SISSA
Scuola
Internazionale
Superiore di
Studi Avanzati

Physics Area - PhD course in
Theory and Numerical Simulation of Condensed Matter

Green-Kubo simulation of transport properties: from *ab initio* to neural-network molecular dynamics

Candidate:
Davide Tisi

Advisor:
Stefano Baroni

Academic Year 2021-22

Contents

1

Introduction

Transport coefficients are archetypal examples of off-equilibrium properties, related to the flow of a conserved quantity, such as energy, momentum, or charge. They are an essential element for the understanding of many physical phenomena and technological applications, such as planetary evolution, energy saving, thermal dissipation, heat management in devices, study of biological processes and chemical reactions.

The microscopic description of the relaxation of small off-equilibrium fluctuations and the response of systems to small perturbations have a rigorous theoretical foundation in the Green–Kubo (GK) theory of linear response [1–4]. The transport coefficients are found to be proportional to the integral of the time auto-correlation function (tACF) of some macroscopic currents computed at equilibrium, which can be evaluated from molecular dynamics (MD) simulations.

Density functional theory (DFT) has established itself as the golden standard for the quantum mechanical modelling of materials. It enables to compute interatomic forces entirely from first principles, starting only from the chemical compositions and the fundamental laws of nature. The combination of DFT with classical MD, in both Born-Oppenheimer [5, 6] or Car-Parrinello [6, 7] formulation, has a considerable impact in many fields, such as chemistry and material science. Despite the apparent simplicity and elegance of the GK and the power of DFT based methods, their combination for the computation of thermal transport coefficients has been deemed to be impossible [8–10]. This alleged incompatibility had roots in the misconception that the intrinsic indeterminacy of the decomposition of the total energy in local contributions would undermine the uniqueness of the transport coefficients that are derived from them. This conundrum has been recently overcome, thanks to the introduction of a so-called *gauge invariance* principle [8], which allowed Marcolongo, Umari and Baroni (MUB) to derive a general DFT expression for the adiabatic energy flux.

The MUB approach made *ab initio* simulations of heat transport possible for disordered systems, like liquids and glasses, albeit at the price of lengthy and costly simulations. Progress in statistical techniques for the analysis of the flux time series [11, 12] made possible to achieve 10% accuracy in the calculated thermal conductivity with simulations of a few dozen to a few hundred picoseconds. Still, the computational burden of *ab initio* MD (AIMD), based on on-the-fly generation

of a potential energy surface (PES) from DFT, is heavy and requires access to high performance computer platforms for substantial wall-clock times.

In the last decade, a combination of standard electronic-structure methods, based on DFT, and new machine-learning techniques have allowed the construction of inter-atomic potentials possessing quantum mechanical accuracy at a cost that is only marginally higher than that of classical force fields. All the machine learned potentials, which are represented either by a deep-neural network (DNN) [13–16] or by a Gaussian-process [17], use a local decomposition of the total potential energy of the system in terms of atomic contributions, which makes straightforward to define the energy flux, or current, from which to compute the heat conductivity via GK theory. We made extensive use of the recently developed deep potential (DP) framework [16, 18]. DP molecular dynamics (DPMD) simulations have been used successfully to study bulk thermodynamic properties beyond the reach of direct DFT calculations [19–25], as well as dynamic properties like mass diffusion in solid state electrolytes [26, 27], infrared spectra of water and ice [28].

In our work, we focus on the study of water. It is composed of two of the three most abundant elements in the universe, making it a quintessential constituent of many planets and moons, and an essential medium for the life on Earth [29–31]. Any attempts at a quantitative evolutionary model of water-rich celestial bodies rely on accurate knowledge of the heat and electrical conductivities of some phase of water at high-pT conditions. However, despite its apparent chemical simplicity, water is a very complex liquid with a rich phase diagram. Although of its importance many of its properties are poorly understood [32, 33]. In our study, we focus on the computation of thermal conductivity and viscosity of water and we present results obtained from both AIMD and DPMD. We show that properly trained neural-network potentials (NNP) are powerful tools for an extensive investigation of transport properties of materials. They allowed us to overcome the limitations of AIMD, that have limited the study of transport properties. For example, the viscous properties of water has always been avoided by any study based on DFT [34–42], because *an accurate computation of the viscosity of water would require exceedingly long first-principles simulations* [40].

The thesis is organized as follows. Chapter 2 introduces the Green-Kubo theory of linear response of general conserved quantities, in Chapter 3 we focus on thermal transport and derive the formulation of energy flux from a classical force field. We, also, introduce the invariance principles for the thermal transport coefficient. In Chapter 4 we describe the data analysis techniques, called *cepstral analysis*, used to compute accurate and reliable transport coefficients from MD trajectories. Chapter 5 contains the derivation of the MUB current and an overview of its implementation in the open-source code `QEHeat` [43]. In Chapter 6, we performed AIMD simulations of liquid water and ice Ih within the PBE generalized gradient

approximation (GGA) [44]. Then, we show our results of thermal conductivity from the DFT adiabatic energy flux using `QEHeat` . Chapter 7 contains a brief presentation of the DP framework. In Chapter 8 we show the results for the thermal conductivity of water obtained from a DP model. Our DNN methodology is first validated by a comparison with the results from AIMD simulations of liquid water at the PBE level of theory [44]. Then, we take advantage of the simple expression for the energy current with neural-network potentials to compute the thermal transport coefficients of liquid water with a complex DFT approximation, the *strongly constrained and appropriately normed* (SCAN) meta-GGA. Chapter 9 contains the results for the viscosity of water, both from AIMD and DPMD. In Chapter 10, we describe a recent extension of the DP model, which includes also the long-range electrostatic contributions, and its application to the computation of electrical conductivity in water at high-pT, where a short range models fails to reproduce the correct *ab initio* results. Conclusions are drawn in Chapter 11.

List of publications

The original work presented in this thesis is based on the following publications and preprint:

[A] A. Marcolongo, R. Bertossa, **D. Tisi**, and S. Baroni, *QEHeat: An open-source energy flux calculator for the computation of heat-transport coefficients from first principles*, Computer Physics Communications 269,108090 (2021);

[B] **D. Tisi**, L. Zhang, R. Bertossa, H. Wang, R. Car, and S. Baroni, *Heat transport in liquid water from first-principles and deep neural network simulations*, Physical Review B 104, 224202 (2021);

[C] C. Malosso, L. Zhang, R. Car, S. Baroni, and **D. Tisi**, *Viscosity in water from first-principles and deep-neural-network simulations*, arXiv:2203.01262 (2022).

2

Green Kubo Theory

The Green-Kubo (GK) theory of linear response is a general theoretical framework for the microscopic description of transport properties. It is rooted in the work of Onsager [45, 46] culminated, twenty years later, with the Green-Kubo equations [1–4]. These equations are a consequence of the fluctuation-dissipation theorem and establish a relation between (non-equilibrium) transport coefficients, called Onsager coefficients, and the time auto-correlation function of the relevant currents at equilibrium.

2.1 Hydrodynamic variables

Extensive variables have the property that the value they have on a system equals the sum of their values on each of its subsystem. Thus, an extensive variable, A, can be written as the integral of a suitably defined density, $a(\boldsymbol{r})$:

$$A[\Omega, t] = \int_{\Omega} a(\boldsymbol{r}, t) d\boldsymbol{r} \tag{2.1}$$

where Ω is the volume of the system. For a locally conserved extensive quantity a current density, $\boldsymbol{\jmath}(\boldsymbol{r}, t)$, can be defined and associated to its conserved density thought the continuity equation:

$$\frac{\partial a(\boldsymbol{r}, t)}{\partial t} = -\nabla \cdot \boldsymbol{\jmath}(\boldsymbol{r}, t), \tag{2.2}$$

where $\nabla\cdot$ represent the divergence. In the following, the densities of conserved quantities will be denote as *conserved densities*. Applying the space Fourier transform to Eq. (2.2):

$$\dot{\tilde{a}}\left(\boldsymbol{q}, t\right) = -i\boldsymbol{q} \cdot \tilde{\boldsymbol{j}}\left(\boldsymbol{q}, t\right), \tag{2.3}$$

where the dot indicates a time derivative and the tilde a Fourier transform. Eq. (2.3) suggests that the longer the wavelength, the slower is the dynamics of a conserved density. Thus, conserved densities are adiabatically decoupled from all the fast atomic degrees of freedom. These long-wavelength Fourier components of conserved densities are called *hydrodynamic variables*.

