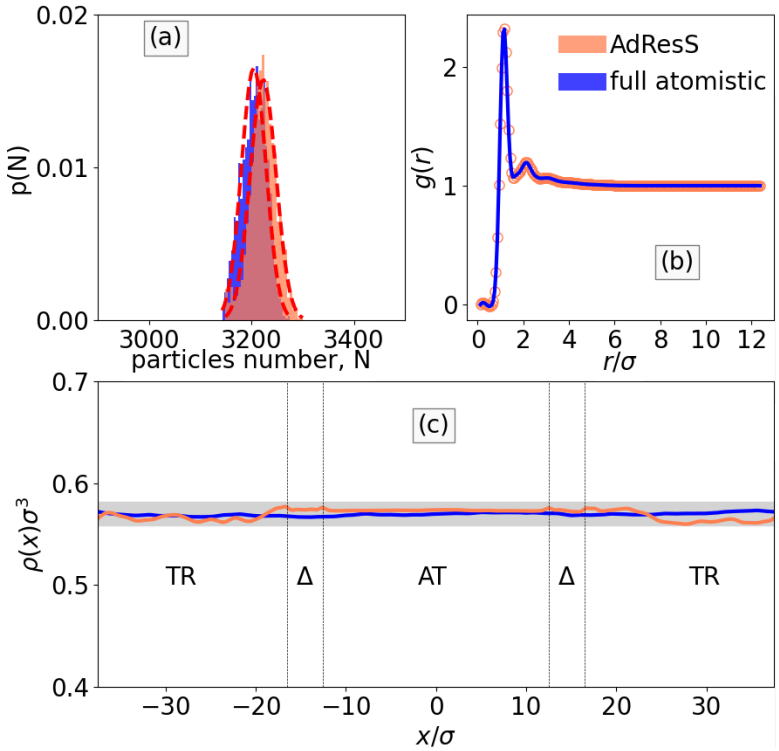


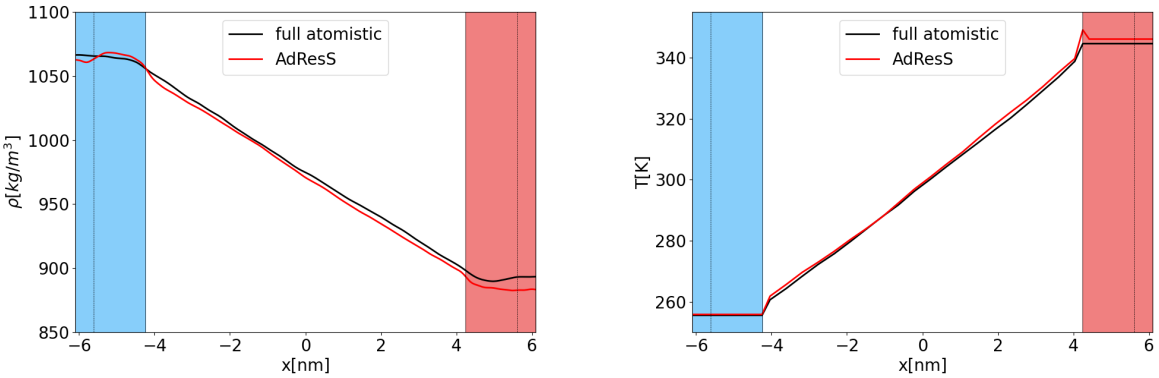
Figure 5.15: Comparison of the probability distribution of particles (a) in the region of interest (AT), radial distribution function (b), and density profile (c) for the AdResS and full atomistic simulation of reference in the equilibrium state of supercritical Argon at $\rho^* = 0.57$ and $T^* = 2.5$.

Fig.5.15 at the state described in technical details ($\rho = 964.82[kg/m^3]$ and $T = 300[K]$) in a box with the dimensions of $75\sigma \times 15\sigma \times 15\sigma$ where the AT region with the size of 25σ is located in the middle of the box.

Non-equilibrium

In the non-equilibrium case, where the left and right sides of the region of interest (AT) are at different thermodynamic state points, AdResS should follow the thermodynamic behaviour of the reference full atomistic simulation. This equivalence has been shown in [57] where the Bergmann-Lebowitz model [21, 123] implies that applying different thermodynamic forces corresponding to the respective equilibrium states will yield the desired non-equilibrium results. In coupling the fluctuating hydrodynamics solver with the adaptive resolution simulator, the non-equilibrium case with different thermodynamic states resulting from the continuum solver at the left and right reservoirs is the most likely scenario happening over the simulation time that leads to a non-equilibrium problem is AdResS.

Here, the most drastic scenario for the temperature gradient in this study is considered where the cold reservoir is at the temperature of $T_{cold} = 255[K](T^*_{cold} = 2.125)$ and the hot reservoir is at the temperature of $T_{hot} = 345[K](T^*_{hot} = 2.875)$. Then, the density of the left and right sides' reservoirs may be calculated in the left and right reservoirs by running a full atomistic simulation with specified temperatures at the corresponding regions which means the densities that yield to $P(\rho_{cold}, T_{cold}) = P(\rho_{hot}, T_{hot})$. However, here, to consider uncertainties due to the use of the empirical equation of state (Eq.5.20), the corresponding densities at the cold and hot reservoirs are calculated by using Eq.5.20 in such a way that the pressure remains constant that



(a) Density profile

(b) Temperature profile

Figure 5.16: The profile of density (a) and temperature (b) for the non-equilibrium problem for AdResS and full atomistic simulation of reference where the left and right sides' reservoirs are at different temperatures with $\rho_{left} = 1073.2[kg/m^3]$, $T_{left} = 255[K]$, $\rho_{right} = 884.5[kg/m^3]$, and $T_{right} = 345[K]$ in such a way that $P_{left} = P_{right}$ according to the equation of state of Lennard-Jones fluid Eq.5.20. The white region in the middle represents the AT region of interest which is connected to the cold (blue) and hot (red) reservoirs through a transition region specified by the dashed line. The proper thermodynamic force for the left and right sides in the AdResS set-up is calculated by interpolation of those in the pre-calculated dictionary of thermodynamic force shown in Fig.5.14.

are: $\rho_{hot} = 884.5[kg/m^3](\rho^* = 0.52)$ and $\rho_{cold} = 1073.2[kg/m^3](\rho^* = 0.64)$.

The simulation protocol reads that the thermodynamic force is calculated separately for the two abovementioned thermodynamic state points at equilibrium and then the left and right sides' reservoirs are set to the mentioned thermodynamic states. Next, the corresponding thermodynamic force related to each state is applied to the reservoir regions. The results of the temperature and density gradient for the described non-equilibrium case are shown in Fig.5.16.

5.6.4 Numerical tests

In previous sections, the fluctuating hydrodynamics code and the adaptive resolution simulation in and out of equilibrium were assessed. In this section, a set of test scenarios are designed to evaluate the application of coupling adaptive resolution simulation to fluctuating hydrodynamics. These include a flat, step function-like, and sinusoidal initial conditions and finally a quasi-1D geometry with varying cross-section.

Initial flat conditios

Initial constant properties (density, velocity, temperature, and pressure) are applied as an initial condition to the AdResS-FHD coupling system. As there are no advective and diffusive forces in the domain with the NS solver(see Eq.5.11), it is expected that with the deterministic Navier-Stokes solver, nothing should change during continuum simulation, but after coupling it to AdResS some fluctuations will arise in the system because of the inevitable deviations of AdResS from the exact target state (less than 2 per cent). However, with the fluctuating hydrodynamics coupling to AdResS, it is expected to have fluctuations in the fluid behaviour

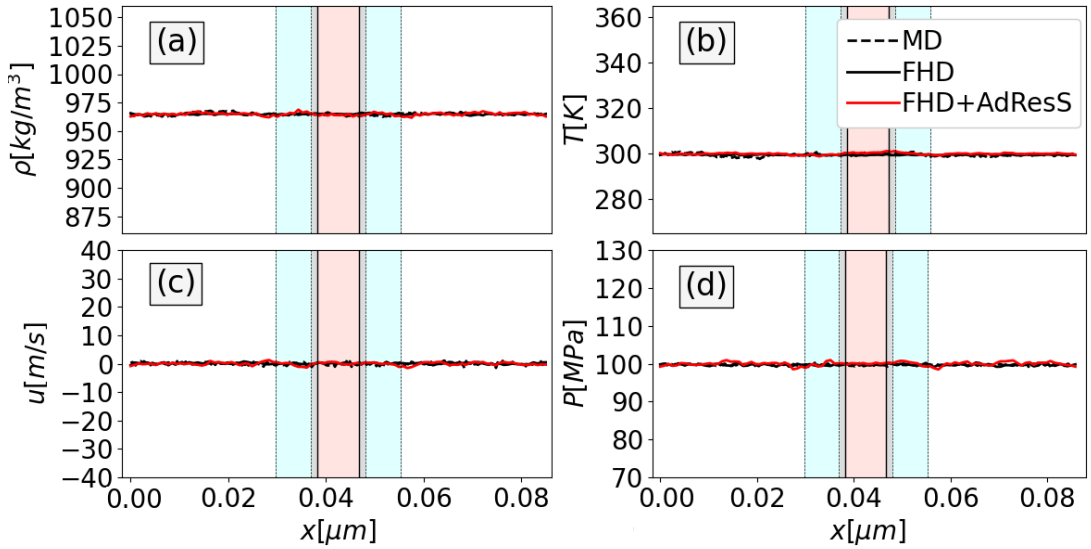


Figure 5.17: The profile of density (a), temperature (b), velocity (c), and pressure (d) for coupling AdResS to FHD with initial flat properties. The red-shaded subregion shows the AT region of AdResS which is connected to the reservoir of non-interacting particles with blue-shaded colour (TR) through a grey transition region (Δ) where the vertical solid line between AT and Δ region represents the interface of MD and continuum subdomains. In all figures, the black solid and dashed lines show the result of the FHD and MD solvers, respectively, and the red solid line represents the result of coupling FHD to AdResS.

that results in a fluctuating domain around the target state of the fluid. In Fig.5.17, the profile of thermodynamic and hydrodynamic properties of the system at some arbitrary time is shown for fully atomistic, fluctuating hydrodynamics, and AdResS-FHD coupling systems. The initial state with zero velocity is set to the described state $\rho = 964.82[kg/m^3]$ and $T = 300[K]$ which corresponds to the pressure of $100[MPa]$ according to Eq.5.20.

Initial step function

An initial step function for the density and temperature where the discontinuity occurs in the middle of the simulation domain may assess the AdResS-FHD coupling system and see whether the particle-based solver can tolerate such a drastic situation while interacting with the continuum solver. In this case, initial discontinuous temperature and density are applied to the system in such a way that the initial pressure of the fluid remains constant all over the simulation domain. This means that the continuum cells in the left and right sides of the domain will have initial different values that obey the relation $P(\rho_{left}, T_{left}) = P(\rho_{right}, T_{right})$ according to the equation of state. This will lead to an extreme initial situation for AdResS where the left and right sides' reservoirs (TR regions) are at different thermodynamic states as those presented in Fig.5.16.

For this purpose, the initial density and temperature of the left side are set to $\rho_{left} = 1010[kg/m^3]$ and $T_{left} = 279.3[K]$ and for the right side they are $\rho_{right} = 919.6[kg/m^3]$ and $T_{right} = 322.3[K]$ while the initial pressure is $P = 100[MPa]$. One expects that such initial perturbation should resolve in the domain and get flat properties after reaching equilibrium which needs a smooth change of properties over time from a discontinuous situation to a flat condition while the pressure remains constant. The evolution of the density and temperature profiles over the simulation time until reaching the equilibrium is shown in Fig.5.18 for the whole domain including AdResS

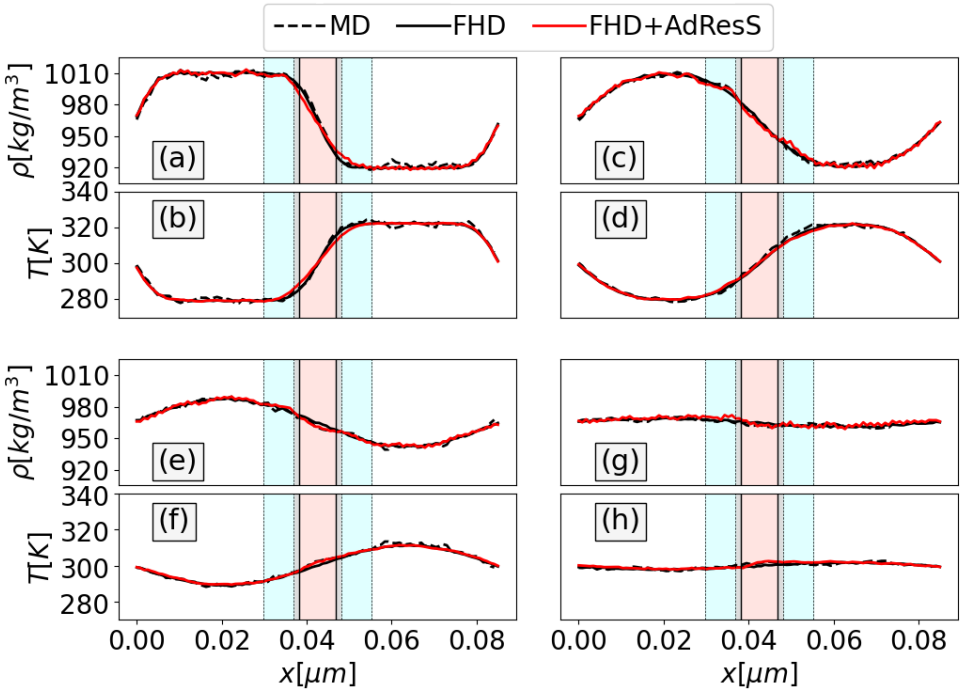


Figure 5.18: The profile of density (a, c, e, and g) and temperature (b, d, f, and h) of fluid with initial step function for density and temperature ($\rho_{left} = 1010[kg/m^3]$, $T_{left} = 279.3[K]$, $\rho_{right} = 919.6[kg/m^3]$, and $T_{right} = 322.3[K]$) with constant pressure ($P = 100[MPa]$) overtime at $t = 0.05[ns]$(a and b), $t = 0.25[ns]$(c and d), $t = 1.2[ns]$(e and f), and $t = 3.5[ns]$(g and h). The coloured regions in the middle of the box is showing the AdResS domain where the atomistic region (red area) is in contact with the TR region (blue area) through a small transition region (gray area). The black solid and dashed lines show the result of reference FHD and MD simulations, respectively, and the red line represents the results of coupling AdResS to FHD.

and continuum subdomains.

As previously mentioned in the algorithm section, the thermodynamic properties of the fluid in the particle subdomain during continuum simulation will be replaced with the quantities in AT region of AdResS. Thus, as illustrated in Fig.5.18 the AdResS density and temperature will change over time with the changing boundary conditions at the left and right neighbour cells of the continuum solver. The details of the density and temperature in AdResS simulation for the corresponding snapshots represented in Fig.5.18 are shown in Fig.5.19.