Phenomenological equations

Let us now consider a system with Q of conserved extensive variables $\{A^1, A^2, ..., A^Q\}$. For instance, in the case of a one-component fluid, $Q = 5$ and $\{A^1, A^2, ..., A^5\}$ are, respectively, the mass (*i.e.* particle number), the energy and the three components of the momentum. Without loss of generality, we can set the average values of the conserved quantities to zero, $\langle A^i \rangle = 0$, so that their densities, $a^i(\boldsymbol{r}, t)$, directly describe the deviation from equilibrium. We indicate by $\boldsymbol{\jmath}^i(\boldsymbol{r}, t)$ the corresponding currents. At equilibrium, all conserved densities and currents vanish. Off equilibrium, it will be assumed that the wavelength and the time scale of the disturbances are so long that the assumption of local thermal equilibrium is valid. Assuming small deviations from equilibrium, we can express the time derivatives of conserved densities as a linear combination of the densities themselves:

$$\dot{a}^i(\boldsymbol{r}, t) = \sum_{j=1}^{Q} \int \Lambda^{ij}(\boldsymbol{r} - \boldsymbol{r}', t - t')\, a^j(\boldsymbol{r}', t')\, d\boldsymbol{r}' dt' \tag{2.4}$$

where Λ is a suitable coefficient that depends only on $\boldsymbol{r} - \boldsymbol{r}'$ and $t - t'$ due to respectively space translation and time translation symmetry of the system. Exploiting the convolution theorem, the space-time Fourier transform of Eq. (2.4) reads:

$$-i\omega \tilde{\tilde{a}}^i(\boldsymbol{q}, \omega) = \sum_{j=1}^{Q} \tilde{\tilde{\Lambda}}^{ij}(\boldsymbol{q}, \omega)\, \tilde{\tilde{a}}^j(\boldsymbol{q}, \omega)\,, \tag{2.5}$$

where the double tildas correspond to the space-time Fourier transforms. By combining the last set of equations with the time Fourier transform of Eq. (2.3), one obtains the *constitutive equations*:

$$-i\boldsymbol{q} \cdot \tilde{\tilde{\boldsymbol{\jmath}}}^i(\boldsymbol{q}, \omega) = \sum_{j=1}^{Q} \tilde{\tilde{\Lambda}}^{ij}(\boldsymbol{q}, \omega)\, \tilde{\tilde{a}}^j(\boldsymbol{q}, \omega)\,, \tag{2.6}$$

and for the longitudinal component of the conserved currents, it holds:

$$\tilde{\tilde{\boldsymbol{\jmath}}}^i(\boldsymbol{q}, \omega) = i\frac{\boldsymbol{q}}{q^2} \sum_{j=1}^{Q} \tilde{\tilde{\Lambda}}^{ij}(\boldsymbol{q}, \omega)\, \tilde{\tilde{a}}^j(\boldsymbol{q}, \omega)\,. \tag{2.7}$$

In isotropic media, the $\tilde{\tilde{\Lambda}}$s are functions of $q = |\boldsymbol{q}|$, whereas $\tilde{\tilde{\Lambda}}^{ij}(q = 0) = 0$. In fact a non-vanishing value would imply a long-range dependence of the currents on density fluctuations, in contrast with the assumption of local thermodynamic equilibrium. Thus, the long-wavelength low-frequency limit of the coupling constants can thus be assumed quadratic in q:

$$\tilde{\tilde{\Lambda}}^{ij}(\boldsymbol{q}, \omega) \sim q^2 \lambda^{ij}. \tag{2.8}$$

Then inserting this last expression in Eq. (2.7) one obtains

$$\tilde{\tilde{\boldsymbol{j}}}^i(\boldsymbol{q},\omega) = i\boldsymbol{q}\sum_{j=1}^{Q}\lambda^{ij}\tilde{\tilde{a}}^j(\boldsymbol{q},\omega), \tag{2.9}$$

valid in the long-wavelength low-frequency regime. Defining the flux as the macroscopic average of $\boldsymbol{\jmath}^i$:

$$\boldsymbol{J}^i = \frac{1}{\Omega}\int_\Omega \boldsymbol{\jmath}^i(\boldsymbol{r})\,d\boldsymbol{r}; \tag{2.10}$$

and the corresponding components of the macroscopic density gradients $\boldsymbol{D}^i$ as:

$$\boldsymbol{D}^i = \frac{1}{\Omega}\int_\Omega \boldsymbol{\nabla} a^i(\boldsymbol{r})\,d\boldsymbol{r}, \tag{2.11}$$

we can relate explicitly these last two quantities through Eq. (2.9) and obtain:

$$\boldsymbol{J}^i = \sum_{j=1}^{Q}\lambda^{ij}\boldsymbol{D}^j. \tag{2.12}$$

Let $\alpha^i = \frac{\partial S}{\partial A^i}$ be the intensive variable conjugate to the extensive variable A^i, where $S = S(A)$ is the system's entropy, and $\chi^{ij} = \frac{1}{\Omega}\frac{\partial A^i}{\partial \alpha^j}$ the corresponding susceptibility. Under the assumption of local thermal equilibrium, local values of intensive variables $\alpha^i(\boldsymbol{r})$ can be defined and the normalized integrals of their gradients are called *thermodynamic forces*:

$$\boldsymbol{F}^i = \frac{1}{\Omega}\int_\Omega \boldsymbol{\nabla}\alpha^i(\boldsymbol{r})\,d\boldsymbol{r}. \tag{2.13}$$

The thermodynamic forces can be related to the macroscopic density gradients through the susceptibilities:

$$\boldsymbol{D}^i = \sum_{j=1}^{Q}\chi^{ij}\boldsymbol{F}^j, \tag{2.14}$$

and inserting this relation into Eq. (2.12), one gets the *phenomenological equations*:

$$\boldsymbol{J}^i = \sum_{j=1}^{Q}L^{ij}\boldsymbol{F}^j, \tag{2.15}$$

where

$$L^{ij} = \sum_{k=0}^{Q}\lambda^{ik}\chi^{kj}, \tag{2.16}$$

are the *Onsager's coefficients* (or *phenomenological coefficients*). Eq. (2.15) states a linear relation between the macroscopic conserves currents $\boldsymbol{J}$ and the thermodynamic forces $\boldsymbol{F}$. In particular the Onsager's coefficients satisfy the so-called Onsager's reciprocal relations [45–47], derived from microscopic reversibility:

$$L^{ij} = L^{ji}, \tag{2.17}$$

that shows the equality of the cross-coefficients.

In the case of heat transport the conserved quantity is the energy and the associated thermodynamic force is the gradient of the inverse of the temperature. Thus, in the simplest case where we consider only one conserved quantity ($Q = 1$), we obtain the renowned Fourier's law of thermal conduction [48]:

$$\boldsymbol{J}^E = -\kappa \nabla T, \tag{2.18}$$

where κ is the thermal conductivity and T is the temperature.

2.2 Linear-Response Theory

In order to evaluate the phenomenological coefficients appearing in Eq. (2.15), we introduce the linear-response theory. First of all we consider a classical system of N interacting atoms described by the Hamiltonian H^0:

$$H^0(\Gamma) = \sum_{i=1}^{N} \frac{1}{2M_n} \boldsymbol{P}_n^2 + \Phi(\boldsymbol{R}_1, \boldsymbol{R}_2, ..., \boldsymbol{R}_N), \tag{2.19}$$

where M_n, $\boldsymbol{P}_n$ and $\boldsymbol{R}_n$ are respectively the mass, the momentum and the position of the n-th particle, while $\Gamma = \{\boldsymbol{R}_n, \boldsymbol{P}_n\}$ indicates the phase-space coordinates of the entire system, and Φ is a generic many-body potential that depends on the atomic positions.

Let us suppose that the system is subjected to a external potential that couples to a linear combination of conserved densities, $\{a^i(\boldsymbol{r}; \Gamma)\}$, as:

$$\Phi'(\Gamma, t) = \sum_i \int_\Omega v^i(\boldsymbol{r}, t)\, a^i(\boldsymbol{r}; \Gamma)\, d\boldsymbol{r}, \tag{2.20}$$

where $\{v^i(\boldsymbol{r}, t)\}$ are time-dependent fields and $\{a^i(\boldsymbol{r}; \Gamma)\}$ are space-phase observables. The conserved densities are theirs ensemble averages:

$$\begin{aligned} a^i(\boldsymbol{r}) &= \langle a^i(\boldsymbol{r}, \Gamma) \rangle_{eq} \\ &= \int a^i(\boldsymbol{r}; \Gamma)\, \mathcal{P}^0(\Gamma)\, d\Gamma. \end{aligned} \tag{2.21}$$

$\mathcal{P}^0(\Gamma)$ is the equilibrium distribution function:

$$\mathcal{P}^0(\Gamma) \propto e^{-\frac{H^0(\Gamma)}{k_B T}}, \tag{2.22}$$

where k_B is the Boltzmann constant and T is the temperature of the system. We suppose that at $t = -\infty$ the system is at equilibrium with respect to the Hamiltonian H^0, implying the coupling fields to be such that $v^i(\boldsymbol{r}, t = -\infty) = 0$. The conserved currents are ensemble averages of the phase-space observables:

$$j^i(\boldsymbol{r}) = \langle j^i(\boldsymbol{r}, \Gamma) \rangle_{eq}. \tag{2.23}$$

Let $\Gamma'_t = \{\boldsymbol{R}_n(t), \boldsymbol{P}_n(t)\}$ be a dynamical trajectory of the system, driven by the perturbed Hamiltonian $H'(t) = H^0 + \Phi'(t)$, through the Hamilton's equations of motions:

$$\begin{aligned} \dot{\boldsymbol{R}}_n(t) &= \frac{\partial H'}{\partial \boldsymbol{P}_n}\Big|_t, \\ \dot{\boldsymbol{P}}_n(t) &= -\frac{\partial H'}{\partial \boldsymbol{R}_n}\Big|_t, \end{aligned} \tag{2.24}$$

whose solutions are uniquely determined by the initial conditions $\Gamma(t = 0) = \Gamma_0$. When a phase-space observable is evaluated along a dynamical trajectory, Γ_t, it will depend on time, through the positions and momenta, as well as on the initial conditions. Averaging with respect to the initial conditions will result in a time dependent expectation value for the conserved densities (or currents):

$$\begin{aligned} a^i(\boldsymbol{r}, t) &= \langle a^i\left(\boldsymbol{r}, \Gamma'_t\right) \rangle_{eq} \\ &= \int a^i\left(\boldsymbol{r}; \Gamma'_t\right) \mathcal{P}^0(\Gamma_0)\, d\Gamma_0 \end{aligned} \tag{2.25}$$

where the dependence on Γ'_t stresses that the dynamic is driven by the perturbed Hamiltonian. According to the Green-Kubo theory [1, 2, 49] the linear response to a perturbation of the α component of a conserved current $\boldsymbol{j}^i$ is:

$$j^i_\alpha(\boldsymbol{r}, t) = \frac{1}{k_B T} \sum_j \int_{-\infty}^{t} dt' \int_\Omega d\boldsymbol{r}' \left\langle j^i_\alpha(\boldsymbol{r}, \Gamma_t)\, \dot{a}^j(\boldsymbol{r}', \Gamma_{t'}) \right\rangle_{eq} v^j(\boldsymbol{r}', t'). \tag{2.26}$$

By combining Eq. (2.26) with the continuity equation and integrating by parts, one obtains:

$$j^i_\alpha(\boldsymbol{r}, t) = \frac{1}{k_B T} \sum_j \sum_\beta \int_{-\infty}^{t} dt' \int_\Omega d\boldsymbol{r}' \left\langle j^i_\alpha(\boldsymbol{r}, \Gamma_t)\, j^i_\beta(\boldsymbol{r}', \Gamma_{t'}) \right\rangle_{eq} \frac{\partial}{\partial r'_\beta} v^j(\boldsymbol{r}', t'), \tag{2.27}$$

where r_β is the β component of the position vector. Thanks to space-time homogeneity and isotropy, the last equation can be averaged to recover the macroscopic flux as in Eq. (2.15).

$$J_\alpha^i(\Gamma_t) = \frac{1}{\Omega}\int_\Omega j_\alpha^i\left(\boldsymbol{r},\Gamma_t\right)d\boldsymbol{r}, \tag{2.28}$$

$$F_\alpha^i(\Gamma_t) = \frac{1}{\Omega T}\int \frac{\partial}{\partial r_\alpha} v^i\left(\boldsymbol{r},\Gamma_t\right)d\boldsymbol{r}, \tag{2.29}$$

$$L_{\alpha\beta}^{ij}(\Gamma_t) = \frac{\Omega}{k_B}\int_0^\infty \Big\langle J_\alpha^i\left(\Gamma_t\right) J_\beta^j\left(\Gamma_0\right)\Big\rangle_{eq} dt. \tag{2.30}$$

Eq. (2.30) is the famous Green-Kubo formula for transport coefficients from classical linear-response theory, it is a powerful and elegant tool to evaluate the Onsager's coefficients from equilibrium molecular dynamics. The Onsager's reciprocity relations Eq. (2.17) can be recovered combining Eq. (2.30) with time-translational invariance $\langle J_\alpha^i(\Gamma_t)J_\beta^i(\Gamma_0)\rangle = \langle J_\alpha^i(\Gamma_0)J_\beta^i(\Gamma_{-t})\rangle$ and micro-reversibility $\langle J_\alpha^i(\Gamma_t)J_\beta^i(\Gamma_0)\rangle = \langle J_\alpha^i(\Gamma_{-t})J_\beta^i(\Gamma_0)\rangle$.

3

Heat Transport

Numerical studies of heat transport at the molecular scale often rely on Boltzmann's kinetic approach [50–53]. This is adequate when the relaxation processes are dominated by binary collisions, as in the case of dilute gases of particles, such as atoms or molecules, or of quasiparticles, such as phonons in crystalline solids. A different microscopic description is needed for high temperature solids, where the anharmonic effects eventually make the quasiparticle picture break down [54], or glasses and liquids where phonon dispersions are not even defined. A more general approach to simulate the heat transport coefficients is provided by molecular dynamics (MD). Non-equilibrium MD [3, 4, 55, 56] directly exploits the Fourier law and applies straightforwardly to finite systems and interfaces, but suffers from severe practical difficulties, such as finite-size and non-linear effects. We will focus on equilibrium MD in combination with the Green-Kubo (GK) theory of linear response [1–4].

3.1 Green-Kubo theory of linear response

Heat transport is determined by temperature gradients which cannot be described by any mechanical perturbation. The concept of temperature distribution implies a local thermal equilibrium over lengths and times bigger than typical atomic distances and relaxation times. Let us now suppose that the system of volume Ω can be subdivided into subsystems, $\{\Omega_1, \Omega_2, \dots \Omega_k\}$, indicating with Γ_k the phase-space variables of the $k-$th subsystem. Local equilibrium means that the phase-space variables of different sub-systems are weakly correlated, thus the total probability density can be factorized:

$$\begin{aligned} P(\Gamma_1, \Gamma_2, \dots) &\approx P(\Gamma_1) P(\Gamma_2) \\ &\propto e^{-\sum_k \frac{\hat{H}_k}{k_B T_k}} \end{aligned} \tag{3.1}$$

where H_k and T_k are respectively the Hamiltonian and the temperature of the the $k-$th subsystem. Defining the two continuous functions $\hat{e}(\boldsymbol{r}, \Gamma)$, as the energy density, and $T(\boldsymbol{r})$ as the local temperature. We can assume small variations of the

temperature:

$$T(\boldsymbol{r}) = T^0 + \Delta T(\boldsymbol{r}), \tag{3.2}$$

where T^0 is the average temperature of the system and $|\Delta T(\boldsymbol{r})| \ll T$. At the first order in ΔT, the probability density becomes:

$$P(\Gamma) = e^{-\frac{1}{k_B T^0}\left(\hat{H}^0(\Gamma) - \int_\Omega \hat{e}(\boldsymbol{r},\Gamma)\frac{\Delta T(\boldsymbol{r})}{T^0} d\boldsymbol{r}\right)} \tag{3.3}$$