Acoustic wave

In this case, similar to the previous case, an acoustic wave with periodic initial conditions for density and temperature is applied to the system, but with a non-constant pressure profile according to the equation of state. Here, as the initial pressure function obeys a periodic behaviour, the flow will have some oscillations until reaching the equilibrium state (flat properties). This example shows how smoothly such oscillations will be handled with the developed coupling code while regenerating the results of the pure continuum and/or fully atomistic.

Here, the initial density oscillates around $\rho = 964.82[kg/m^3]$ with an amplitude of $\sim 30[kg/m^3]$ and frequency of $4\pi/l$ where l is the simulation domain length. On the other hand, the tem-

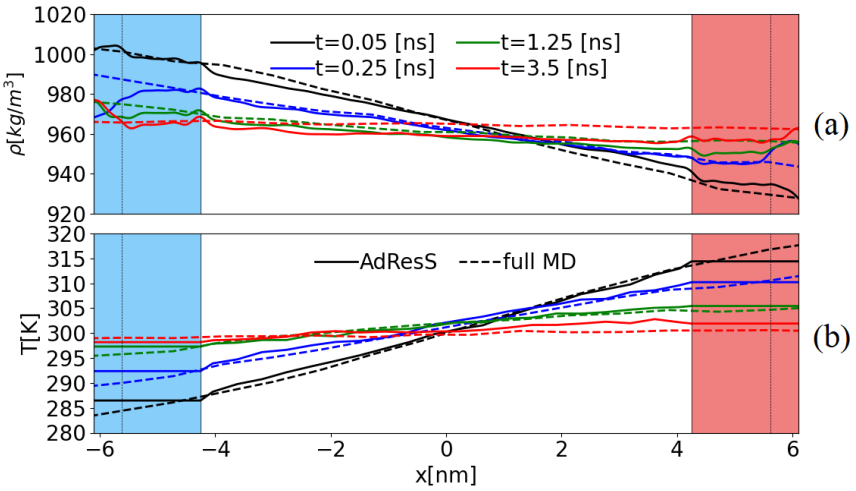


Figure 5.19: The profile of density (a) and temperature (b) in the AdResS domain and the full atomistic simulation over time, for the initial step function for the density and temperature. The red and blue regions represent the hot and cold reservoirs, respectively. The data correspond to the points in time of Fig.5.18.

perature initial function is set similarly with an oscillation amplitude of $\sim 40[K]$. One expects that these initial perturbations should resolve in the whole simulation domain over time and the system should equilibrate at $\rho = 964.82[kg/m^3]$ and $T = 300[K]$ after some oscillations. The results of this case are presented in Fig.5.20 over time. The profile of density and temperature of AdResS simulation for the time steps recorded in Fig.5.20 is shown in Fig.5.21.

Quasi-1D varying cross-section geometry

In this part, to assess the capability of the AdResS-continuum coupling code for non-constant cross-sections of the tube, a new geometry with varying cross-sections is designed. Such geometry with changing cross-section will generate a changing density profile and consequently pressure domain along the length of the box. The cross-section of the new geometry linearly increases in the continuum domain on the left and symmetrically decreases on the right side while having a constant value at the middle part corresponding to the AT region of the AdResS domain.

Applying such conditions to the continuum solver needs some corrections in the conservation equation set and Navier-Stokes equation discretization as the cell's cross-section area is not constant but is a function of the length x. This requires the addition of a factor of $A(x)$ to all terms in Eq.5.11 which leads to the same factor in all terms in the discretization algorithm of Eq.5.16. Moreover, this change will add a source term to the momentum equation of the Navier-Stokes equation set as below.

$$A\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ J\\ E\end{pmatrix} = -\frac{\partial}{\partial x}\begin{pmatrix}\rho uA\\ (\rho u^2+P)A\\ (E+P)uA\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}0\\ \frac{4}{3}\eta A\partial_x u\\ \frac{4}{3}\eta uA\partial_x u + kA\partial_x T\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}0\\ sA\\ (q+us)A\end{pmatrix}+P\frac{\partial}{\partial x}\begin{pmatrix}0\\ A\\ 0\end{pmatrix} \quad (5.21)$$

The cross-section area in the middle is similar to previous cases $A_{middle} = (15\sigma)^2$ and its value at the left and right borders of the domain is $A_{left} = A_{right} = (12.25\sigma)^2$. The result of the AdResS-continuum simulation is shown for density, temperature, and velocity in Fig.5.22 over time. The new solver with coupling AdResS to fluctuating hydrodynamics code is calculating the evolution of the system correctly until reaching equilibrium.

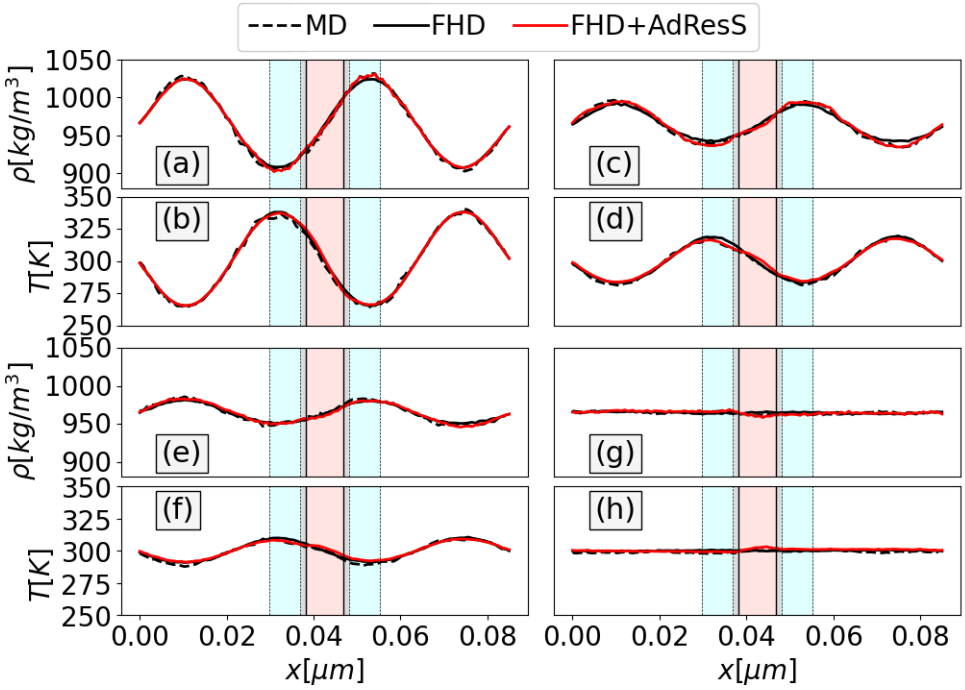


Figure 5.20: The profile of density (a, c, e, and g) and temperature (b, d, f, and h) of fluid with initial sinusoidal function for density and temperature overtime at $t = 0.0[ns]$ (a and b), $t = 0.25[ns]$ (c and d), $t = 0.45[ns]$ (e and f), and $t = 1.5[ns]$ (g and h). The initial density function is $\rho(x) = 964.82(1 + 0.03sin(4\pi x/l))$ and the temperature is $T(x) = 300(1 - 0.13sin(4\pi x/l))$. The solid black line, dashed black line, and red line represent the results of FHD, MD, and coupling of AdResS to FHD, respectively.

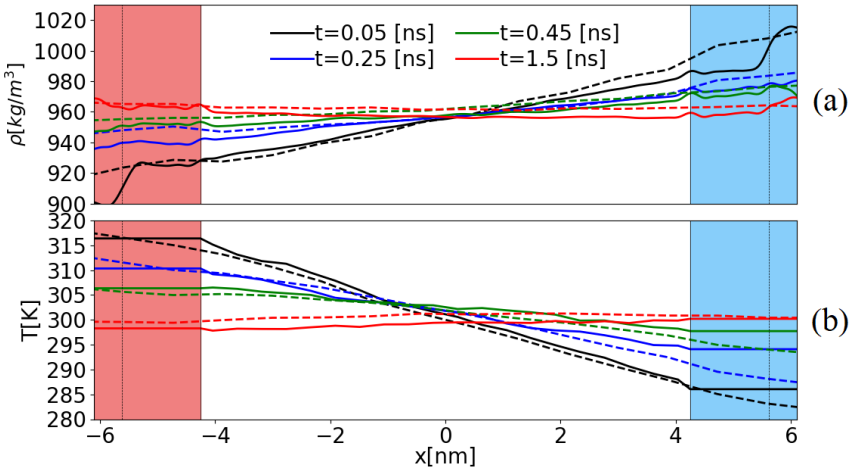


Figure 5.21: The profiles of density (a) and temperature (b) in the AdResS domain over time for the initial periodic density and temperature during coupling AdResS to FHD. The snapshots correspond to the times in Fig.5.20 and the order is illustrated in the labels. The regions with red and blue colours represent the initially hot and cold reservoirs, respectively.

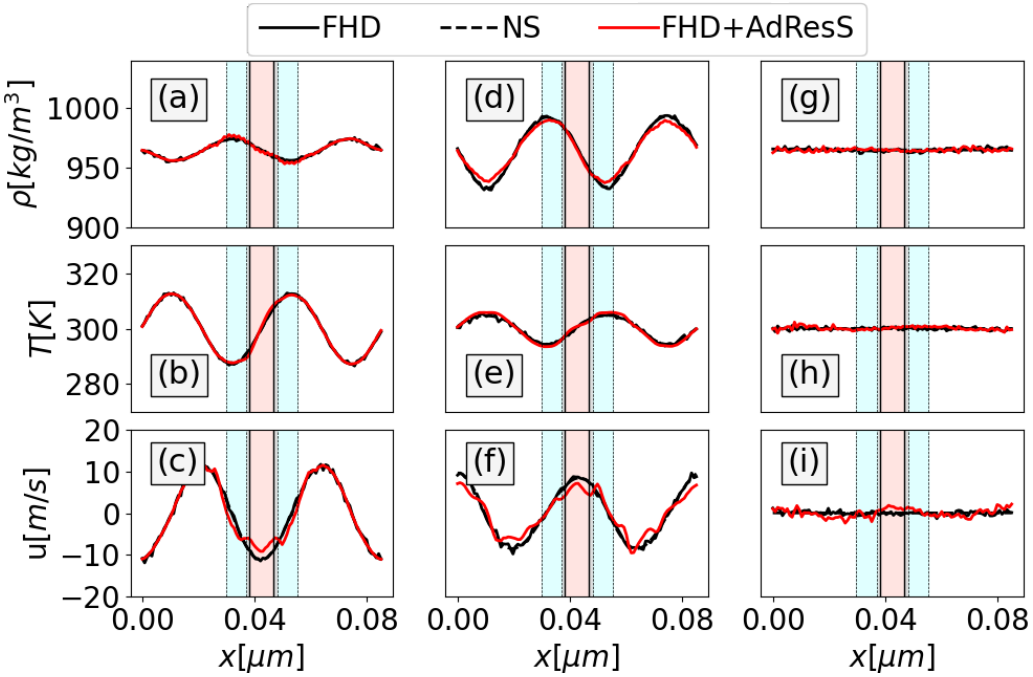


Figure 5.22: The evolution of density (a, d, and g), temperature (b, e, and h), and flow velocity (c, f, and i) for the case with varying cross-section where it increases and decreases linearly in the left and right continuum subdomains, respectively, while being constant in the middle at $t = 0.01[ns]$ (a, b, and c), $t = 0.1[ns]$ (d, e, and f), and $t = 10[ns]$ (g, h, and i). The coupling simulation is started with an arbitrary initial condition which is set to an initial uniform density and sinusoidal temperature with an oscillation amplitude of $15[K]$ around the target state.

5.7 Conclusion

Adaptive Resolution Scheme (AdResS) is re-generating the same thermodynamic behaviour of the reference full-atomistic system with a smooth change of resolution from a small enough fully atomistic subdomain to a reservoir of non-interacting particles by using a thermostat applied to the reservoir region and implementing an external force (thermodynamic force). As the non-equilibrium situation is the most probable case while doing AdResS simulation coupled to a continuum solver, the possibility of simulating non-equilibrium scenarios with the use of AdResS by employing the Bergmann-Lebowitz model is studied and proved. On the other hand, adding a stochastic flux term to the compressible Navier-Stokes equations makes it able to capture fluctuations that have crucial importance in complex systems with atomic resolutions.

To further develop the idea of coupling different regions with different resolutions, in this work, the AdResS domain is coupled to the fluctuating hydrodynamic solver with the use of a pre-calculated list of thermodynamic forces for information exchange purposes. The results for different test scenarios show satisfactory correspondence for the thermodynamic properties within an acceptable range of error.

To make the AdResS-FHD coupling solver more general, it is possible to extend the dictionary of thermodynamic force for a wider range of thermodynamic state points. For the moment, the AdResS domain is a three-dimensional region while exchanging information with the continuum domain in one direction as the continuum solver is one-dimensional. It is possible to extend the

continuum solver for multi-dimension domains to make it possible to study more generic flow situations.

Chapter 6

Summary and outlook

In this thesis, we have studied Adaptive Resolution Simulation (AdResS) as a tool for modelling open systems with a multiscale approach. The fundamental statistical mechanics of the AdResS technique is considered to derive a mathematical and physical framework to analyse the open system and relate the macroscopic behaviour of the system to its microscopic details. These fundamental characteristics include the chemical potential of the system and its pressure and their relation to the reference system. Later, by using the findings and foundations developed in this work, the AdResS domain is coupled to a developed continuum solver to control the environment and reservoir conditions within a macroscale resolution during a continuum simulation. In the following, the work for each section is summarized and the connection between them is explained.

AdResS is a multi-resolution method for performing molecular modelling on different scales to reduce the computational costs required for fully atomistic simulations. In this method, part of the simulation domain is considered with atomistic resolution and the rest is filled with point-like reservoir particles which is connected to the atomistic region through a small hybrid region. To recreate the behaviour of the reference full atomistic simulation and compensate for the removed degrees of freedom, it is required to implement some additional tools in the AdResS system. These tools include a thermostat and an external force called thermodynamic force applied to the reservoir region to mimic the thermodynamics and structure of the reference system. The thermodynamic force is calculated self-consistently in an iterative manner during equilibrium runs. As the particles in the reservoir region do not interact with each other, they may experience unphysical forces while entering the open system with the atomistic resolution; thus, it is required to normalize those unphysical forces during simulation. However, this correction does not change the thermodynamics of the system at the statistical mechanics level and this fact is shown in this work.

As the first step after the preparation of the AdResS set-up, the chemical potential of AdResS and its reference simulation are related to each other through a rigorous mathematical approach. This is done by starting from the equivalence of the grand potential of open system in AdResS and reference set-ups and expanding the mathematical relations by the physical consistency. The developed equation for the chemical potential relates the change of the chemical potential of the system while changing the resolutions to the applied corrections for AdResS simulations. Thus, the difference between the chemical potential of the AdResS set-up and its reference simulation is due to the loss of interactions with the reservoir region and also the implementation of the thermodynamic force in the reservoir region of AdResS. The developed relation for the chemical potential is numerically assessed for the Lennard-Jones (LJ) fluid at four different thermodynamic state points and also water as a complex liquid and showed satisfactory results. This research, in addition to providing a straightforward approach for the calculation of chemical potential in AdResS, further supports the idea of open system treatment for AdResS from the

first principles.