where $\hat{H}^0(\Gamma)+\Phi'(\Gamma)$ is the perturbed Hamiltonian and $\Phi'(\Gamma) = -\int_\Omega \hat{e}(\boldsymbol{r},\Gamma)\frac{\Delta T(\boldsymbol{r})}{T^0} d\boldsymbol{r}$ the perturbation, equivalent to Eq. (2.20). In a system where the only non-trivial conserved quantity is the energy, the constitutive equation that couples the energy flow to temperature gradients is:

$$J^E_\alpha = -\kappa_{\alpha\beta}\nabla_\beta T(\boldsymbol{r}). \tag{3.4}$$

$\kappa_{\alpha\beta}$ is the thermal conductivity tensor, defined as:

$$\kappa_{\alpha\beta} = \frac{L^{EE}_{\alpha\beta}}{T^2} \tag{3.5}$$

$$\kappa_{\alpha\beta} = \frac{\Omega}{k_B T^2}\int_0^\infty \left\langle J^E_\alpha(\Gamma_t)\, J^E_\beta(\Gamma_0)\right\rangle_{eq} dt. \tag{3.6}$$

where $L^{EE}_{\alpha\beta}$ are the Onsager's coefficients and J^E_α the macroscopic energy flux:

$$J^E_\alpha(\Gamma) = \frac{1}{\Omega}\int_\Omega \jmath^E_\alpha(\boldsymbol{r},\Gamma)\, d\boldsymbol{r}. \tag{3.7}$$

$\jmath^E_\alpha(\boldsymbol{r})$ is the current energy density, defined by the continuity equation, Eq. (2.2), for $e(\boldsymbol{r},\Gamma)$:

$$\frac{\partial e(\boldsymbol{r},\Gamma_t)}{\partial t} = -\nabla\cdot\boldsymbol{\jmath}^E(\boldsymbol{r},\Gamma_t). \tag{3.8}$$

In isotropic systems the thermal transport coefficient is a multiple of the identity, $\kappa_{\alpha\beta} = \kappa\delta_{\alpha\beta}$ where:

$$\kappa = \frac{1}{3}\sum_{\alpha=x,y,z}\kappa_{\alpha\alpha} \tag{3.9}$$

$$= \frac{\Omega}{3k_B T^2}\int_0^\infty \left\langle \boldsymbol{J}^E(\Gamma_t)\cdot\boldsymbol{J}^E(\Gamma_0)\right\rangle_{eq}, \tag{3.10}$$

analogously an isotropic Onsager coefficient, L^{EE}, can be defined.

To conclude the section, we derive an explicit relation for the energy current

from a microscopic expression for the energy density. Let's take the Taylor expansion of the spatial Fourier transform of Eq. (3.8):

$$\frac{d\tilde{e}(\boldsymbol{q}=\boldsymbol{0},\Gamma_t)}{dt} + \nabla_{\boldsymbol{q}} \frac{d\tilde{e}(\boldsymbol{q},\Gamma_t)}{dt}|_{\boldsymbol{q}=\boldsymbol{0}} = -i\boldsymbol{q}\cdot\tilde{\boldsymbol{\jmath}}^E(\boldsymbol{q}=\boldsymbol{0},\Gamma_t) + \mathcal{O}(q^2) = 0 \tag{3.11}$$

$$\frac{d}{dt}\int_\Omega e(\boldsymbol{r},\Gamma_t)d\boldsymbol{r} - i\boldsymbol{q}\cdot\left[\int_\Omega \dot{e}\left(\boldsymbol{r},\Gamma_t\right)\boldsymbol{r}d\boldsymbol{r} - \int_\Omega \boldsymbol{\jmath}^E(\boldsymbol{r},\Gamma_t)d\boldsymbol{r}\right] + \mathcal{O}(q^2) = 0. \tag{3.12}$$

in the thermodynamic limit, the first term of Eq. (3.12) is zero because the time-derivative of the total energy. Thus, at the first order in q the term inside the square brackets is zero, allowing me to write J^E in terms of derivatives of the energy density:

$$\boldsymbol{J}^E = \frac{1}{\Omega}\int_\Omega \dot{e}\left(\boldsymbol{r},\Gamma_t\right)\boldsymbol{r}d\boldsymbol{r} \tag{3.13}$$

$$= \frac{1}{\Omega}\int_\Omega d\boldsymbol{r}\boldsymbol{r}\sum_n\left(\frac{\partial e(\boldsymbol{r},\Gamma_t)}{\partial \boldsymbol{R}_n}\cdot\boldsymbol{V}_n + \frac{\partial e(\boldsymbol{r},\Gamma_t)}{\partial \boldsymbol{P}_n}\cdot\boldsymbol{F}_n\right), \tag{3.14}$$

where the sum over n correspond to a sum over the atoms of the system, $\boldsymbol{F}_n = \frac{\partial \boldsymbol{P}_n}{\partial t}$ is the force acting on n-th atom, and $\boldsymbol{V}_n = \frac{\boldsymbol{P}_n}{M_n}$ its velocity. Eq. (3.14) is ill-defined in PBC, this issue will be resolved once we will derive the exact expression of $\boldsymbol{J}^E$ from either a classical force field or DFT.

3.2 Green-Kubo *vs.* DFT: a long-lived misconception

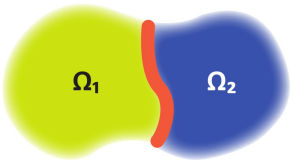
A long-lasting obstacle at the combination of GK theory and DFT *ab initio* methods was the inherent indeterminacy of any quantum mechanical expression for the energy density. In classical MD the computation of the energy current relies on the fact that the total energy of an extended system can be decomposed into localised contributions (local energies). In a quantum-mechanical setting, it is not possible to uniquely define atomic energies and the total energy of a system can at most be expressed in terms of an energy *density*, which is also ill-defined. For instance, the electrostatic energy of a continuous charge-density distribution can be expressed as either one half the integral of the density times the potential, or of $\frac{1}{8\pi}$ the squared modulus of the field. For this reason, it has long been feared that *"the Green-Kubo relation does not serve our purposes* [of computing the thermal conductivity] *because in first-principles calculations it is impossible to uniquely decompose the total energy into individual contributions from each atom"* [57].

Actually, although not generally fully appreciated, the same indeterminacy

arises with classical force fields as well, because classical atomic energies themselves are ill-defined. For example for pair-wise interactions any different partition of the interaction energy of the nm pair into individual atomic contributions would be equally acceptable and, yet, would lead to a different expression for the energy flux [58]. This long-standing problem has been recently solved with the introduction of a *gauge invariance* principle for the transport coefficients [8–10, 58].

In order to introduce, and understand, the recently discovered *gauge* and *convective* invariance principles for the transport coefficients, it is helpful to define the concept of *diffusive* flux. A flux is said to be diffusive if its GK integral, as defined in Eq. (3.6), is different from zero; the flux is said to be non-diffusive otherwise.

3.3 Gauge invariance



$$E(\Omega_1 \cup \Omega_2) = E(\Omega_1) + E(\Omega_2) + W_{12}$$
$$\stackrel{?}{=} \mathcal{E}(\Omega_1) + \mathcal{E}(\Omega_2)$$

Figure 3.1: The energy of an isolated system is the sum of the energies of its (isolated) subsystems, $E(\Omega_i)$ plus their interaction, W_{12}. In the thermodynamic limit, while $E(\Omega_i)$ are proportional to the volume of the subsystem, the interaction energy scales as the surface of the interface. The latter is, then, negligible in the limit, but it is not when defining energy densities or local energies. The picture is taken from [9]

Fig. 3.1 (taken from [9]) illustrates the intrinsic difficulty of partitioning the energy of a system into local contribution. The energy can be written as: $\mathcal{E}(\Omega_i) = E(\Omega_i) + \frac{1}{2}(1 \pm \lambda)W_{12}$, where $E(\Omega_i)$ are the energies of the two isolated subsystems, W_{12} their interaction energy, and λ an arbitrary constant. In the thermodynamic limit the interaction energy can be neglected and so the value of the λ constant is irrelevant. This is because, at the leading order, all $E(\Omega_i)$ are linear in the volume, while the interaction between subsystem is sub-linear. When it comes to defining energy densities (i.e. energies of infinitesimal portions of a system) or atomic energies, instead, the magnitude of the interaction between different subsystems is comparable to their energies, which become therefore intrinsically ill-defined.

The total energy, as any extensive quantity, is defined by the integral over the

(energy) density:

$$E = \int_\Omega e(\boldsymbol{r}) d\boldsymbol{r}. \tag{3.15}$$

Notice that, in the thermodynamic limit, we should consider as equivalent any two energy densities whose integral over the volume Ω differs by a quantity that scales as the volume boundary. For instance, two densities are equivalent if they differ by the divergence of a (bounded) vector field $\boldsymbol{b}(\boldsymbol{r}, t)$:

$$e'(\boldsymbol{r}) = e(\boldsymbol{r}) - \nabla \cdot \boldsymbol{b}(\boldsymbol{r}, t). \tag{3.16}$$

We can think of $e'(\boldsymbol{r})$ and $e(\boldsymbol{r})$ as different gauges of the same scalar field. Since the energy is conserved, an energy current density (and its related macroscopic flux) can be associated to any gauge of the energy density via the continuity equation, Eq. (2.2).

$$\boldsymbol{\jmath}'(\boldsymbol{r}) = \boldsymbol{\jmath}(\boldsymbol{r}) - \dot{\boldsymbol{b}}(\boldsymbol{r}, t) \tag{3.17}$$

$$\boldsymbol{J}' = \boldsymbol{J} - \dot{\boldsymbol{B}}(t) \tag{3.18}$$

where $\boldsymbol{B} = \frac{1}{\Omega}\int_\Omega \boldsymbol{b} d\boldsymbol{r}$. Thus, the macroscopic energy fluxes in two different energy gauges differ by the total time derivative of a bounded phase-space vector function. By inserting $\boldsymbol{J}'$ inside the Green Kubo equation one obtains:

$$\kappa' = \frac{\Omega}{k_B T^2} \int_0^\infty \left\langle \boldsymbol{J}'(t) \cdot \boldsymbol{J}'(0) \right\rangle_{eq} dt \tag{3.19}$$

$$= \kappa + \frac{\Omega}{k_B T^2} \left[\left\langle \boldsymbol{B}(t) \cdot \dot{\boldsymbol{B}}(0) \right\rangle_{eq} |_{-\infty}^{+\infty} + \left\langle \boldsymbol{B}(t) \cdot \dot{\boldsymbol{J}}(0) \right\rangle_{eq} |_{-\infty}^{+\infty} \right] \tag{3.20}$$

Since $\boldsymbol{B}$ is bounded and the equilibrium expectations of both a total time derivative and a current vanish, we conclude that $\kappa' = \kappa$.

3.4 Energy flux from a classical force field

In classical MD the total potential energy, E can be divided in local atomic contributions, $E = \sum_n w_n$. The energy density can be defined accordingly:

$$e(\boldsymbol{r}, \Gamma_t) = \sum_n \delta(\boldsymbol{r} - \boldsymbol{R}_n) e_n(\Gamma_t) \tag{3.21}$$

$$e_n(\Gamma_t) = \frac{\boldsymbol{P}_n^2}{2M_n} + w_n(\{\boldsymbol{R}\}) \tag{3.22}$$

where $\frac{\boldsymbol{P}_n^2}{2M_n}$ is the atomic kinetic energy.

In order to evaluate Eq. (3.14), we need the derivative of the energy density:

$$\frac{\partial e(\boldsymbol{r},\Gamma_t)}{\partial \boldsymbol{R}_n} = \delta(\boldsymbol{r}-\boldsymbol{R}_n)\frac{\partial e_n(\Gamma_t)}{\partial \boldsymbol{R}_n} + e_n(\Gamma_t)\frac{\partial}{\partial \boldsymbol{R}_n}\delta(\boldsymbol{r}-\boldsymbol{R}_n) \tag{3.23}$$

$$\frac{\partial e(\boldsymbol{r},\Gamma_t)}{\partial \boldsymbol{P}_n} = \boldsymbol{V}_n. \tag{3.24}$$

By inserting Eqs. (3.23) and (3.24) into Eq. (3.14), we get an expression for the energy flux:

$$\begin{aligned}\boldsymbol{J}^E &= \frac{1}{\Omega}\int_\Omega d\boldsymbol{r}\boldsymbol{r}\sum_n \Bigg[\boldsymbol{V}_n\delta(\boldsymbol{r}-\boldsymbol{R}_n)\frac{\partial e_n}{\partial \boldsymbol{R}_n}+ \\ &\qquad +\boldsymbol{V}_n e_n\frac{\partial}{\partial \boldsymbol{R}_n}\delta(\boldsymbol{r}-\boldsymbol{R}_n)+\boldsymbol{V}_n\cdot\boldsymbol{F}_n\delta(\boldsymbol{r}-\boldsymbol{R}_n)\Bigg] \qquad (3.25)\\ &= \frac{1}{\Omega}\left[\sum_n \boldsymbol{V}_n e_n + \sum_n \boldsymbol{R}_n\left(\boldsymbol{F}_n\cdot\boldsymbol{V}_n + \sum_m \boldsymbol{V}_m\cdot\frac{\partial w_n}{\partial \boldsymbol{R}_m}\right)\right] \qquad (3.26)\\ &= \sum_n\left[\boldsymbol{V}_n e_n + \sum_m(\boldsymbol{R}_m-\boldsymbol{R}_n)\frac{\partial w_m}{\partial \boldsymbol{R}_n}\cdot\boldsymbol{V}_n\right] \qquad (3.27)\end{aligned}$$

The Eq. (3.27) is the expression for the energy current with classical force field: the first depends on the atomic (potential and kinetic) energy $\boldsymbol{J}^{conv} = \sum_n \boldsymbol{V}_n e_n$ and it is often called *convective contribution*; the second, denoted *virial contribution*, depends on the derivative of the atomic energy $\boldsymbol{J}^{virial} = \sum_{nm}(\boldsymbol{R}_m-\boldsymbol{R}_n)\frac{\partial w_m}{\partial \boldsymbol{R}_n}\cdot\boldsymbol{V}_n$.

The matrix $W_n^{\alpha,\beta} = \sum_m(R_m^\alpha - R_n^\alpha)\frac{\partial w_m}{\partial R_n^\beta}$, often referred to as the atomic virial, is symmetric in the Cartesian indexes only in the case of two body potentials [59, 60]. In which case, the term $\frac{\partial w_m}{\partial \boldsymbol{R}_n}$ can be interpreted as the negative of the force on atom m due to atom n.

3.5 Multicomponent systems

A system made of Q atomic species (a *Q-component system*) has $Q+4$ conserved quantities (the number of atoms of each species, the energy, and the three components of the momentum). The energy and atomic-number currents are vector quantities, whereas the momentum currents are 3×3 (stress) tensors, which do not couple with the former in a rotationally invariant system. The total momentum is

not only a conserved quantity by itself, but is also a linear combination of the volume integral of the atomic-number currents (*atomic-number fluxes*). This reduces the number of independent mass fluxes from Q to $Q-1$. We conclude that, when dealing with an Q-component system, the conserved quantities relevant to heat transport are the total energy and the total particle numbers (or masses) of each one of the $Q-1$ independent atomic components. The *phenomenological equations* in Eq. (2.15) becomes:

$$\boldsymbol{J}^i = \sum_{j=0}^{Q-1} L^{ij} \boldsymbol{F}^j, \tag{3.28}$$

where the energy flux is identified as the zero-th term and the remaining $Q-1$ fluxes being any linearly independent combinations of the mass fluxes. Given $F^0 = -\frac{\nabla T}{T^2}$, the thermal conductivity is defined as the ratio between the energy flux and the opposite temperature gradient:

$$\kappa = \frac{J^0}{T^2 F^0}. \tag{3.29}$$

Dropping the Cartesian indices, from Eq. (3.28) J^0 is:

$$J^0 = \sum_{i=0}^{Q-1} L^{0i} F^i \tag{3.30}$$

$$\frac{J^0}{F^0} = L^{00} + \frac{1}{F^0} \sum_{i=1}^{Q-1} L^{0i} F^i. \tag{3.31}$$