Later, a complementary research for the previous step for the calculation of pressure in the AdResS domain is done. The derivation of the chemical potential in the AdResS set-up started with the idea of equivalence for the grand potential of the open systems in the AdResS and its reference simulations. However, to explore the grand potential in detail, it is required to calculate the pressure in the AdResS domain as an inhomogeneous system. The pressure in molecular simulations with homogeneous nature and the absence of external force fields can be calculated by the virial relation. Here, we have utilized a new approach for the calculation of pressure by separating the kinetic and interaction contributions of the pressure that can be extended to the inhomogeneous systems. In this approach, a sliding plane is considered in the simulation domain and the kinetic contribution of the particles crossing the plane is added to the interaction contribution of particles on the opposite sides of the plane to calculate the normal and transverse components of the stress tensor. The mathematical and physical derivations of this research are numerically analysed for the four LJ fluids with different thermodynamic state points explored in the previous stage for the chemical potential. The results of this research support the idea of grand potential's equivalence between AdResS and the reference simulation. Moreover, the point-by-point calculation for the pressure in the hybrid region corresponds to the potential energy provided to the system by the thermodynamic force. This latter shows that the thermodynamic force is a tool to compensate for the pressure difference resulting from the abrupt change of resolutions and thus equivalence of the grand potentials for the AdResS and its reference set-ups.

Apart from the AdResS methodology, the continuum simulation for incorporating microscale fluctuations in the continuum solver is designed. Microscale fluctuations are always deviating a system from staying in a constant state even at equilibrium. In this sense, Landau and Lifshitz introduced a stochastic flux term in the Navier-Stokes equations for the simulation of compressible flow. Here, a new Fluctuating Hydrodynamics (FHD) code is developed to simulate the Landau-Lifshitz Navier-Stokes equations to incorporate the fluctuations into the deterministic continuum slover. For the discretization of the equations, a third-order Runge Kutta method is used for time integration with a central finite difference method for space discretization. The results of this new solver are presented for different initial conditions for a one-dimensional continuum domain and showed satisfactory correspondence to the expected behaviour from continuum simulations.

To further develop the idea of the multiscale simulator and have a reasonable approximation of the flow behaviour in the reservoir region outside the atomistic subdomain, the AdResS simulation is coupled to the fluctuating hydrodynamics solver. This will make the AdResS method able to capture the flow behaviour in the reservoir region far from the atomistic resolution while including the atomic level fluctuations in the continuum reservoir. For this purpose, a novel coupling algorithm is developed based on the non-equilibrium analogy of the AdResS model and the pressure balancing results. In this approach, the continuum reservoir will provide information as a state boundary condition at the interface to the particle subdomain and the particle-based solver will provide information on the flux of mass, momentum, and energy of particles passing through the interface. Accordingly, a set of pre-calculated thermodynamic forces is prepared for several states around the target state of the fluid under study (dictionary of thermodynamic forces). Based on the resulting state of the continuum solver at the interface, a proper thermodynamic force for the AdResS simulation is obtained by interpolation from the dictionary of the thermodynamic force. Finally, the AdResS simulator is coupled to the fluctuating hydrodynamics solver based on the developed algorithm that uses the dictionary of thermodynamic forces on the fly. The new coupling solver is later tested for several scenarios with different initial conditions and a quasi-one-dimensional domain. The results show satisfactory correspondence compared to those concluded from pure continuum and fully atomistic simulations.

The new set-up with coupling AdResS to fluctuating hydrodynamics provides a tool for controlling the thermodynamics of the atomistic subdomain with the macroscopic quantities in the continuum reservoir while having an acceptable level of accuracy in the reservoir region. Thus, it makes the solver able to simulate some challenging applications e.g. biological membranes and proteins by putting the membrane or protein inside the atomistic subdomain with higher resolution and filling the reservoir continuum region with the solvent and water. It is also helpful to extend the continuum solver to a three-dimensional solver and make it able to simulate more realistic problems with complex geometries. This will enable the coupling system to simulate two or three-dimensional problems, e.g. Couette flow problem, in fluid mechanics.

In addition to the abovementioned application ideas for future endeavours, it would be beneficial for future studies to develop a predictor module to calculate the thermodynamic forces for different states of LJ liquids. As previously mentioned, LJ fluids are useful for theoretical studies as simple fluids with essential basic characteristics. However, as presented in this work, the thermodynamic force of the fluid depends on its density and temperature. Thus, performing various AdResS simulations for a wide range of thermodynamic state points of LJ fluid will establish a large data set for thermodynamic force as a function of density and temperature. Later, these data could be used to develop a predictor function for interpolating the required thermodynamic force for any thermodynamic state point in between. Such function would compensate considerable computational resources for future AdResS studies on fundamental topics with LJ fluids.

Appendix A

HALMD package

In this section, some tools for the implementation of AdResS in the HALMD package are explained and later the simulation code for running the AdResS simulation is presented.

A.1 AdResS simulation tools

A.1.1 Canonical scheme for thermodynamic force

Here, we have introduced a new approach for calculating thermodynamic force with an iterative formula based on the canonical ensemble assumption. This scheme has been used in all AdResS simulations with the HALMD package, i.e. coupling AdResS to fluctuating hydrodynamics solver.

The density distribution in the presence of an external potential for a canonical ensemble with constant N, V, and T is,

$$\rho = z \times \exp(-\beta\phi) \tag{A.1}$$

in which β is thermodynamic beta defined as $\frac{1}{k_B T}$, ϕ is the intended external potential, and z is the canonical partition function that is defined as below,

$$z = \frac{1}{h^3} \int e^{-\beta H(q,p)} d^3q d^3p \tag{A.2}$$

that h is the Planck constant and $H(q,p)$ is the Hamiltonian of the system as a function of position (q) and momentum (p).

According to Eq.A.1, the proper external potential (ϕ_t) that results the target density profile (ρ_t) in a certain direction, e.g. x, should obey the following format,

$$\rho_t = z_t \times \exp(-\beta\phi_t) \tag{A.3}$$

Computing the density profile resulting from an external potential is straightforward; however, doing the calculation vice versa and computing the needed external potential to produce a wished density profile in a certain direction over the simulation box is more challenging. An idea is to develop and use an iterative formula and apply an initial external potential to the system and compute the density profile based on that external potential and then correct the external potential according to the deviation of the resulting density from the target one. Let's suppose that the density profile at nth iteration after applying nth external potential (ϕ_n) is ρ_n; then, according to Eq.A.1,

$$\rho_n = z_n \times exp(-\beta\phi_n) \tag{A.4}$$

Consequently, the external potentials based on Eq.A.3 and Eq.A.4 can be written as below,

$$\phi_t(x) = -k_BT \times ln\left(\frac{\rho_n(x)}{z_n}\right) \quad \text{and} \quad \phi_n(x) = -k_BT \times ln\left(\frac{\rho_n(x)}{z_n}\right) \tag{A.5}$$

The iterative formulation can be considered as,

$$\phi_{n+1}(x) = \phi_n(x) + g(x) \tag{A.6}$$

The final goal is to find an expression for $g(x)$. At the end of aforementioned iterative procedure, ϕ_{n+1} should converge to the target external potential ϕ_t that means $\phi_{n+1}(x) \approx \phi_t(x)$. Thus, $\phi_t(x) = \phi_n(x) + g(x)$ and,

$$g(x) = \phi_t(x) - \phi_n(x) = -k_BT\left[ln\left(\frac{\rho_t}{z_t}\right) - ln\left(\frac{\rho_n}{z_n}\right)\right] \tag{A.7}$$

According to Eq.A.2, the canonical partition function depends on the Hamiltonian of the system which is the sum of interaction potentials and kinetic and this value should be the same for the system of study before and after applying external force. Therefore, z_t and z_n in Eq.A.7 are equal and one can rewrite the Eq .A.6 as below,

$$\phi_{n+1}(x) = \phi_n(x) + k_BT \times ln\left(\frac{\rho_n}{\rho_t}\right) \tag{A.8}$$

Eq.A.8 presents an iterative formulation for computing needed external potential to reach a certain target density (ρ_t). The corresponding external force would be:

$$F(x) = -\nabla\phi(x) \quad \Rightarrow \quad F_{n+1}(x) = F_n(x) - k_BT\left(\frac{\nabla\rho_n(x)}{\rho_n(x)} - \frac{\nabla\rho_t(x)}{\rho_t(x)}\right) \tag{A.9}$$

The iterative formula in Eq.A.9 and Eq.A.8 can be implemented in AdResS simulation for the calculation of the thermodynamic force.

A.1.2 External force implementation

To use the benefit of the formulation discussed in Eq.A.8 and Eq.A.9 in an MD simulation, a tool/module must be developed that works besides other tools in the MD package to apply the intended external potential to the simulation box. For this purpose, an external force module in the HALMD package has been developed which takes the provided external force and potential along with the corresponding binning as an array and then interpolates a continuous function for the provided coefficients and applies it to the system.

The intended module was developed for a three-dimensional simulation box that is capable to apply an external force field in all three directions; however, concerning the physics of the AdResS, a one-dimensional external force module will satisfy the need as the aim is to reach a target density profile in one direction. This module takes the coefficients array with values of external force and potential at certain nodes as a one-dimensional array based on a certain order of elements.

The external force module uses a cubic Hermite spline for interpolation between external potential values which is a spline where each piece is a third-degree polynomial specified in Hermite form: i.e, by its values (potentials) and first derivatives (minus force) at the endpoints of the corresponding domain interval. Cubic Hermite splines are typically used for interpolation of numeric data specified at given argument values $x_1, x_2, ..., x_n$, to obtain a smooth continuous function. The data should consist of the desired function value and derivative at each x_k. The

Hermite formula is applied to each interval (x_k, x_{k+1}), separately. The resulting spline will be continuous and will have a continuous first derivative. On the arbitrary interval of (x_k, x_{k+1}), given a starting point p_k at $x = x_k$ and an ending point p_{k+1} at $x = x_{k+1}$ with starting tangent m_k at $x = x_k$ and ending tangent m_{k+1} at $x = x_{k+1}$, one can find the unknown parameters of the target third-degree polynomial of $\phi(x) = ax^3 + bx^2 + cx + d$ by implementing boundary conditions for the corresponding interval. The following function for interpolation between two points x_k and x_{k+1} is derived,

$$\phi(t) = h_{00}(t)p_k + h_{10}(t)(x_{k+1} - x_k)m_k + h_{01}(t)p_{k+1} + h_{11}(t)(x_{k+1} - x_k)m_{k+1} \tag{A.10}$$

in which $t = \frac{x - x_k}{x_{k+1} - x_k}$, $h_{00} = 2t^3 - 3t^2 + 1$, $h_{10} = t^3 - 2t^2 + t$, $h_{01} = -2t^3 + 3t^2$, and $h_{11} = t^3 - t^2$.

Using the latter equation, one can compute the interpolated value for external potential (and consequently external force) by providing desired coefficients at the specific nodes.

The following function will calculate the desired potential coefficients for performing AdResS simulation on the HALMD package.

Listing A.1: External potential for thermodynamic force calculator

```
def compute_coefficients(nodes, T_dictionary, T_left, T_right, N, N_left, N_right,
    dx_p, AT, Delta, TR, step, potential_dict_directory):
    potential_dict =
        np.genfromtxt(potential_dict_directory).reshape((len(T_dictionary), len(N),
        int(np.prod(nodes[:])*8)))[:, :,0:int(nodes[0])*8:8]
    p_th_new = np.zeros(int(nodes[0]))

    for i in range(int(nodes[0]/2)):
        p_th_new[i] = interp2d(T_dictionary, N, potential_dict[:, :, i].transpose(),
            kind='linear', copy=True, bounds_error=False, fill_value=None)(T_left,
            N_left)
        p_th_new[-1-i] = interp2d(T_dictionary, N, potential_dict[:, :,
            -i-1].transpose(), kind='linear', copy=True, bounds_error=False,
            fill_value=None)(T_right, N_right)

    f_th_new = np.gradient(p_th_new, dx_p)
    temp = 0.5 * (f_th_new[0] + f_th_new[-1])
    f_th_new[0] = f_th_new[-1] = temp

    p_th_new[int((TR+Delta)/dx_p):int((TR+Delta+AT)/dx_p)+1] = 0
    f_th_new[int((TR+Delta)/dx_p):int((TR+Delta+AT)/dx_p)+1] = 0

    coefficients = np.zeros((int(nodes[0]), 1, 8))
    coefficients[:, 0, 0] = p_th_new
    coefficients[:, 0, 1] = -f_th_new
    coefficients = coefficients.flatten()
    coefficients_ = coefficients

    for i in range(int(np.prod(nodes[1:])) - 1):
        coefficients_ = np.vstack((coefficients_, coefficients))

    coefficients = coefficients_.flatten()

    return coefficients, p_th_new
```

A.1.3 Number density calculation

According to the discussion in the two previous sections, the coefficients of the external potential/force function could be computed by calculating density at each step and comparing it with the target one as described in Eq.A.8 and Eq.A.9. However, computing a continuous density function in a discrete particle-based environment is always challenging as one would encounter ups and downs and even zero values along the simulation domain due to its discrete nature which couldn't be acceptable. To avoid this behaviour, a common strategy is to provide density modes by computing complex Fourier modes of the particle density field in MD simulation according to Eq.A.11 and then doing the calculations on density modes and returning them to real space by taking a Fourier back transformation.

$$\rho(\overrightarrow{k}) = \sum_{n=1}^{N} exp(i\,\overrightarrow{k}.\overrightarrow{r_n}) \tag{A.11}$$

This scheme has been used in AdResS simulations in the HALMD package for the calculation of the density profile. The following Python script calculates the number density from the density modes recorded during MD simulations.

Listing A.2: number density calculator from density modes

```
def compute_density_profile(wavevector_list, density_modes_list, box_edges, width,
    user_axis):

    one_norm = np.linalg.norm(wavevector_list, axis=1, ord=1)
    wavevector_axis_component = wavevector_list[:, user_axis]
    idx, = np.where(one_norm == np.abs(wavevector_axis_component))

    wavevector_list_of_interest = wavevector_axis_component[idx]
    density_modes_list_of_interest = density_modes_list[idx]

    # generating and applying 1D-Gaussian filter
    gaussian = np.exp(-0.5 * width**2 * pow(wavevector_list_of_interest, 2))
    density_modes_smoothed = density_modes_list_of_interest * gaussian

    # In general (arbitrary dimensions) the density modes are a scalar field on the
        space of the wavevectors k. The wavevectors forming a cubic grid around 0
        (with halmd.observables.utility.wavevector(...,dense=true)) carry the
        information of the position in the k-space of each density mode. This
        information can be used to restructure the list of density modes to a density
        mode matrix. Hereby transforming the wavevectors by some factor to the set of
        smallest integers w, produces the index to locate the density modes in a
        matrix.
    w = np.array(np.round(wavevector_list_of_interest * box_edges[user_axis] / (2 *
        np.pi)), dtype=int)

    # initialising and filling the 1D-density_modes_matrix
    assert np.min(w) == -np.max(w), "Density-modes need to be on a symmetric grid
        around 0, i.e. k_max = -k_min"
    length = 2*np.max(w)+1
    density_modes_matrix = np.zeros(length, dtype=complex)
    density_modes_matrix[w] = density_modes_smoothed
    # Fourier back-transform
    density_unnormalized = np.fft.fftshift(np.fft.ifft(density_modes_matrix)).real

    tmp_err_edtr = 0.5*(density_unnormalized[0]+density_unnormalized[-1])
    density_unnormalized[-1] = tmp_err_edtr
```

```
    density_unnormalized[0] = tmp_err_edtr

    # normalisation
    # even though a one dimensional fft was done, the density_mode_values on the
        desired coordinate-axis represent the density averaged over the other
        coordinate axes. Therefore after the inverse Fourier transform, they
        correspond to an integral of the density field in the other coordinate axes by
        one number, but still refer to the full-dimensional box.
    volume = np.prod(box_edges)
    density = density_unnormalized * len(density_unnormalized) / volume
    # Generate the corresponding positions in real space, from the wavevectors
    position = box_edges[user_axis] / np.double(length) * np.linspace(np.min(w),
        np.max(w), length)

    return position, density
```

A.2 AdResS simulation code

Performing AdResS simulation in the HALMD package consists of two steps including a Python code and a Lua script. The Python code will do the calculation of thermodynamic force and the iterative procedure until reaching the flat condition. On the other hand, the Lua script will run the MD simulation on the HALMD package by calling the related modules. The necessary functions for the iterative procedure are presented in previous sections and the Lua script is presented in the following.