By imposing that all particle currents vanish ($J^i = 0$ for $i > 0$), we get the component of the vector $[F^1, F^2, \cdots, F^{Q-1}]^T$ as solution of the remaining linear system:

$$F^i = -\sum_{j=1}^{Q-1} (L_{Q-1}^{-1})^{ij} L^{j0} F^0, \tag{3.32}$$

where L_{Q-1}^{-1} is the inverse of the $(Q-1)\times(Q-1)$ mass block of the Onsager matrix, which elements are all L^{ij} with $i, j \neq 0$. Combining Eq. (3.31) and Eq. (3.32) we can get the final expression for the thermal conductivity:

$$\kappa = \frac{1}{T^2} \left[L^{00} - \sum_{i,j=1}^{Q-1} L^{0i} (L_{Q-1}^{-1})^{ij} L^{j0} \right]. \tag{3.33}$$

The expression in square brackets in Eq. (3.33) is called the *Schur complement* of the mass block in the Onsager matrix, and is nothing but the inverse of the 00

element of the inverse Onsager matrix:

$$\kappa = \frac{1}{T^2 (L^{-1})^{00}}. \tag{3.34}$$

3.6 Convective invariance

From Eq. (3.33) a new invariance of the thermal transport coefficient arises, which we call *convective invariance*: *any linear combination of the mass fluxes can be added to the energy flux without affecting the thermal conductivity* [9, 10, 12]. If a new energy flux $(\boldsymbol{J}^0)'$ is defined as:

$$(\boldsymbol{J}^0)' = \boldsymbol{J}^0 + \sum_{i=1}^{Q-1} c_i \boldsymbol{J}^i, \tag{3.35}$$

where c_i are arbitrary real numbers, even though all the Onsager coefficients in Eq. (3.33) change, the resulting thermal transport coefficient is the same: $\kappa' = \kappa$.

This also implies that the heat conductivity is insensitive to a change of the zero of energy of each atomic species (crucial in the case of *ab initio* simulations), which would result in a change of the energy flux:

$$(\boldsymbol{J}^0)' = \boldsymbol{J}^0 + \sum_{i=1}^{Q-1} \frac{\delta\epsilon_i}{M_i} \boldsymbol{J}^i, \tag{3.36}$$

being $\delta\epsilon_i$ the change in energy of the species i and M_i the mass of the atoms of that species.

4

Data analysis

In this chapter we will illustrate the data analysis tool called *ceptral analysis* [11, 58]. We will consider the case of heat transport but the equations can be generalised to any transport coefficient which can be expressed as the Green-Kubo integral of an auto-correlation function of one flux.

4.1 Power spectrum and Wiener-Khintchine theorem

Equation (3.6) relates the thermal transport coefficient to the time auto-correlation function (tACF) of the energy current:

$$C(t) = \left\langle J^E\left(\Gamma_t\right) J^E\left(\Gamma_0\right)\right\rangle_{eq} \tag{4.1}$$

$$\kappa = \frac{\Omega}{k_B T^2}\int_0^\infty C(t)dt, \tag{4.2}$$

where we considered only one Cartesian component. Being the flux a stationary process we can apply the Wiener-Khintchine theorem [61, 62] to write:

$$S^E(\omega) \doteq \lim_{t\to\infty}\frac{1}{t}\left\langle\left|\int_0^t J^E(t')e^{i\omega t'}dt'\right|^2\right\rangle \tag{4.3}$$

$$= \int_{-\infty}^{+\infty} C(t)e^{i\omega t}dt. \tag{4.4}$$

Where we defined the *power spectrum*, $S(\omega)$, as the expectation of the squared modulus of the Fourier transform of the flux. By combining Eqs. (4.3) and (4.4) we can write the thermal transport coefficient as:

$$\kappa = \frac{\Omega}{2k_B T^2} S^E(\omega = 0). \tag{4.5}$$

In a general multi-component system, when several conserved currents interact with each other, one can define the *cross-spectrum* of the conserved fluxes as the Fourier transform of the cross-time correlation functions:

$$\bar{S}^{ij}(\omega) = \int_{-\infty}^{\infty} \langle J^i(\Gamma_t) \cdot J^j(\Gamma_0) \rangle e^{-i\omega t} dt \tag{4.6}$$

Then, even the thermal transport coefficient in Eq. (3.33) can be written as $\omega = 0$ value of a multi-component spectrum, $\kappa = S(\omega = 0)$ [9, 10, 12]:

$$S(\omega) = \frac{\Omega}{2k_B T^2} \frac{1}{[\bar{S}^{-1}(\omega)]^{00}} \tag{4.7}$$

where $[\bar{S}^{-1}(\omega)]^{00}$ is the 00 element of the inverse of the *cross spectrum* matrix defined in Eq. (4.6).

4.2 Cepstral analysis

The next sections will be dedicated to the cepstral analysis as presented in Refs. [9, 11, 12].

In every MD simulation, we have only access to a discrete sample of the flux (a time series), $\boldsymbol{J}_n^E = \boldsymbol{J}^E(n\epsilon)$, $0 \leq n \leq N-1$ where ϵ is the sampling period of the flux and N the length of the time series, assumed to be even for simplicity. Let us define the discrete Fourier transform of the flux time series as:

$$\hat{J}_k^E = \sum_{n=0}^{N-1} J_n^E e^{2\pi i \frac{kn}{N}}, \tag{4.8}$$

for $0 \neq k \neq N-1$. The *periodogram* is defined as:

$$\hat{S}_k^E = \frac{\epsilon}{N} \sum \left| \tilde{J}_k^E \right|^2. \tag{4.9}$$

For large N, $\langle \hat{S}_k^E \rangle = \hat{S}^E(\omega_k = 2\pi \frac{k}{N\epsilon})$, which means that the periodogram is an unbiased estimator of the power spectrum of the process at ω_k. Denoting $\mathcal{N}(\mu, \sigma^2)$ a normal distribution with mean value μ and variance σ^2, $\mathrm{Re}\left(\tilde{J}_k\right), \mathrm{Im}\left(\tilde{J}_k\right) \sim \mathcal{N}\left(0, \frac{N}{2\epsilon} S^E(\omega_k)\right)$. For simplicity, we neglect to treat differently $k = 0$ and $k = N/2$ committing an error of order $O(1/N)$, which vanishes in the long-time limit, more details in Refs [9, 11]. We conclude that in the large-N limit the sample spectrum of the energy flux time series is:

$$\hat{S}_k^E = S^E(\omega_k)\xi_k, \tag{4.10}$$

where the ξs are independent random variables distributed as a $\frac{1}{2}\chi^2_2$ [63].

In many applications we have access to multiple time series to estimate the power spectrum of a same process, $\{{}^pJ_n\}, p = 1, \dots l$. For example, in molecular dynamics a single trajectory provides one independent time series per Cartesian component of the flux, all equivalent in a isotropic system. It is convenient to define a mean sample spectrum:

$$ {}^l\hat{S}^E_k = \frac{\epsilon}{lN} \sum_{p=1}^{l} \left| {}^p\tilde{J}^E_k \right|^2, \tag{4.11} $$

$$ = S^E(\omega_k) {}^l\xi_k, \tag{4.12} $$

where $\xi_k \sim \frac{1}{2l}\chi^2_{2l}$. Eqs. (4.10) and (4.12) suggest that ${}^l\hat{S}^E_0$ is an unbiased estimator of the zero-frequency value of the true power spectrum. However, the variance of this estimator does not vanish in the large-N limit.

4.3 Log-spectrum and cepstral space

Fig. 4.1 displays the (estimator) of the power spectrum of water at ambient conditions, obtained from a 100 ps classical MD trajectory, showing its extremely noisy behaviour. The multiplicative nature of the statistical noise in Eq. (4.12) makes difficult to decouple the noise from the signal. Ercole et al. [11] suggest to pass to the logarithm space and apply a linear filter.

The logarithm of Eq. (4.10), called *log-periodogram* or log-spectrum, splits the signal and the noise in two addenda:

$$ {}^l\hat{L}_k = \log\left({}^l\hat{S}^E_k\right) \tag{4.13} $$

$$ = \log\left(S^E(\omega_k)\right) + \log\left({}^l\xi_k\right) \tag{4.14} $$

$$ = \log\left(S^E(\omega_k)\right) + {}^l\Lambda + {}^l\lambda_k \tag{4.15} $$

where we defined $\log\left({}^l\xi_k\right) = {}^l\Lambda + {}^l\lambda_k$, ${}^l\Lambda = \langle \log\left({}^l\xi_k\right)\rangle$ and ${}^l\lambda_k = \log\left({}^l\xi_k\right) - {}^l\Lambda$ are zero mean identically distributed independent stochastic variables with variance σ^2_l:

$$ {}^l\Lambda = \phi(l) - \log(l) \tag{4.16} $$

$$ \sigma^2_l = \phi'(l), \tag{4.17} $$

where $\phi(l)$ and $\phi'(l)$ are the digamma and trigamma functions [64], respectively.

Now a low-pass filter to Eq. (4.15) can be applied to reduce the noise without

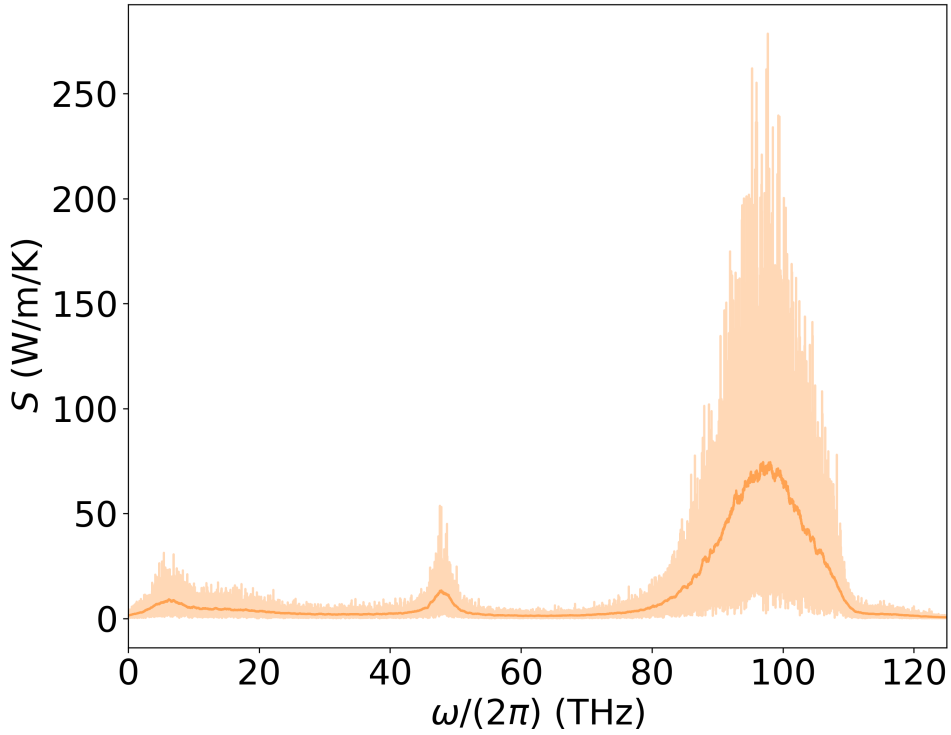


Figure 4.1: The (estimator of the) power spectrum of the heat current of liquid water computed with a NN potential trained over PBE data. The darker line is the window filtered spectrum, shown to better reproduce the overall behaviour.

affecting the real signal. Defining the *cepstrum* as the inverse Fourier transform of the log-spectrum [11, 65]:

$$ {}^{l}\hat{C}_n = \frac{1}{N}\sum_{k=0}^{N-1} {}^{l}\hat{L}_k e^{-2\pi i \frac{kn}{N}}. \tag{4.18} $$

Similarly to the sample power spectrum, the cepstral coefficients defined in Eq. (4.18) are real, periodic, and even: $\hat{C}_n = \hat{C}_{N-n}$. A generalized form of the central-limit theorem ensures that, in the large-N limit, the inverse Fourier transform of the ${}^{l}\hat{\lambda}_k$ in Eq. (4.15) are a set of independent (almost) identically distributed zero-mean normal deviates [66]. It follows that:

$$ {}^{l}\hat{C}_n = C_n + \delta_{n0}\,{}^{l}\Lambda + {}^{l}\hat{\mu}_n, \tag{4.19} $$

$$ C_n = \frac{1}{N}\sum_{k=0}^{N-1} \log\left(S^E(\omega_k)\right) e^{-2\pi i \frac{kn}{N}}, \tag{4.20} $$

where ${}^l\hat{\mu}_n$ are independent zero-mean normal deviates with variance:

$$\langle {}^l\hat{\mu}_n^2 \rangle = 2\frac{\sigma_l^2}{N} \tag{4.21}$$

If the log-periodogram of the true signal is smooth enough, the number of non-null C_n in Eq. (4.20), is much smaller than N, only few of them really contributes to the signal we want to compute, $S(\omega_k)$. One can, thus, discard the high-frequency part of ${}^l\hat{L}_k$, only keeping the first P^* coefficents, assuming that any C_n for $n > P^*$ do not contribute to the real signal but only to the power of the noise. Then one can retrieve the filtered log-periodogram from a discrete Fourier transform of Eq. (4.20):

$${}^l\hat{L}_k^* = {}^l\hat{C}_0 + 2\sum_{n=1}^{P^*-1} {}^l\hat{C}_n \cos\left(\frac{2\pi kn}{N}\right). \tag{4.22}$$

A simple example of the effects of the filter is shown in Fig. 4.2. The upper

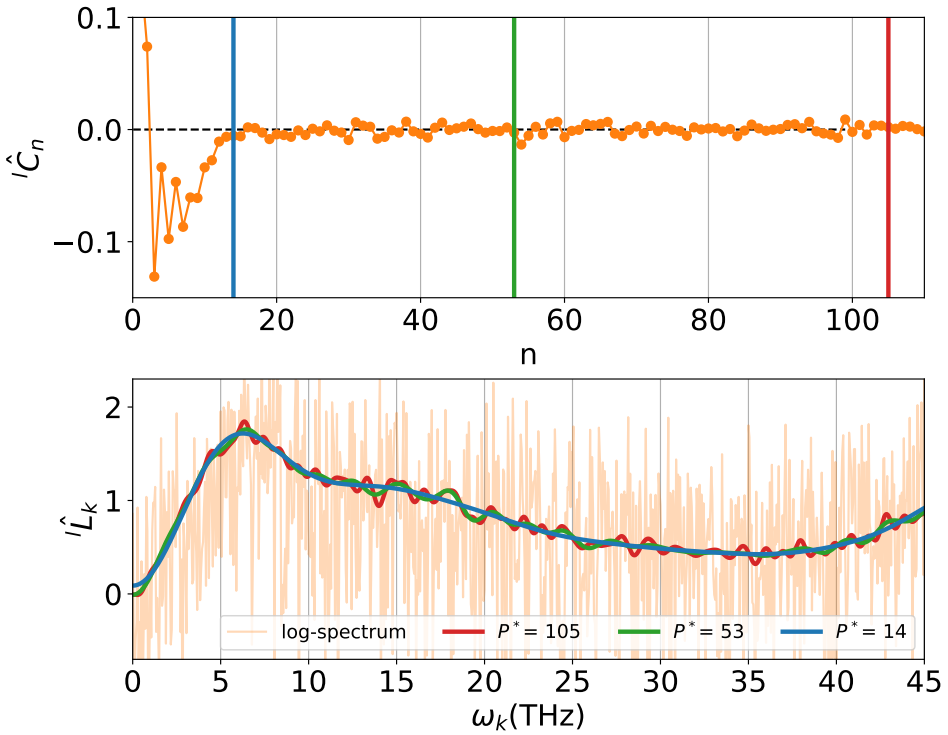


Figure 4.2: Upper panel shows the cepstral coefficients ${}^l\hat{C}_n$ as function of n. The vertical coloured lines correspond to the different values of P^* used for the filter in the lower panel. The lower panel shows the full log-spectrum (shade area) and three different filtered log-spectra for different values of P^*. The trajectory is the same as in Fig. 4.1.

panel displays the values of the cepstral coefficients ${}^{l}\hat{C}_n$ for $n \leq 110$, the lower panel exhibits, instead, the full log-spectrum (shade area) and three different filtered log-spectra for different values of P^*. Limiting the number of cepstral coefficients smooths the curves ${}^{l}\hat{L}^*_k$. It can be seen that only a few ${}^{l}\hat{C}_n$ (those below the blue vertical line) are not negligible, thus increasing P^* results only in including noise and not new information from the real signal. For instance, in the lower panel of Fig. 4.2 the red, green and blue lines converge to the same value at $\omega = 0$. Nonetheless, the lower P^*, the smoother is the filtered line.