Listing A.3: Lua script for running MD/AdResS simulation on HALMD

```
-- grab modules
local log = halmd.io.log
local mdsim = halmd.mdsim
local numeric = halmd.numeric
local observables = halmd.observables
local writers = halmd.io.writers
local readers = halmd.io.readers
local utility = halmd.observables.utility
local random = halmd.random

function main(args)

    --parameters
    local timestep = args.time_step
    local temperature_left = args.temperature_left
   local temperature_right = args.temperature_right
    local steps = math.ceil(args.time / timestep)
    local equiblibration_steps = args.eq_to_pr * steps
    local nknots = {args.nknots, 2, 2}
    local DL_size = args.Delta_size
    local AT_size = args.AT_size
    local TR_size = args.TR_size
   local n_slice = args.n_slice

    --open H5MD file for reading
    local file_read = readers.h5md({path = args.input})
    local reader, sample = observables.phase_space.reader({file = file_read, location
        = {"particles", "all"}, fields = {"position", "velocity", "species", "mass"}})

    -- read phase space sample at last step in file
```

```
reader:read_at_step(-1)
-- determine system parameters from phase space sample
local nparticles = assert(sample.nparticle)
local nspecies = assert(sample.nspecies)
local dimension = assert(sample.dimension)

-- read edge vectors of simulation domain from file
local edges = mdsim.box.reader({file = file_read, location = {"particles", "all"}})

-- create simulation domain with periodic boundary conditions
local box = mdsim.box({edges = edges})

--setup
local particle = mdsim.particle({dimension = dimension, particles = nparticles,
    species = nspecies})

--all_group
local all_group = mdsim.particle_groups.all({particle = particle, label = "all"})

-- create group from cuboid region for TR in AT and vice versa
local DL_AT_box = mdsim.geometries.cuboid({lowest_corner = {-edges[1][1]/2 +
    TR_size, -edges[2][2]/2, -edges[3][3]/2}, length = {AT_size + 2*DL_size,
    edges[2][2] , edges[3][3]}})
local invader_TR = mdsim.particle_groups.region_species({species = 1, particle =
    particle, selection = "included", geometry = DL_AT_box, box = box, label =
    "invader_TR"})
local invader_AT = mdsim.particle_groups.region_species({species = 0, particle =
    particle, selection = "excluded", geometry = DL_AT_box, box = box, label =
    "invader_AT"})
local TR_group = mdsim.particle_groups.region({particle = particle, selection =
    "excluded", geometry = DL_AT_box, box = box, label = "TR"})

local AT_box = mdsim.geometries.cuboid({lowest_corner = {-edges[1][1]/2 + TR_size
    + DL_size, -edges[2][2]/2, -edges[3][3]/2}, length = {AT_size, edges[2][2] ,
    edges[3][3]}})
local AT_group = mdsim.particle_groups.region({particle = particle, selection =
    "included", geometry = AT_box, box = box, label = "AT"})
local Non_AT_group = mdsim.particle_groups.region({particle = particle, selection
    = "excluded", geometry = AT_box, box = box, label = "Non_AT"})

local l_Non_AT_box = mdsim.geometries.cuboid({lowest_corner =
    {-edges[1][1]/2-0.001, -edges[2][2]/2, -edges[3][3]/2}, length = {TR_size +
    DL_size + 0.001, edges[2][2] , edges[3][3]}})
local l_Non_AT_group = mdsim.particle_groups.region({particle = particle,
    selection = "included", geometry = l_Non_AT_box, box = box, label =
    "l_Non_AT"})
local r_Non_AT_box = mdsim.geometries.cuboid({lowest_corner = {-edges[1][1]/2 +
    TR_size + DL_size + AT_size, -edges[2][2]/2, -edges[3][3]/2}, length =
    {DL_size + TR_size + 0.001, edges[2][2] , edges[3][3]}})
local r_Non_AT_group = mdsim.particle_groups.region({particle = particle,
    selection = "included", geometry = r_Non_AT_box, box = box, label =
    "r_Non_AT"})

local slice_box = {}
local slice_group = {}
for i = 1,n_slice,1
do
  slice_box[i] = mdsim.geometries.cuboid({lowest_corner = {-edges[1][1]/2 +
```

```
        TR_size + DL_size + (i-1) * AT_size/n_slice, -edges[2][2]/2,
        -edges[3][3]/2}, length = {AT_size/n_slice, edges[2][2] , edges[3][3]}})
    slice_group[i] = mdsim.particle_groups.region({particle = particle, selection =
        "included", geometry = slice_box[i], box = box, label =
        "slice"..tostring(i)})
end

 local capping = mdsim.forces.cape_force({group = Non_AT_group , threshold = 500})

 local TR_change = mdsim.change_property({group = invader_TR , new_value= 0})
 local AT_change = mdsim.change_property({group = invader_AT , new_value= 1})

 -- truncated Lennard-Jones potential
 local potential = mdsim.potentials.pair.lennard_jones({species = 2,
    epsilon = {
        {1, 0} -- onAT_fromAT, onAT_fromTR
      , {0, 0} -- onTR_fromAT, onTR_fromTR
    }
   , sigma = {
        {1, 1} -- onAT_fromAT, onAT_fromTR
      , {1, 1} -- onTR_fromAT, onTR_fromTR
    }
 })
 -- smoothing at potential cutoff
 potential = potential:truncate({"smooth_r4",
       cutoff = {
        {2.5, 2.5}
      , {2.5, 2.5}
    }
   , h = 0.005
 })

 -- compute forces
 local force = mdsim.forces.pair({box = box, particle = particle, potential =
     potential})

 --external force/potential coefficients
 local interpolation = mdsim.forces.interpolation.cubic_hermite({box = box, nknots
     = nknots, precision = "single"})
 local virial_interpolation = mdsim.forces.interpolation.linear({box = box, nknots
     = nknots, precision = "single"})
 local tabulated_forces = mdsim.forces.tabulated_external({particle = particle, box
     = box, interpolation = interpolation, virial_interpolation =
     virial_interpolation})
 tabulated_forces:read_coefficients({file = file_read, location = {"parameters",
     "halmd"}})

 log.info(("Closing H5MD file %s"):format(file_read.path))
 file_read:close()

 --integrator
local l_Non_AT_integrator = mdsim.integrators.verlet_nvt_andersen({group =
    l_Non_AT_group, box = box, timestep = timestep, temperature = temperature_left,
    rate = 15})
 local r_Non_AT_integrator = mdsim.integrators.verlet_nvt_andersen({group =
     r_Non_AT_group, box = box, timestep = timestep, temperature =
     temperature_right, rate = 15})
 --local Non_AT_integrator = mdsim.integrators.verlet_nvt_andersen({group =
```

```
        Non_AT_group, box = box, timestep = timestep, temperature = temperature, rate
        = 15})

    local AT_integrator = mdsim.integrators.verlet({group = AT_group, box = box,
        timestep = timestep})

    -- sample phase space
    local phase_space = observables.phase_space({box = box, group = all_group}) --
        FIXME

    -- set particle positions, velocities, species
    phase_space:set(sample)

    -- H5MD file writer
    local file_write = writers.h5md({path = args.output, mode = "truncate", overwrite
        = true}) -- FIXME

    --equilibration run
    -- sample current state
    observables.sampler:sample()
    -- run half of the simulation
    observables.sampler:run(equibliration_steps)

--writer
    -- write trajectory of particle groups to H5MD file
   --local interval = 500000 or steps
    phase_space:writer({file = file_write, fields = {"position", "velocity",
        "species", "mass"}, every = 1000}) -- FIXME

    --sample macroscopic state variables for all particles.
    if args.sampling.state_vars > 0 then
        observables.thermodynamics({box = box, group = all_group})
            :writer({file = file_write, every = args.sampling.state_vars})

      for i = 1,n_slice,1
      do
         observables.thermodynamics({box = box, group = slice_group[i]})
            :writer({file = file_write, fields = {
                "potential_energy", "pressure", "temperature" -- fluctuating quantities
              , "internal_energy", "center_of_mass_velocity" , "heat_flux" -- conserved
                  quantities
            }, every = args.sampling.state_vars})
      end

      observables.thermodynamics({box = box, group = l_Non_AT_group})
         :writer({file = file_write, fields = {
               "potential_energy", "pressure", "temperature" -- fluctuating quantities
            , "internal_energy", "center_of_mass_velocity" , "heat_flux" -- conserved
                quantities
            }, every = args.sampling.state_vars})
      observables.thermodynamics({box = box, group = r_Non_AT_group})
         :writer({file = file_write, fields = {
               "potential_energy", "pressure", "temperature" -- fluctuating quantities
            , "internal_energy", "center_of_mass_velocity" , "heat_flux" -- conserved
                quantities
            }, every = args.sampling.state_vars})

    end
```

```
    --write density modes to H5MD file
    local kmax = (nknots[1] + 1) / 2 * (2 * math.pi / box.length[1])
    local wavevector = observables.utility.wavevector({box = box, wavenumber = {kmax},
        filter = {1, 0, 0}, dense = true})
    local density_mode = observables.density_mode({group = all_group, wavevector =
        wavevector})
    density_mode:writer({file = file_write, every = 100})

--average run
    -- run rest of the simulation
    observables.sampler:run(steps - equibliration_steps)

    -- log profiler results
    halmd.utility.profiler:profile()

end

-- Define command line parser.
function define_args(parser)
    parser:add_argument("output,o", {type = "string", action =
        parser.action.substitute_date_time,
        default = "result_out_%Y%m%d_%H%M%S", help = "prefix of output files"})
    parser:add_argument("overwrite", {type = "boolean", default = false, help =
        "overwrite output file"})

    parser:add_argument("input", {type = "string", required = true, action =
        function(args, key, value)
        readers.h5md.check(value)
        args[key] = value
    end, help = "H5MD input file"})

    parser:add_argument("time", {type = "number", default = 100, help = "integration
        time"})
   parser:add_argument("time_step", {type = "number", default = 0.002, help =
       "integration time step"})
   parser:add_argument("eq_to_pr", {type = "number", default = 0.3, help = "ratio of
       equilibrium to production run"})
    parser:add_argument("temperature_left", {type = "number", default = 1.5, help =
        "temperature of heat bath at the left side"})
   parser:add_argument("temperature_right", {type = "number", default = 1.5, help =
       "temperature of heat bath at the right side"})
   parser:add_argument("AT_size", {type = "number", default = 20, help = "size of the
       atomistic region"})
   parser:add_argument("Delta_size", {type = "number", default = 2.5, help = "size of
       the transition region"})
   parser:add_argument("TR_size", {type = "number", default = 15, help = "size of the
       reservoir (tracer) region"})
    parser:add_argument("nknots", {type = "number", help = "number of knots"})
    parser:add_argument("n_slice", {type = "number", default=10, help = "number of
        slices for recording temperature"})
    parser:add_argument("smoothing", {type = "number", default = 0.005, help = "cutoff
        smoothing parameter"})
    parser:add_argument("cutoff", {type = "float32", default = math.pow(2, 1 / 6),
        help = "potential cutoff radius"})
    parser:add_argument("force_threshold", {type = "float32", default = 50, help =
        "force_threshold"})
    local sampling = parser:add_argument_group("sampling", {help = "sampling intervals
```

```
        (0: disabled)"})
    sampling:add_argument("trajectory", {type = "integer", help = "for trajectory"})
    sampling:add_argument("state-vars", {type = "integer", default = 1000, help = "for
        state variables"})
    sampling:add_argument("structure", {type = "integer", default = 1000, help = "for
        density modes, static structure factor"})
    sampling:add_argument("correlation", {type = "integer", default = 100, help = "for
        correlation functions"})
    sampling:add_argument("average", {type = "integer", help = "output averages of
        given number of samples"})

    local wavevector = parser:add_argument_group("wavevector", {help = "wavevector
        shells in reciprocal space"})
    observables.utility.wavevector.add_options(wavevector, {tolerance = 0.01,
        max_count = 7})
    observables.utility.semilog_grid.add_options(wavevector, {maximum = 15, decimation
        = 0})
end
```

Appendix B

GROMACS package

For running AdResS on the GROMACS package, it is required to implement some changes in the parameter, topology, and configuration files. These include introducing AdResS parameters in the ".mdp" file and defining imaginary particles for each molecule in the ".conf" and ".top" files. In addition, to implement the iterative procedure for running AdResS and calculating thermodynamic force, the following scripts have been used.

Listing B.1: script for starting iterative procedure of AdResS

```
i=$1
j=$2

case $3 in
 [0]*)
rm F_th_* tabletf* SOL.dens.s* dens_mix_* \#* SOL.dens_s*.xvg
./Sym_manual_tf_calc_hlrn.sh 0 dens.SOL.out SOL.dens.out s0 2

;;
 [1]*)
rm F_th_* tabletf* SOL.dens.s* dens_mix_* \#* SOL.dens_s*.xvg
gmx grompp -f adress.mdp -c adress.gro -p adress.top -n index.ndx
./Sym_manual_tf_calc_hlrn.sh 0 dens.SOL.out SOL.dens.out s0 2

for z in `seq $i $j`; do
./Sym_manual_tf_calc_hlrn.sh 3 dens.SOL.out SOL.dens.out s$z 2
done

;;
 [2]*)
for z in `seq $i $j`; do
./Sym_manual_tf_calc_hlrn.sh 3 dens.SOL.out SOL.dens.out s$z 2
done
;;
esac
```

This script run the following script taking into account whether a simulation is a continuation of the previous simulation or is a new simulation.

Listing B.2: script for performing the AdResS simulation in an iterative manner until reaching equilibrium

```
#!/bin/bash

j=$1
```

```
l=$2
k=$3
m=$4
n=$5

rmin=7.5    # where the delta region begins with respect to the center of the box
    (where we would like Fth to begin)
rmax=60   # where the Fth will end with respect to center of box
rbox=120     # length of box in x direction
rref=60   # reference point (center of box) as measure in the positive x direction
lc=401      # number of bins for density calculation
rstep=$(echo $rbox/$lc | bc -l) # dx for density calc and Fth
r_cut=$(echo $rmin| bc -l)
rm_c=$(echo $rmax | bc -l)
echo $r_cut, $rm_c

prep_sys=$j
case $prep_sys in
 [0]*)

if [ ! -f F_th_step0 ]; then
   seq -f%g 0.0 $rstep $rbox | sed 's/,/./' | awk '{print $1, "0.0", "0.0"}'>
       F_th_step0
fi
cp F_th_step0 tabletf_WCG.xvg

echo "Beginning preliminary md simulation without thermodynamic force..."
sleep 2
gmx mdrun -v

;;
 [1]*)

if [ -f tabletf_CAT.xvg ]; then
   awk 'BEGIN{OFS="\t"}(NR>9){print $1, $2, $3}' tabletf_CAT.xvg > F_th_step0
fi
;;
 [2]*)

if [ -f tabletf_ANN.xvg ]; then
   awk 'BEGIN{OFS="\t"}(NR>9){print $1, $2, $3}' tabletf_ANN.xvg > F_th_step0
fi
;;
 [3]*)

if [ -f tabletf_WCG.xvg ]; then
   awk 'BEGIN{OFS="\t"}(NR>9){print $1, $2, $3}' tabletf_WCG.xvg > F_th_step0
fi
;;
esac

rn1=$(echo $rn+$rstep | bc -l)
echo $rn $rn1
dens_sys=$n
case $dens_sys in
 [0]*)

  csg_density --axis r --rmax 9.6 --ref [5.5,5.5,5.5] --trj
```

```
      ../7000_mmin/traj_comp.xtc --top ../7000_mmin/topol.tpr --out test.dens.comp
  csg_density --axis r --rmax 9.6 --ref [5.5,5.5,5.5] --trj
      ../7000_mmin/traj_comp.xtc --top ../7000_mmin/topol.tpr --molname Cl --out
      Cl.dens.comp
  csg_density --axis r --rmax 9.6 --ref [5.5,5.5,5.5] --trj
      ../7000_mmin/traj_comp.xtc --top ../7000_mmin/topol.tpr --molname MIL --out
      MIL.dens.comp
  csg_density --axis r --rmax 9.6 --ref [5.5,5.5,5.5] --trj traj_comp.xtc --top
      topol.tpr --out test.dens.out
  csg_density --axis r --rmax 9.6 --ref [5.5,5.5,5.5] --cg "MIL.cg.xml;cl.cg.xml"
      --trj traj_comp.xtc --top topol.tpr --molname Cl --out Cl.dens.out
  csg_density --axis r --rmax 9.6 --ref [5.5,5.5,5.5] --cg "MIL.cg.xml;cl.cg.xml"
      --trj traj_comp.xtc --top topol.tpr --molname MIL --out MIL.dens.out

;;
 [1]*)

  csg_density --axis r --rmax 9.6 --ref [5.5,5.5,5.5] --trj traj_comp.xtc --top
      topol.tpr --out test.dens.out
  csg_density --axis r --rmax 9.6 --ref [5.5,5.5,5.5] --cg "MIL.cg.xml;cl.cg.xml"
      --trj traj_comp.xtc --top topol.tpr --molname Cl --out Cl.dens.out
  csg_density --axis r --rmax 9.6 --ref [5.5,5.5,5.5] --cg "MIL.cg.xml;cl.cg.xml"
      --trj traj_comp.xtc --top topol.tpr --molname MIL --out MIL.dens.out

;;
 [2]*)

  echo "Calculating density..."
  sleep 2
  echo "2" > t_c; gmx density -d X -dens number -sl $lc -f traj_comp.xtc -o
      SOL.dens.xvg < t_c
  echo "Printing SOL.dens.out..."
  sleep 2
  awk 'BEGIN{OFS="\t"}(NR>24){print $1, $2}' SOL.dens.xvg > $k

;;
esac

# delete last line of density tables
# this avoids problems when calculating Fth
# to the end of the box
sed -i '$d' SOL.dens.out

# folding and symmetrizing density for smoothing:
awk '{d=$1-'$rref'; print ((d>0)?d:-d), $2}' $k > calc_dens_t

# duplicate the center line (0) so that there is again an even number
# of lines in the density table
sed -i 's/0 .*/&\n&/' calc_dens_t
rm x1 x2 xaa xab ref_dens calc_dens

# split the symmetrized density and average
split -l $(( ($lc-1)/2 )) calc_dens_t
awk '{printf "%10f %10f\n", $1, $2}' xaa | sort -g > x1
awk '{printf "%10f %10f\n", $1, $2}' xab | sort -g > x2
paste x1 x2 | awk '{print $1,($2+$4)/2.0}' > calc_dens

# print averaged dens profile in region of Fth calculation only
```

```
awk '{if(($1>=('$r_cut')) && ($1<=('$rm_c'))) print $1,$2}' calc_dens > dens_mix_hy

# interpolate between Fth region points
# and further take the derivative of this interpolated curve.
# this gives us our contribution to the Fth for this iteration
echo "Beginning interpolation..."
sleep 2
./smooth_dens.sh
cp dens_smooth d_s
cp pot_smooth p_s

echo "#","manually gen. Thermodyn. Force approx." > $m.xvg
echo "#","Parameter:">> $m.xvg
echo "#","start hy-region:", $rmin >> $m.xvg
echo "#","end of hy-region:",$rmax >> $m.xvg
echo "#","start tf: from xsplit" >> $m.xvg
echo "#","start hy-region:", $rmin >> $m.xvg
echo "#","end of hy-region:",$rmax >> $m.xvg
echo "#" >> $m.xvg
echo "#" >> $m.xvg

fmax=$(head -n 1 d_s | awk '{print $2}')
fmin=$(tail -n 1 d_s | awk '{print $2}')
pmax=$(head -n 1 p_s | awk '{print $2}')
pmin=$(tail -n 1 p_s | awk '{print $2}')
r_hy_1_temp=$(head -n 1 d_s | awk '{print $1-'$rstep'}')
r_hy_2=$(tail -n 1 d_s | awk '{print $1+'$rstep'}')
r_hy_1=$(echo $r_hy_1_temp+$rstep | bc -l)
rbox_new=$(echo $rbox+$rstep | bc -l)

seq -f%g 0.0 $rstep $r_hy_1 | sed 's/,/./' | awk '{print $1, 0.0}' > d0
awk '{print $1, $2}' d_s > d1
seq -f%g $r_hy_2 $rstep $rbox_new | sed 's/,/./' | awk '{print $1, 0.0}' > d2
seq -f%g 0.0 $rstep $r_hy_1 | sed 's/,/./' | awk '{print $1, '$pmax'}' > p0
awk '{print $1, $2}' p_s > p1
seq -f%g $r_hy_2 $rstep $rbox_new | sed 's/,/./' | awk '{print $1, '$pmin'}' > p2

cat d0 d1 d2 > d_m
cat p0 p1 p2 > p_m
paste p_m d_m | awk '{print $1,$2,$4}' > dens_mix
paste F_th_step0 dens_mix | awk 'BEGIN{OFS="\t"}{printf("%e %e
    %e\n",$1,($2+$5),($3-$6)) }' >> $m.xvg

cp $m.xvg tabletf_WCG_$m.xvg
cp $m.xvg tabletf_WCG.xvg
gmx mdrun -v
echo "Printing SOL.dens_s?.xvg..."
sleep 2
echo "2" > t_c; gmx density -d X -dens number -f traj_comp.xtc -o SOL.dens_$m.xvg <
    t_c
awk '{ print $1, $2/2 }' SOL.dens_$m.xvg > temp
mv temp SOL.dens_$m.xvg

# CLEAN UP
rm p? d? x? x?? \#*
rm SOL.dpot.*
rm p_m d_m
rm d_s d_s_t
```

```
rm t_c
rm s?.xvg s??.xvg
rm ref_dens
rm ref_dens_t calc_dens*
rm s.d.o
rm p_s
rm ref_t
rm calc_t
rm dens.ref
```

The following script calculates the thermodynamic force while performing the iterative procedure in AdResS simulation.

Listing B.3: script for calculating thermodynamic force

```
#!/usr/bin/env python

import sys
from NumPy import *
from scipy.signal import savgol_filter
from scipy import integrate
from scipy import interpolate
import matplotlib.pyplot as plt

f = file('dens_mix_hy', 'r')
tgt = loadtxt(f, usecols=(0,1))
r = tgt[:,0]
dr = tgt[:,1]
dtr = tgt[:,1]

T = 180
kbT = 0.00831451*T
prefac = 1

tck = interpolate.splrep(r, dr, s=0)
dr_s = interpolate.splev(r, tck, der=0)
dr_d = interpolate.splev(r, tck, der=1)

dr_p = dr_s*kbT*prefac
force_r = dr_d*kbT*prefac

SPL = column_stack((r, dr_s))
savetxt('dens_spline', SPL)

DAT = column_stack((r, force_r))
savetxt('dens_smooth', DAT)
POT = column_stack((r, dr_p))
savetxt('pot_smooth', POT)

plt.plot(r, dr)
plt.plot(r, dr_s)
plt.plot(r,dr_p)
plt.plot(r,force_r)
```

Appendix C

Fluctuating Hydrodynamics (FHD) code

The code of fluctuating hydrodynamics solver is designed in 3 integration steps as described in the RK3 model in the main text. In the following, the code for the first integration step is presented. The two other steps are similar to the following case but with different coefficients as explained in the main text.

Listing C.1: first integration step

```
def first_integral(rho, u, T, nx, boundary_type, fluctuations, kB, m, dt, Vc, dx,
    advection, diffusion, energy_equation):
    # arrays at first intermediate step
    rho_i = np.zeros_like(rho); u_i = np.zeros_like(u); T_i = np.zeros_like(T)
    #calculating edge values + applying boundary conditions to them
    rho_e, u_e, T_e, P_e, v_e, k_e, s_e, q_e = edge(rho, u, T, nx, boundary_type,
        fluctuations, kB, m, dt, Vc)
    for j in range(1, nx-1):

        rho_i[j] = rho[j] + (dt/dx) * (-advection * (rho_e[j+1]*u_e[j+1] -
            rho_e[j]*u_e[j]) )

        u_i[j] = (rho[j]*u[j] + (dt/dx) * (-advection * ((rho_e[j+1]*u_e[j+1]**2 +
            P_e[j+1]) - (rho_e[j]*u_e[j]**2 + P_e[j])) + diffusion *
            4.0/3.0*((v_e[j+1]*(u[j+1]-u[j])/dx) - (v_e[j]*(u[j]-u[j-1])/dx)) +
            (s_e[j+1]-s_e[j])) ) / rho_i[j]

        T_i[j] = (functions.E(rho[j], u[j], T[j]) + (dt/dx) * (-advection *
            ((functions.E(rho_e[j+1], u_e[j+1], T_e[j+1])+P_e[j+1])*u_e[j+1] -
            (functions.E(rho_e[j], u_e[j], T_e[j])+P_e[j])*u_e[j]) + diffusion *
            ((4.0/3.0*v_e[j+1]*u_e[j+1]*(u[j+1]-u[j])/dx+k_e[j+1]*(T[j+1]-T[j])/dx) -
            (4.0/3.0*v_e[j]*u_e[j]*(u[j]-u[j-1])/dx+k_e[j]*(T[j]-T[j-1])/dx)) +
            ((q_e[j+1]+u_e[j+1]*s_e[j+1]) - (q_e[j]+u_e[j]*s_e[j])) ) -
            1.0/2.0*rho_i[j]*u_i[j]**2)/(functions.cv_calculator(rho_i[j],
            T[j])*rho_i[j])

    #BOUNDARY CONDITION
    rho_i[0], rho_i[nx-1], u_i[0], u_i[nx-1], T_i[0], T_i[nx-1] =
        first_integral_boundary(boundary_type, rho, u, T, rho_e, u_e, T_e, P_e, v_e,
        k_e, s_e, q_e, nx, dt, dx, advection, diffusion)


    return rho_i, u_i, T_i
```

In this script, some interpolation functions are called to calculate the fluid's thermal properties at each step. In addition, at the beginning of each step, it is required to compute the edge values to integrate the fluxes for each continuum cell. The following function calculates the edge values based on the central values of the continuum cells.

Listing C.2: edge value calculator

```
def edge(rho, u, T, nx, boundary_type, fluctuations, kB, m, dt, Vc):
    # interpolation coefficients
    alpha = [(math.sqrt(7.0) + 1.0) / 4.0, (math.sqrt(7.0) - 1.0) / 4.0]

    mu = 0 # Gaussian distributed random values' mean
    sigma = 1 # Gaussian distributed random values' variance

    if fluctuations=='on':
        fluctuation_coefficient=1.0
    elif fluctuations=='off':
        fluctuation_coefficient=0

    # arrays at edge of cells
    rho_e = np.zeros(nx + 1); u_e = np.zeros(nx + 1); T_e = np.zeros(nx + 1)
    P_e = np.zeros(nx + 1); v_e = np.zeros(nx + 1); k_e = np.zeros(nx + 1); s_e =
        np.zeros(nx + 1); q_e = np.zeros(nx + 1)
    #interplotation to find conservative quantities at j+1/2 and j-1/2 from
        cell-centred values
    for j in range(2, nx-1):

        rho_e[j] = alpha[0] * (rho[j-1] + rho[j]) - alpha[1] * (rho[j-2] + rho[j+1])
        u_e[j] = (alpha[0] * (rho[j-1]*u[j-1] + rho[j]*u[j]) - alpha[1] *
            (rho[j-2]*u[j-2] + rho[j+1]*u[j+1])) / rho_e[j]
        T_e[j] = ((alpha[0] * (functions.E(rho[j-1], u[j-1], T[j-1]) +
            functions.E(rho[j], u[j], T[j])) - alpha[1] * (functions.E(rho[j-2],
            u[j-2], T[j-2]) + functions.E(rho[j+1], u[j+1], T[j+1]))) -
            1.0/2.0*rho_e[j]*u_e[j]**2) / (functions.cv_calculator(rho_e[j],
            T[j])*rho_e[j])

    #BOUNDARY CONDITION on edge values
    rho_e[0], rho_e[1], rho_e[nx], rho_e[nx-1], u_e[0], u_e[1], u_e[nx], u_e[nx-1],
        T_e[0], T_e[1], T_e[nx], T_e[nx-1], s_e[0], s_e[1], s_e[nx], s_e[nx-1],
        q_e[0], q_e[1], q_e[nx], q_e[nx-1] = edge_boundary(boundary_type,
        fluctuation_coefficient, alpha, rho, u, T, nx, kB, dt, Vc, mu, sigma)

    #computing flow properties
    for j in range(nx+1):
        P_e[j] = functions.P_calculator(rho_e[j],T_e[j])
        v_e[j] = functions.v_calculator(rho_e[j], T_e[j])
        k_e[j] = functions.k_calculator(rho_e[j], T_e[j])

    #stochastic terms
    for j in range(2, nx-1):
        s_e[j] = fluctuation_coefficient *
            (1+math.sqrt(3)/3)*math.sqrt(kB/(dt*Vc))*(functions.v_calculator(rho[j-1],
            T[j-1])*T[j-1] + functions.v_calculator(rho[j], T[j])*T[j])) * gauss(mu,
            sigma) #stochastic stress tensor
        q_e[j] = fluctuation_coefficient *
            math.sqrt(kB/(dt*Vc))*(functions.k_calculator(rho[j-1], T[j-1])*T[j-1]**2 +
            functions.k_calculator(rho[j], T[j])*T[j]**2)) * gauss(mu, sigma) #
            stochastic heat flux
```

```
    return rho_e, u_e, T_e, P_e, v_e, k_e, s_e, q_e
```

It is also necessary to apply appropriate boundary conditions, which in our case is a periodic boundary condition, to the system. As an example, the boundary condition function for the first integral is presented in the following. The logic for the boundary condition of the two other integration steps and also the edge values is the same.

Listing C.3: periodic boundary condition for the first integration step

```
def first_integral_boundary(boundary_type, rho, u, T, rho_e, u_e, T_e, P_e, v_e, k_e,
    s_e, q_e, nx, dt, dx, advection, diffusion):
    #periodic boundary condition
    if (boundary_type==1):

        rho_i_0 = rho[0] + (dt/dx) * (-advection * (rho_e[1]*u_e[1] - rho_e[0]*u_e[0]))
        rho_i_nx_1 = rho[nx-1] + (dt/dx) * (-advection * (rho_e[nx]*u_e[nx] -
            rho_e[nx-1]*u_e[nx-1]) )

        u_i_0 = (rho[0]*u[0] + (dt/dx) * (-advection * ((rho_e[1]*u_e[1]**2 + P_e[1])
            - (rho_e[0]*u_e[0]**2 + P_e[0])) + diffusion * 4.0/3.0 * ((v_e[1] * (u[1]
            - u[0])/dx) - (v_e[0] * (u[0] - u[nx-1])/dx)) + (s_e[1]-s_e[0])) ) /
            rho_i_0
        u_i_nx_1 = (rho[nx-1]*u[nx-1] + (dt/dx) * (-advection * ((rho_e[nx]*u_e[nx]**2
            + P_e[nx]) - (rho_e[nx-1]*u_e[nx-1]**2 + P_e[nx-1])) + diffusion * 4.0/3.0
            * ((v_e[nx] * (u[0] - u[nx-1])/dx) - (v_e[nx-1] * (u[nx-1] - u[nx-2])/dx))
            + (s_e[nx] - s_e[nx-1])) ) / rho_i_nx_1

        T_i_0 = (functions.E(rho[0], u[0], T[0]) + (dt/dx) * ( -advection *
            ((functions.E(rho_e[1], u_e[1], T_e[1]) + P_e[1]) * u_e[1] -
            (functions.E(rho_e[0], u_e[0], T_e[0]) + P_e[0]) * u_e[0]) + diffusion *
            ((4.0/3.0 * v_e[1] * u_e[1] * (u[1] - u[0])/dx + k_e[1] * (T[1] -
            T[0])/dx) - (4.0/3.0 * v_e[0] * u_e[0] * (u[0] - u[nx-1])/dx + k_e[0] *
            (T[0] - T[nx-1])/dx)) + ((q_e[1] + u_e[1] * s_e[1]) - (q_e[0] + u_e[0] *
            s_e[0])) ) - 1.0/2.0*rho_i_0 * u_i_0**2) /
            (functions.cv_calculator(rho_i_0, T[0])*rho_i_0)
        T_i_nx_1 = (functions.E(rho[nx-1], u[nx-1], T[nx-1]) + (dt/dx) * ( -advection
            * ((functions.E(rho_e[nx], u_e[nx], T_e[nx]) + P_e[nx]) * u_e[nx] -
            (functions.E(rho_e[nx-1], u_e[nx-1], T_e[nx-1]) + P_e[nx-1]) * u_e[nx-1])
            + diffusion * ((4.0/3.0 * v_e[nx] * u_e[nx] * (u[0] - u[nx-1])/dx +
            k_e[nx] * (T[0] - T[nx-1])/dx) - (4.0/3.0 * v_e[nx-1] * u_e[nx-1] *
            (u[nx-1] - u[nx-2])/dx + k_e[nx-1] * (T[nx-1] - T[nx-2])/dx)) + ((q_e[nx]
            + u_e[nx] * s_e[nx]) - (q_e[nx-1] + u_e[nx-1] * s_e[nx-1])) ) -
            1.0/2.0*rho_i_nx_1*u_i_nx_1**2) / (functions.cv_calculator(rho_i_nx_1,
            T[nx-1])*rho_i_nx_1)

    return rho_i_0, rho_i_nx_1, u_i_0, u_i_nx_1, T_i_0, T_i_nx_1
```

Appendix D

Zusammenfassung

Die Adaptive Resolution Simulation (AdResS) ist eine Mehrfachauflösungsmethode mit Eigenschaften eines offenen Systems zur Modellierung von Systemen auf atomistischer Ebene. Bei AdResS steht ein hochauflösendes offenes System in Kontakt mit einem Reservoir von Teilchen und Energie, und das System bildet die Thermodynamik und Physik des vollständigen atomistischen Bezugssystems nach. In dieser Arbeit werden die grundlegenden Eigenschaften der AdResS-Methode untersucht, um ein besseres Verständnis der statistischen Mechanik in einem offenen System zu ermöglichen.

Zu den wichtigsten Ergebnissen gehört die Äquivalenz zwischen dem theoretisch ermittelten Großkanonischen Potential und dem numerisch berechneten Druck. Darüber hinaus führte diese Analyse zu einer einfachen Berechnung des chemischen Potenzials der untersuchten Flüssigkeit für ein breites Spektrum thermodynamischer Bedingungen. Es wurde gezeigt, dass der Druckunterschied, der sich aus der abrupten Änderung der Auflösung ergibt, durch die Energie kompensiert wird, die von der äußeren Kraft (thermodynamische Kraft) in AdResS bereitgestellt wird.

Als Nächstes wird ein fluktuierender hydrodynamischer (FHD) Solver entwickelt, um die kleinräumigen Fluktuationen in den Kontinuumssimulationen zu erfassen, indem ein stochastischer Flussterm zur Navier-Stokes-Gleichung der kompressiblen Strömung hinzugefügt wird. Anschließend wird dieser Kontinuumslöser mit dem zuvor entwickelten AdResS-Simulator durch eine kleine Schnittstellenregion gekoppelt, indem ein neuartiger Kopplungsalgorithmus entsprechend der Nicht-Gleichgewichts AdResS-Simulation eingesetzt wird. Zu diesem Ziel wird ein Satz von vorberechneten thermodynamischen Kräften vorbereitet und die Informationen auf der Kontinuumsseite werden durch Interpolation geeigneter thermodynamischer Kräfte auf das Partikel-Subdomain transferieren. Das AdResS-FHD-Kopplungssystem wurde für verschiedene Fälle mit unterschiedlichen Bedingungen entwickelt und getestet und zeigte zufriedenstellende Übereinstimmung mit den Ergebnissen der Referenzkontinuums- und vollständig atomistischen Simulationen.

Bibliography

[1] R. Adhikari, K. Stratford, M. Cates, and A. Wagner. Fluctuating lattice boltzmann. *EPL (Europhysics Letters)*, 71(3):473, 2005.

[2] R. Adiyaman and L. J. McGuffin. Methods for the refinement of protein structure 3d models. *International journal of molecular sciences*, 20(9):2301, 2019.

[3] A. Agarwal, C. Clementi, and L. Delle Site. Path integral-gc-adress simulation of a large hydrophobic solute in water: a tool to investigate the interplay between local microscopic structures and quantum delocalization of atoms in space. *Physical Chemistry Chemical Physics*, 19(20):13030–13037, 2017.

[4] A. Agarwal, C. Clementi, and L. D. Site. Path integral-gc-adress simulation of a large hydrophobic solute in water: A tool to investigate the interplay between local microscopic structures and quantum delocalization of atoms in space. *Phys.Chem. Chem. Phys.*, 19:13030–13037, 2017.

[5] A. Agarwal and L. Delle Site. Path integral molecular dynamics within the grand canonical-like adaptive resolution technique: Simulation of liquid water. *The Journal of chemical physics*, 143(9):094102, 2015.

[6] A. Agarwal, J. Zhu, C. Hartmann, H. Wang, and L. D. Site. Molecular dynamics in a grand ensemble: Bergmann–lebowitz model and adaptive resolution simulation. *New Journal of Physics*, 17(8):083042, aug 2015.

[7] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. Cell junctions. In *Molecular Biology of the Cell. 4th edition*. Garland Science, 2002.

[8] I. Aleksandr and A. Khinchin. *Mathematical foundations of statistical mechanics*. Courier Corporation, 1949.

[9] F. J. Alexander and A. L. Garcia. The direct simulation monte carlo method. *Computers in Physics*, 11(6):588–593, 1997.

[10] M. P. Allen and D. J. Tildesley. *Computer simulation of liquids*. Oxford university press, 2017.

[11] H. C. Andersen. Molecular dynamics simulations at constant pressure and/or temperature. *The Journal of chemical physics*, 72(4):2384–2393, 1980.

[12] R. D. Astumian and P. Hänggi. Brownian motors. *Physics today*, 55(11):33–39, 2002.

[13] P. J. Atzberger, P. R. Kramer, and C. S. Peskin. A stochastic immersed boundary method for fluid-structure dynamics at microscopic length scales. *J. Comput. Phys*, pages 1255–1292, 2007.

[14] C. Aust, M. Kröger, and S. Hess. Structure and dynamics of dilute polymer solutions under shear flow via nonequilibrium molecular dynamics. *Macromolecules*, 32(17):5660–5672, 1999.

[15] G. Batchelor. *An introduction to fluid dynamics*. Cambridge university press, 2000.

[16] J. B. Bell, A. L. Garcia, and S. A. Williams. Numerical methods for the stochastic landau-lifshitz navier-stokes equations. *Physical Review E*, 76(1):016708, 2007.

[17] I. Bena, F. Baras, and M. M. Mansour. Hydrodynamic fluctuations in the kolmogorov flow: Nonlinear regime. *Physical Review E*, 62(5):6560, 2000.

[18] I. Bena, M. M. Mansour, and F. Baras. Hydrodynamic fluctuations in the kolmogorov flow: Linear regime. *Physical Review E*, 59(5):5503, 1999.

[19] H. J. Berendsen, J. v. Postma, W. F. Van Gunsteren, A. DiNola, and J. R. Haak. Molecular dynamics with coupling to an external bath. *The Journal of chemical physics*, 81(8):3684–3690, 1984.

[20] P. Bergmann and J. Lebowitz. New approach to nonequilibrium processes. *Phys. Rev.*, 99:578, 1955.

[21] P. G. Bergmann and J. L. Lebowitz. New approach to nonequilibrium processes. *Physical Review*, 99(2):578, 1955.

[22] A. Beskok and G. Karniadakis. *Microflows and Nanoflows: Fundamentals and Simulation. Interdisciplinary Applied Mathematics*. Springer, 2005.

[23] P. L. Bhatnagar, E. P. Gross, and M. Krook. A model for collision processes in gases. i. small amplitude processes in charged and neutral one-component systems. *Phys. Rev.*, 94:511–525, May 1954.

[24] J. Q. Broughton, F. F. Abraham, N. Bernstein, and E. Kaxiras. Concurrent coupling of length scales: methodology and application. *Physical review B*, 60(4):2391, 1999.

[25] D. Brown and S. Neyertz. A general pressure tensor calculation for molecular dynamics simulations. *Molecular Physics*, 84(3):577–595, 1995.

[26] W. Cai, M. de Koning, V. V. Bulatov, and S. Yip. Minimizing boundary reflections in coupled-domain simulations. *Physical Review Letters*, 85(15):3213, 2000.

[27] J. A. Carlson, A. Jaffe, and A. Wiles. *The millennium prize problems*. Citeseer, 2006.

[28] S. Chen and G. D. Doolen. Lattice boltzmann method for fluid flows. *Annual review of fluid mechanics*, 30(1):329–364, 1998.

[29] C. Choudhury, U. D. Priyakumar, and G. N. Sastry. Dynamics based pharmacophore models for screening potential inhibitors of mycobacterial cyclopropane synthase. *Journal of chemical information and modeling*, 55(4):848–860, 2015.

[30] G. Ciccotti and L. Delle Site. The physics of open systems for the simulation of complex molecular environments in soft matter. *Soft Matter*, 15:2114–2124, 2019.

[31] P. Colella and P. R. Woodward. The piecewise parabolic method (ppm) for gas-dynamical simulations. *Journal of computational physics*, 54(1):174–201, 1984.

[32] G. Csányi, T. Albaret, M. Payne, and A. De Vita. "learn on the fly": A hybrid classical and quantum-mechanical molecular dynamics simulation. *Physical review letters*, 93(17):175503, 2004.

[33] G. De Fabritiis, R. Delgado-Buscalioni, and P. Coveney. Multiscale modeling of liquids with molecular specificity. *Physical review letters*, 97(13):134501, 2006.

[34] G. De Fabritiis, R. Delgado-Buscalioni, and P. V. Coveney. Energy controlled insertion of polar molecules in dense fluids. *The Journal of chemical physics*, 121(24):12139–12142, 2004.

[35] G. De Fabritiis, M. Serrano, R. Delgado-Buscalioni, and P. Coveney. Fluctuating hydrodynamic modeling of fluids at the nanoscale. *Physical Review E*, 75(2):026307, 2007.

[36] E. de Miguel and G. Jackson. The nature of the calculation of the pressure in molecular simulations of continuous models from volume perturbations. *The Journal of Chemical Physics*, 125(16):164109, 2006.

[37] R. Delgado-Buscalioni and P. Coveney. Continuum-particle hybrid coupling for mass, momentum, and energy transfers in unsteady fluid flow. *Physical Review E*, 67(4):046704, 2003.

[38] R. Delgado-Buscalioni and P. Coveney. Usher: an algorithm for particle insertion in dense fluids. *The Journal of chemical physics*, 119(2):978–987, 2003.

[39] R. Delgado-Buscalioni and P. V. Coveney. Structure of a tethered polymer under flow using molecular dynamics and hybrid molecular-continuum simulations. *Physica A: Statistical Mechanics and its Applications*, 362(1):30–35, 2006.

[40] R. Delgado-Buscalioni and G. De Fabritiis. Embedding molecular dynamics within fluctuating hydrodynamics in multiscale simulations of liquids. *Phys. Rev. E*, 76:036709, Sep 2007.

[41] R. Delgado-Buscalioni and G. De Fabritiis. Embedding molecular dynamics within fluctuating hydrodynamics in multiscale simulations of liquids. *Physical Review E*, 76(3):036709, 2007.

[42] R. Delgado-Buscalioni, E. Flekkøy, and P. Coveney. Fluctuations and continuity in particle-continuum hybrid simulations of unsteady flows based on flux-exchange. *EPL (Europhysics Letters)*, 69(6):959, 2005.

[43] R. Delgado-Buscalioni, K. Kremer, and M. Praprotnik. Concurrent triple-scale simulation of molecular liquids. *The Journal of chemical physics*, 128(11):114110, 2008.

[44] L. Delle Site. Some fundamental problems for an energy-conserving adaptive-resolution molecular dynamics scheme. *Phys. Rev. E*, 76:047701, 2007.

[45] L. Delle Site. Formulation of liouville's theorem for grand ensemble molecular simulations. *Phys.Rev.E*, 93:022130, 2016.

[46] L. Delle Site. Grand canonical adaptive resolution simulation for molecules with electrons: a theoretical framework based on physical consistency. *Computer Physics Communications*, 222:94–101, 2018.

[47] L. Delle Site and R. Klein. Liouville-type equations for the n-particle distribution functions of an open system. *Journal of Mathematical Physics*, 61(8):083102, 2020.

[48] L. Delle Site, C. Krekeler, J. Whittaker, A. Agarwal, R. Klein, and F. Höfling. Molecular dynamics of open systems: construction of a mean-field particle reservoir. *Adv.Th.Sim.*, 2:1900014, 2019.

[49] L. Delle Site, C. Krekeler, J. Whittaker, A. Agarwal, R. Klein, and F. Höfling. Molecular dynamics of open systems: Construction of a mean-field particle reservoir. *Advanced Theory and Simulations*, 2(5):1900014, 2019.

[50] L. Delle Site, C. Krekeler, J. Whittaker, A. Agarwal, R. Klein, and F. Höfling. Molecular dynamics of open systems: Construction of a mean-field particle reservoir. *Advanced Theory and Simulations*, 2(5):1900014, 2019.

[51] A. Donev, J. B. Bell, A. L. Garcia, and B. J. Alder. A hybrid particle-continuum method for hydrodynamics of complex fluids. *Multiscale Modeling & Simulation*, 8(3):871–911, 2010.

[52] A. Donev, E. Vanden-Eijnden, A. Garcia, and J. Bell. On the accuracy of finite-volume schemes for fluctuating hydrodynamics. *Communications in Applied Mathematics and Computational Science*, 5(2):149–197, 2010.

[53] L. Dong, R. W. Smith, and D. J. Srolovitz. A two-dimensional molecular dynamics simulation of thin film growth by oblique deposition. *Journal of Applied Physics*, 80(10):5682–5690, 1996.

[54] B. Dünweg, U. D. Schiller, and A. J. Ladd. Statistical mechanics of the fluctuating lattice boltzmann equation. *Physical Review E*, 76(3):036704, 2007.

[55] A. Dupuis, E. Kotsalis, and P. Koumoutsakos. Coupling lattice boltzmann and molecular dynamics models for dense fluids. *Physical Review E*, 75(4):046704, 2007.

[56] R. Ebrahimi Viand, F. Höfling, R. Klein, and L. Delle Site. Theory and simulation of open systems out of equilibrium. *J.Chem.Phys.*, 153:101102, 2020.

[57] R. Ebrahimi Viand, F. Höfling, R. Klein, and L. Delle Site. Theory and simulation of open systems out of equilibrium. *The Journal of Chemical Physics*, 153(10):101102, 2020.

[58] R. Ebrahimi Viand, F. Höfling, R. Klein, and L. Delle Site. Theory and simulation of open systems out of equilibrium. *The Journal of Chemical Physics*, 153(10):101102, 2020.

[59] J. Eggers. Dynamics of liquid nanojets. *Physical review letters*, 89(8):084502, 2002.

[60] P. Español, R. Delgado-Buscalioni, R. Everaers, R. Potestio, D. Donadio, and K. Kremer. Statistical mechanics of hamiltonian adaptive resolution simulations. *J. Chem. Phys.*, 142:064115, 2015.

[61] P. Español, R. Delgado-Buscalioni, R. Everaers, R. Potestio, D. Donadio, and K. Kremer. Statistical mechanics of hamiltonian adaptive resolution simulations. *J. Chem. Phys.*, 142:064115, 2015.

[62] X. Fan, N. Phan-Thien, S. Chen, X. Wu, and T. Yong Ng. Simulating flow of dna suspension using dissipative particle dynamics. *Physics of Fluids*, 18(6):063102, 2006.

[63] E. Flekkøy, G. Wagner, and J. Feder. Hybrid model for combined particle and continuum dynamics. *EPL (Europhysics Letters)*, 52(3):271, 2000.

[64] E. G. Flekkøy, R. Delgado-Buscalioni, and P. V. Coveney. Flux boundary conditions in particle simulations. *Physical Review E*, 72(2):026703, 2005.

[65] R. W. Fox, A. T. McDonald, and J. W. Mitchell. *Fox and McDonald's introduction to fluid mechanics*. John Wiley & Sons, 2020.

[66] D. Frenkel and B. Smit. *Understanding Molecular Simulation*. Academic Press, 1996.

[67] D. Frenkel and B. Smit. *Understanding molecular simulation: from algorithms to applications*, volume 1. Elsevier, 2001.

[68] S. Fritsch, S. Poblete, C. Junghans, G. Ciccotti, L. Delle Site, and K. Kremer. Adaptive resolution molecular dynamics simulation through coupling to an internal particle reservoir. *Phys. Rev. Lett.*, 108:170602, Apr 2012.

[69] A. García and C. Penland. Fluctuating hydrodynamics and principal oscillation pattern analysis. *Journal of statistical physics*, 64(5):1121–1132, 1991.

[70] A. L. Garcia, J. B. Bell, W. Y. Crutchfield, and B. J. Alder. Adaptive mesh and algorithm refinement using direct simulation monte carlo. *Journal of computational Physics*, 154(1):134–155, 1999.

[71] A. L. Garcia, M. M. Mansour, G. C. Lie, and E. Cementi. Numerical integration of the fluctuating hydrodynamic equations. *Journal of statistical physics*, 47(1):209–228, 1987.

[72] H. Geng, F. Chen, J. Ye, and F. Jiang. Applications of molecular dynamics simulation in structure prediction of peptides and proteins. *Computational and Structural Biotechnology Journal*, 17:1162–1170, 2019.

[73] A. Gholami, F. Höfling, R. Klein, and L. Delle Site. Thermodynamic relations at the coupling boundary in adaptive resolution simulations for open systems. *Advanced Theory and Simulations*, 4(4):2000303, 2021.

[74] A. Gholami, R. Klein, and L. Delle Site. On the relation between pressure and coupling potential in adaptive resolution simulations of open systems in contact with a reservoir. *Advanced Theory and Simulations*, page 2100212, 2022.

[75] A. Gholami, R. Klein, and L. D. Site. Simulation of a particle domain in a continuum/fluctuating hydrodynamics reservoir. *arXiv preprint arXiv:2210.15516*, 2022.

[76] A. Gholami and A. Shamloo. Antibody consumption reduction in lateral flow immunoassays within porous media. *Chemical Engineering and Processing-Process Intensification*, 147:107773, 2020.

[77] J. W. Gibbs. *Elementary principles in statistical mechanics: developed with especial reference to the rational foundations of thermodynamics*. C. Scribner's sons, 1902.

[78] R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Monthly notices of the royal astronomical society*, 181(3):375–389, 1977.

[79] G. Giupponi, G. De Fabritiis, and P. V. Coveney. Hybrid method coupling fluctuating hydrodynamics and molecular dynamics for the simulation of macromolecules. *The Journal of Chemical Physics*, 126(15):154903, 2007.

[80] G. Giupponi, G. De Fabritiis, and P. V. Coveney. Hybrid method coupling fluctuating hydrodynamics and molecular dynamics for the simulation of macromolecules. *The Journal of chemical physics*, 126(15):154903, 2007.

[81] C. G. Gray, K. E. Gubbins, and C. G. Joslin. *Theory of Molecular Fluids: Volume 2: Applications*, volume 10. Oxford University Press, 2011.

[82] G. Greczynski, S. Mráz, J. M. Schneider, and L. Hultman. Metal-ion subplantation: A game changer for controlling nanostructure and phase formation during film growth by physical vapor deposition. *Journal of Applied Physics*, 127(18):180901, 2020.

[83] N. G. Hadjiconstantinou. The limits of navier-stokes theory and kinetic extensions for describing small-scale gaseous hydrodynamics. *Physics of Fluids*, 18(11):111301, 2006.

[84] N. G. Hadjiconstantinou and A. T. Patera. Heterogeneous atomistic-continuum representations for dense fluid systems. *International Journal of Modern Physics C*, 8(04):967–976, 1997.

[85] J. M. Haile, I. Johnston, A. J. Mallinckrodt, and S. McKay. Molecular dynamics simulation: Elementary methods. *Computers in Physics*, 7(6):625–625, 1993.

[86] J. P. Hansen and I. McDonald. *Theory of Simple Liquids*. Academic, London, 1990.

[87] J.-P. Hansen and I. R. McDonald. *Theory of simple liquids: with applications to soft matter*. Academic press, 2013.

[88] M. Heidari, R. Cortes-Huerto, D. Donadio, and R. Potestio. Accurate and general treatment of electrostatic interaction in hamiltonian adaptive resolution simulations. *The European Physical Journal Special Topics*, 225(8):1505–1526, 2016.

[89] H. Heinz, W. Paul, and K. Binder. Calculation of local pressure tensors in systems with many-body interactions. *Phys. Rev. E*, 72:066704, Dec 2005.

[90] L. Heo and M. Feig. Experimental accuracy in protein structure refinement via molecular dynamics simulations. *Proceedings of the National Academy of Sciences*, 115(52):13276–13281, 2018.

[91] A. Heyden, H. Lin, and D. G. Truhlar. Adaptive partitioning in combined quantum mechanical and molecular mechanical calculations of potential energy functions for multiscale simulations. *The Journal of Physical Chemistry B*, 111(9):2231–2241, 2007.

[92] F. B. Hildebrand. *Introduction to numerical analysis*. Courier Corporation, 1987.

[93] C.-M. Ho and Y.-C. Tai. Micro-electro-mechanical-systems (mems) and fluid flows. *Annual review of fluid mechanics*, 30(1):579–612, 1998.

[94] W. G. Hoover. Canonical dynamics: Equilibrium phase-space distributions. *Physical review A*, 31(3):1695, 1985.

[95] G. Hu and D. Li. Multiscale phenomena in microfluidics and nanofluidics. *Chemical engineering science*, 62(13):3443–3454, 2007.

[96] J. Hu, I. Korotkin, and S. Karabasov. A multi-resolution particle/fluctuating hydrodynamics model for hybrid simulations of liquids based on the two-phase flow analogy. *The Journal of Chemical Physics*, 149(8):084108, 2018.

[97] K. Huang. *Statistical mechanics*. Wiley, 1986.

[98] F. Höfling and S. Dietrich. Finite-size corrections for the static structure factor of a liquid slab with open boundaries. *J. Chem. Phys.*, 153(5):054119, 2020.

[99] J. H. Irving and J. G. Kirkwood. The statistical mechanical theory of transport processes. iv. the equations of hydrodynamics. *The Journal of Chemical Physics*, 18(6):817–829, 1950.

[100] B. S. Jabes, R. Klein, and L. D. Site. Structural locality and early stage of aggregation of micelles in water: An adaptive resolution molecular dynamics study. *Adv.Th.Sim.*, 1:1800025, 2018.

[101] D. Janežič, M. Praprotnik, and F. Merzel. Molecular dynamics integration and molecular vibrational theory. i. new symplectic integrators. *The Journal of chemical physics*, 122(17):174101, 2005.

[102] C. Jarzynski. Equilibrium free-energy differences from nonequilibrium measurements: A master-equation approach. *Physical Review E*, 56(5):5018, 1997.

[103] C. Jarzynski. Nonequilibrium equality for free energy differences. *Physical Review Letters*, 78(14):2690, 1997.

[104] J. K. Johnson, J. A. Zollweg, and K. E. Gubbins. The lennard-jones equation of state revisited. *Molecular Physics*, 78(3):591–618, 1993.

[105] L. Kalé, R. Skeel, M. Bhandarkar, R. Brunner, A. Gursoy, N. Krawetz, J. Phillips, A. Shinozaki, K. Varadarajan, and K. Schulten. Namd2: greater scalability for parallel molecular dynamics. *Journal of Computational Physics*, 151(1):283–312, 1999.

[106] W. Kang and U. Landman. Universality crossover of the pinch-off shape profiles of collapsing liquid nanobridges in vacuum and gaseous environments. *Physical Review Letters*, 98(6):064504, 2007.

[107] S. Karabasov, D. Nerukh, A. Hoekstra, B. Chopard, and P. V. Coveney. Multiscale modelling: approaches and challenges, 2014.

[108] G. Karniadakis, A. Beskok, and N. Aluru. *Microflows and nanoflows: fundamentals and simulation*, volume 29. Springer Science & Business Media, 2006.

[109] M. Karplus and J. Kuriyan. Molecular dynamics and protein function. *Proceedings of the National Academy of Sciences*, 102(19):6679–6685, 2005.

[110] C. Kittel and H. Kroemer. Thermal physics freeman. *San Francisco*, 1980.

[111] R. Klein, R. Ebrahimi Viand, F. Höfling, and L. Delle Site. Nonequilibrium induced by reservoirs: Physico-mathematical models and numerical tests. *Advanced Theory and Simulations*, 4:210071, 2021.

[112] P. Koumoutsakos. Multiscale flow simulations using particles. *Annu. Rev. Fluid Mech.*, 37:457–487, 2005.

[113] C. Krekeler, A. Agarwal, C. Junghans, M. Praprotnik, , and L. Delle Site. Adaptive resolution molecular dynamics technique: Down to the essential. *J.Chem.Phys.*, 149:24104, 2018.

[114] C. Krekeler, A. Agarwal, C. Junghans, M. Praprotnik, and L. Delle Site. Adaptive resolution molecular dynamics technique: Down to the essential. *The Journal of chemical physics*, 149(2):024104, 2018.

[115] C. Krekeler, A. Agarwal, C. Junghans, M. Praprotnik, and L. Delle Site. Adaptive resolution molecular dynamics technique: Down to the essential. *The Journal of Chemical Physics*, 149(2):024104, 2018.

[116] C. Krekeler and L. Delle Site. Towards open boundary molecular dynamics simulation of ionic liquids. *Physical Chemistry Chemical Physics*, 19(6):4701–4709, 2017.

[117] A. J. Ladd. Short-time motion of colloidal particles: Numerical simulation via a fluctuating lattice-boltzmann equation. *Physical Review Letters*, 70(9):1339, 1993.

[118] L. Landau, E. Lifshitz, and L. E. Reichl. Statistical physics, part 1. *Physics Today*, 34(1):74, 1981.

[119] L. D. Landau and E. M. Lifshits. *Fluid mechanics, by LD Landau and EM Lifshitz*, volume 11. Pergamon Press Oxford, UK, 1959.

[120] L. D. Landau, E. M. Lifšic, E. M. Lifshitz, and L. Pitaevskii. *Statistical physics: theory of the condensed state*, volume 9. Butterworth-Heinemann, 1980.

[121] B. H. Lavenda. *Statistical physics: A probabilistic approach*. Courier Dover Publications, 2016.

[122] J. Lebowitz and P. Bergmann. Irreversible Gibbsian ensembles. *Ann. Phys.*, 1:1, 1957.

[123] J. L. Lebowitz and P. G. Bergmann. Irreversible gibbsian ensembles. *Annals of Physics*, 1(1):1–23, 1957.

[124] E. W. Lemmon. Thermophysical properties of fluid systems. *NIST chemistry WebBook*, 1998.

[125] J. Li, D. Liao, and S. Yip. Coupling continuum to molecular-dynamics simulation: Reflecting particle method and the field estimator. *Physical Review E*, 57(6):7259, 1998.

[126] G.-R. Liu and S. S. Quek. *The finite element method: a practical course*. Butterworth-Heinemann, 2013.

[127] X. Liu and S. Zhang. Development of adaptive multi-resolution mps method for multiphase flow simulation. *Computer Methods in Applied Mechanics and Engineering*, 387:114184, 2021.

[128] L. B. Lucy. A numerical approach to the testing of the fission hypothesis. *The astronomical journal*, 82:1013–1024, 1977.

[129] M. M. Mansour, A. L. Garcia, G. C. Lie, and E. Clementi. Fluctuating hydrodynamics in a dilute gas. *Physical review letters*, 58(9):874, 1987.

[130] M. M. Mansour, C. Van den Broeck, I. Bena, and F. Baras. Spurious diffusion in particle simulations of the kolmogorov flow. *EPL (Europhysics Letters)*, 47(1):8, 1999.

[131] M. Mareschal, M. M. Mansour, G. Sonnino, and E. Kestemont. Dynamic structure factor in a nonequilibrium fluid: A molecular-dynamics approach. *Physical Review A*, 45(10):7180, 1992.

[132] S. Matysiak, C. Clementi, M. Praprotnik, K. Kremer, and L. Delle Site. Modeling diffusive dynamics in adaptive resolution simulation of liquid water. *The Journal of chemical physics*, 128(2):024503, 2008.

[133] P. Meurs, C. Van den Broeck, and A. Garcia. Rectification of thermal fluctuations in ideal gases. *Physical Review E*, 70(5):051109, 2004.

[134] R. E. Miller and E. B. Tadmor. The quasicontinuum method: Overview, applications and current directions. *Journal of Computer-Aided Materials Design*, 9(3):203–239, 2002.

[135] P. Minary, M. Tuckerman, and G. Martyna. Long time molecular dynamics for enhanced conformational sampling in biomolecular systems. *Physical review letters*, 93(15):150201, 2004.

[136] M.J.Abraham, T.Murtola, R.Schulz, S.Pall, J. Smith, B.Hess, and E.Lindahl. Gromacs: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, 1-2:19 – 25, 2015.

[137] K. Mohamed and A. Mohamad. A review of the development of hybrid atomistic–continuum methods for dense fluids. *Microfluidics and Nanofluidics*, 8(3):283–302, 2010.

[138] E. Moro. Hybrid method for simulating front propagation in reaction-diffusion systems. *Physical Review E*, 69(6):060101, 2004.

[139] M. Moseler and U. Landman. Formation, stability, and breakup of nanojets. *Science*, 289(5482):1165–1169, 2000.

[140] J. Nicolas, K. Gubbins, W. Streett, and D. Tildesley. Equation of state for the lennard-jones fluid. *Molecular Physics*, 37(5):1429–1454, 1979.

[141] X. Nie, S. Chen, M. Robbins, et al. A continuum and molecular dynamics hybrid method for micro-and nano-fluid flow. *Journal of Fluid Mechanics*, 500:55–64, 2004.

[142] X. Nie, S. Chen, and M. O. Robbins. Hybrid continuum-atomistic simulation of singular corner flow. *Physics of Fluids*, 16(10):3579–3591, 2004.

[143] X. Nie, M. O. Robbins, and S. Chen. Resolving singular forces in cavity flow: multiscale modeling from atomic to millimeter scales. *Physical review letters*, 96(13):134501, 2006.

[144] H. Noguchi, N. Kikuchi, and G. Gompper. Particle-based mesoscale hydrodynamic techniques. *EPL (Europhysics Letters)*, 78(1):10005, 2007.

[145] K. Okabe, N. Inada, C. Gota, Y. Harada, T. Funatsu, and S. Uchiyama. Intra-cellular temperature mapping with a fluorescent polymeric thermometer and fluorescence lifetime imaging microscopy. *Nat. Commun.*, 3:705, 2012.

[146] G. Oster. Brownian ratchets: Darwin's motors. *Nature*, 417(6884):25–25, 2002.

[147] S. T. O'connell and P. A. Thompson. Molecular dynamics–continuum hybrid computations: a tool for studying complex fluid flows. *Physical Review E*, 52(6):R5792, 1995.

[148] R. PATHRIA. Chapter 1 - the statistical basis of thermodynamics. In R. PATHRIA, editor, *Statistical Mechanics (Second Edition)*, pages 9–29. Butterworth-Heinemann, Oxford, second edition edition, 1996.

[149] A. Patronis and D. A. Lockerby. Multiscale simulation of non-isothermal microchannel gas flows. *Journal of Computational Physics*, 270:532–543, 2014.

[150] J. H. Peters, R. Klein, and L. Delle Site. Simulation of macromolecular liquids with the adaptive resolution molecular dynamics technique. *Phys. Rev. E*, 94:023309, Aug 2016.

[151] O. Pironneau. *Finite element methods for fluids*. Wiley Chichester, 1989.

[152] S. Poblete, M. Praprotnik, K. Kremer, and L. Delle Site. Coupling different levels of resolution in molecular simulations. *The Journal of Chemical Physics*, 132(11):114101, 2010.

[153] R. Potestio, P. Español, R. Delgado-Buscalioni, R. Everaers, K. Kremer, and D. Donadio. Monte carlo adaptive resolution simulation of multicomponent molecular liquids. *Phys. Rev. Lett.*, 111:060601, 2013.

[154] R. Potestio, S. Fritsch, P. Español, R. Delgado-Buscalioni, K. Kremer, R. Everaers, and D. Donadio. Hamiltonian adaptive resolution simulation for molecular liquids. *Phys. Rev. Lett.*, 110:108301, 2013.

[155] R. Potestio, S. Fritsch, P. Espanol, R. Delgado-Buscalioni, K. Kremer, R. Everaers, and D. Donadio. Hamiltonian adaptive resolution simulation for molecular liquids. *Physical review letters*, 110(10):108301, 2013.

[156] M. Praprotnik, L. Delle Site, and K. Kremer. Adaptive resolution molecular-dynamics simulation: Changing the degrees of freedom on the fly. *The Journal of Chemical Physics*, 123(22):224106, 2005.

[157] M. Praprotnik, L. Delle Site, and K. Kremer. Adaptive resolution molecular-dynamics simulation: Changing the degrees of freedom on the fly. *The Journal of chemical physics*, 123(22):224106, 2005.

[158] M. Praprotnik, L. Delle Site, and K. Kremer. Adaptive resolution molecular-dynamics simulation: Changing the degrees of freedom on the fly. *J. Chem. Phys.*, 123:224106, 2005.

[159] M. Praprotnik, S. Poblete, L. Delle Site, and K. Kremer. Comment on "Adaptive multiscale molecular dynamics of macromolecular fluids". *Phys. Rev. Lett.*, 107:099801, 2011.

[160] M. Praprotnik, L. D. Site, and K. Kremer. Multiscale simulation of soft matter: From scale bridging to adaptive resolution. *Annual Review of Physical Chemistry*, 59(1):545–571, 2008. PMID: 18062769.

[161] K. R. Cortes-Huerto, M. Praprotnik and L. Delle Site. From adaptive resolution to molecular dynamics of open systems. *Eur.Phys.Journ.B*, 94:189, 2021.

[162] H. Rafii-Tabar, L. Hua, and M. Cross. A multi-scale atomistic-continuum modelling of crack propagation in a two-dimensional macroscopic plate. *Journal of Physics: Condensed Matter*, 10(11):2375, 1998.

[163] M. Rao and B. Berne. On the location of surface of tension in the planar interface between liquid and vapour. *Molecular Physics*, 37(2):455–461, 1979.

[164] D. C. Rapaport. *The art of molecular dynamics simulation*. Cambridge university press, 2004.

[165] R. Rennie and J. Law. *A dictionary of physics*. Oxford University Press, 2019.

[166] K. Ritos, M. K. Borg, D. A. Lockerby, D. R. Emerson, and J. M. Reese. Hybrid molecular-continuum simulations of water flow through carbon nanotube membranes of realistic thickness. *Microfluidics and Nanofluidics*, 19(5):997–1010, 2015.

[167] J. Rowlinson and B. Widom. *Molecular Theory of Capillarity*. Dover books on chemistry. Dover Publications, 2002.

[168] D. Savio, J. Hamann, P. A. Romero, C. Klingshirn, R. Bactavatchalou, M. Dienwiebel, and M. Moseler. Multiscale friction simulation of dry polymer contacts: reaching experimental length scales by coupling molecular dynamics and contact mechanics. *Tribology Letters*, 69(2):1–16, 2021.

[169] T. Schlick. *Molecular modeling and simulation: an interdisciplinary guide*, volume 2. Springer, 2010.

[170] B. Shadrack Jabes and L. Delle Site. Nanoscale domains in ionic liquids: A statistical mechanics definition for molecular dynamics studies. *The Journal of Chemical Physics*, 149(18):184502, 2018.

[171] B. Shadrack Jabes, C. Krekeler, R. Klein, and L. Delle Site. Probing spatial locality in ionic liquids with the grand canonical adaptive resolution molecular dynamics technique. *J.Chem.Phys.*, 148:193804, 2018.

[172] A. Shakeel, H. Mahmood, U. Farooq, Z. Ullah, S. Yasin, T. qbal, C. Chassagne, and M. Moniruzzaman. Rheology of pure ionic liquids and their complex fluids: A review. *ACS Sustainable Chem. Eng.*, 7:13586–13626, 2019.

[173] N. Sharma and N. A. Patankar. Direct numerical simulation of the brownian motion of particles by using fluctuating hydrodynamic equations. *Journal of Computational Physics*, 201(2):466–486, 2004.

[174] L. D. Site and M. Praprotnik. Molecular systems with open boundaries: Theory and simulation. *Physics Reports*, 693:1–56, 2017. Molecular systems with open boundaries: Theory and Simulation.

[175] J. A. Smirnova, L. V. Zhigilei, and B. J. Garrison. A combined molecular dynamics and finite element method technique applied to laser induced pressure wave propagation. *Computer physics communications*, 118(1):11–16, 1999.

[176] T. M. Squires and S. R. Quake. Microfluidics: Fluid physics at the nanoliter scale. *Reviews of modern physics*, 77(3):977, 2005.

[177] M. O. Steinhauser. Introduction to molecular dynamics simulations: Applications in hard and soft condensed matter physics. *Molecular dynamics-Studies of synthetic and biological macromolecules*, pages 3–28, 2012.

[178] A. Stirling, N. N. Nair, A. Lledós, and G. Ujaque. Challenges in modelling homogeneous catalysis: new answers from ab initio molecular dynamics to the controversy over the wacker process. *Chemical Society Reviews*, 43(14):4940–4952, 2014.

[179] W. C. Swope, H. C. Andersen, P. H. Berens, and K. R. Wilson. A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters. *The Journal of chemical physics*, 76(1):637–649, 1982.

[180] T.-R. Teschner, L. Könözsy, and K. W. Jenkins. Progress in particle-based multiscale and hybrid methods for flow applications. *Microfluidics and Nanofluidics*, 20(4):68, 2016.

[181] S. Thaler, M. Praprotnik, and J. Zavadlav. Back-mapping augmented adaptive resolution simulation. *J. Chem. Phys.*, 153:164118, 2020.

[182] I. G. Tironi, R. Sperb, P. E. Smith, and W. F. van Gunsteren. A generalized reaction field method for molecular dynamics simulations. *The Journal of chemical physics*, 102(13):5451–5459, 1995.

[183] B. D. Todd, D. J. Evans, and P. J. Daivis. Pressure tensor for inhomogeneous fluids. *Phys. Rev. E*, 52:1627–1638, Aug 1995.

[184] M. Tuckerman. *Statistical mechanics: theory and molecular simulation*. Oxford university press, 2010.

[185] M. J. Turner, R. W. Clough, H. C. Martin, and L. Topp. Stiffness and deflection analysis of complex structures. *journal of the Aeronautical Sciences*, 23(9):805–823, 1956.

[186] W. F. van Gunsteren, S. Billeter, A. Eising, P. Hünenberger, P. Krüger, A. Mark, W. Scott, and I. Tironi. Biomolecular simulation: the gromos96 manual and user guide. *Vdf Hochschulverlag AG an der ETH Zürich, Zürich*, 86:1–1044, 1996.

[187] F. Varnik, J. Baschnagel, and K. Binder. Molecular dynamics results on the pressure tensor of polymer films. *The Journal of Chemical Physics*, 113(10):4444–4453, 2000.

[188] L. Verlet. Computer" experiments" on classical fluids. i. thermodynamical properties of lennard-jones molecules. *Physical review*, 159(1):98, 1967.

[189] H. K. Versteeg and W. Malalasekera. *An introduction to computational fluid dynamics: the finite volume method*. Pearson education, 2007.

[190] N. K. Voulgarakis and J.-W. Chu. Bridging fluctuating hydrodynamics and molecular dynamics simulations of fluids. *The Journal of chemical physics*, 130(13):04B605, 2009.

[191] N. K. Voulgarakis and J.-W. Chu. Bridging fluctuating hydrodynamics and molecular dynamics simulations of fluids. *The Journal of Chemical Physics*, 130(13):134111, 2009.

[192] D. WADSWORTH and D. ERWIN. One-dimensional hybrid continuum/particle simulation approach for rarefied hypersonic flows. In *5th Joint Thermophysics and Heat Transfer Conference*, page 1690, 1990.

[193] J. H. Walther, M. Praprotnik, E. M. Kotsalis, and P. Koumoutsakos. Multiscale simulation of water flow past a c540 fullerene. *Journal of Computational Physics*, 231(7):2677–2681, 2012.

[194] J. Walton, D. Tildesley, J. Rowlinson, and J. Henderson. The pressure tensor at the planar surface of a liquid. *Molecular Physics*, 48(6):1357–1368, 1983.

[195] H. Wang, C. Hartmann, C. Schütte, and L. Delle Site. Grand-canonical-like molecular-dynamics simulations by using an adaptive-resolution technique. *Phys. Rev. X*, 3:011018, Mar 2013.

[196] H. Wang, C. Schütte, and L. D. Site. Adaptive resolution simulation (adress): A smooth thermodynamic and structural transition from atomistic to coarse grained resolution and vice versa in a grand canonical fashion. *J. Chem. Th.Comp.*, 8:2878, 2012.

[197] J. Whittaker and L. Delle Site. Investigation of the hydration shell of a membrane in an open system molecular dynamics simulation. *Physical Review Research*, 1(3):033099, 2019.

[198] J. Whittaker and L. Delle Site. Investigation of the hydration shell of a membrane in an open system molecular dynamics simulation. *Phys.Rev.Res.*, 1:033099, 2019.

[199] B. Widom. Some topics in the theory of fluids. *J.Chem.Phys.*, 39:2808–2812, 1963.

[200] H. S. Wijesinghe and N. G. Hadjiconstantinou. Discussion of hybrid atomistic-continuum methods for multiscale hydrodynamics. *International Journal for Multiscale Computational Engineering*, 2(2), 2004.

[201] S. A. Williams. *A multiscale hybrid algorithm for fluctuating hydrodynamics*. University of California, Davis, 2007.

[202] S. A. Williams, J. B. Bell, and A. L. Garcia. Algorithm refinement for fluctuating hydrodynamics. *Multiscale Modeling & Simulation*, 6(4):1256–1280, 2008.

[203] Y. Zhao. Brief introduction to the thermostats. Technical report, Tech. Rep. Research-Cerca con Google, 2011.