4.4 Extracting the transport coefficient

The transport coefficient is related to the value at zero of Eq. (4.22):

$$
\begin{aligned}
{}^{l}\hat{L}^*_0 &= {}^{l}\hat{C}_0 + 2\sum_{n=1}^{P^*-1} {}^{l}\hat{C}_n, && (4.23)\\
&= \log\left(S^E(\omega=0)\right) + {}^{l}\Lambda + {}^{l}\hat{\mu}_0 + 2\sum_{n=0}^{P^*-1} {}^{l}\hat{\mu}_n. && (4.24)
\end{aligned}
$$

Since ${}^{l}\hat{L}^*_0$ is a Gaussian variable $\sim \mathcal{N}\left(\log\left(S(\omega=0)\right) + {}^{l}\Lambda, \phi'(l)\frac{4P^*-2}{N}\right)$, the transport coefficient and its statistical uncertainty can be extracted:

$$
\begin{aligned}
\kappa &= \frac{\Omega}{2k_BT}\exp\left(\langle {}^{l}\hat{L}^*_0\rangle - \phi(l) - \log(l)\right), && (4.25)\\
\sigma^2_\kappa &= \kappa^2\phi'(l)\frac{4P^*-2}{N}. && (4.26)
\end{aligned}
$$

Equation (4.26) shows that the statistical error on κ vanishes for $N \to +\infty$ and, for a fixed trajectory, it depends only on P^*. To select the optimal value of the latter the authors in Ref. [11] adopt the Akaike's Information Criterion [67]. The cepstral analysis can be extended to multi-component systems. Here we report only the final results, more details in Refs. [9, 12].

$$
\begin{aligned}
\kappa &= \frac{\Omega}{2k_BT}\exp\left(\langle {}^{l}\hat{L}^*_0\rangle - \phi(l-Q+1) - \log(l-Q+1)\right), && (4.27)\\
\sigma^2_\kappa &= \kappa^2\phi'(l-Q+1)\frac{4P^*-2}{N}. && (4.28)
\end{aligned}
$$

The cepstral analysis is fully implemented in the code `SporTran` [68].

4.5 Sampling and cut-off frequencies

For any discrete signal, such as the heat flux in a MD simulation, the Nyqvist frequency is the maximum frequency available for spectral/cepstral analysis [69]. It is determined by the sampling period ϵ as $f_{Ny} = \frac{1}{2\epsilon}$. Transport coefficients depend on the behaviour of the spectrum near zero, which is independent of ϵ, if the latter is small enough as to avoid aliasing effects. For this reason, L. Ercole et al [11] suggest to eliminate the high-frequency portion of the spectrum ($f > f^*$) by applying a low-pass filter to the time series (e.g a moving average as implemented in `SporTran`) and then resample the latter with a sampling period $\epsilon^* = \frac{1}{2f^*}$, thus resulting in a time series of $N^* = N\frac{f^*}{f_{Ny}}$ time steps.

In general, both the optimal number of cepstral coefficients and σ_κ^2 depend on the choice of the cutoff frequency. The smaller f^*, the smaller are the number of cepstral coefficients necessary to describe the log-spectrum to any given accuracy over the shorter frequency range [9, 11]. However, since $N^* < N$, the variance of κ is increased, following Eqs. (4.26) and (4.28).

4.6 Improving the data analysis with the multi-component analysis: the case of molecular fluids

Notwithstanding gauge and convective invariance, the statistical noise affecting the estimate of the heat conductivity does depend on the energy flux of the non-diffusing components that are added to the diffusive energy flux. Gauge invariance can then be leveraged to tune the optimal linear combination of non-diffusive fluxes to minimize the statistical error on the heat conductivity. In molecular fluids, all mass fluxes are non diffusive [8] and the energy is the only conserved quantity relevant to heat transport. Therefore in Eq. (4.7), we actually have that $S(0) = \frac{V}{2k_B T^2}\bar{S}^{00}(0)$ and, strictly speaking, no multi-component analysis would be needed. However data analysis is greatly facilitated when the power spectrum is as smooth as possible (to be precise, when the number of inverse Fourier coefficients of the logarithm of the spectrum are as few as possible [11]). For this reason, it may be convenient to complement the diffusive energy flux with a number of non-diffusive ones, which, while not altering the value of the spectrum in Eq. (4.7) at $\omega = 0$, decrease the total power, thus easing data analysis [9, 10, 12, 70]. This can be effectively done by treating the energy current as one component of an Q-component system, where all the other currents are non-diffusive ones [12]. This is illustrated in Fig. 4.3 where we report the power spectrum obtained via Eq. (4.4)

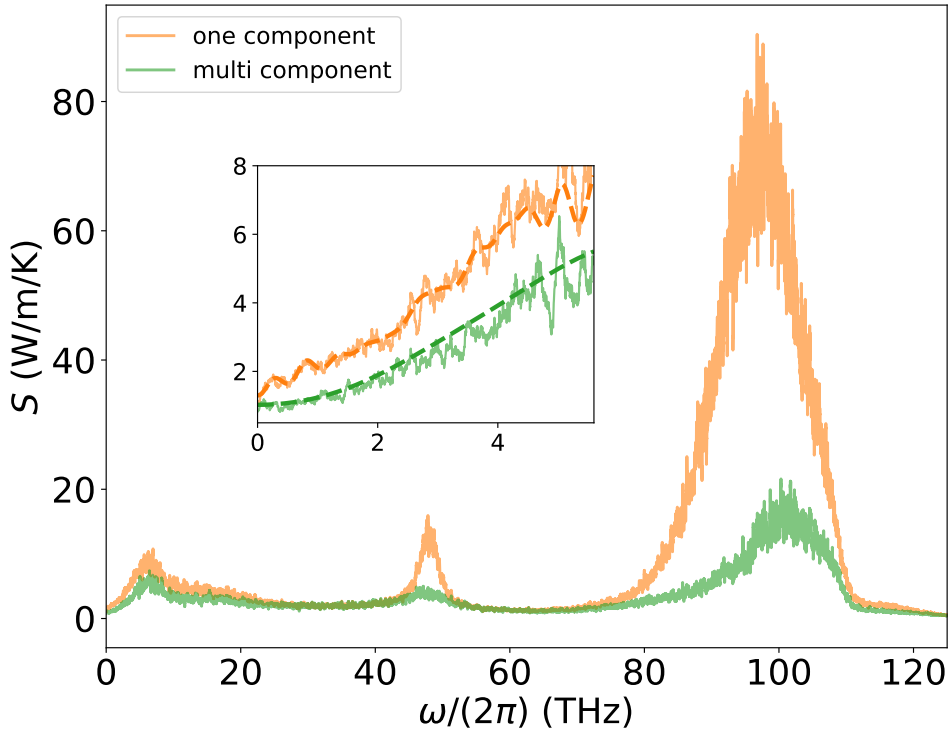


Figure 4.3: The power spectrum obtained via Eq. (4.4) (orange), and the multi-component spectrum in Eq. (4.7) (green). Both spectra are filtered with a window of 0.5 Thz. The inset shows the behaviour near $\omega = 0$ where the transport coefficients are extracted and the two spectra collapse to the same value. The green line has always a lower power. The dashed lines represent the cepstral-filtered spectra. The trajectory is the same as in Figs. 4.1 and 4.2.

(orange), and the multi-component spectrum in Eq. (4.7) (green). The latter is computed choosing $Q = 2$ and taking the momentum of the oxygen atoms as the auxiliary non-diffusive flux. This decorrelation decreases the power of the spectrum and flattens the spectrum near $\omega = 0$ facilitating data analysis by reducing the number of the required cepstral coefficients.

5

DFT energy current

A lot of progress has been made in recent years to develop *ab initio* approaches to heat transport based on electronic DFT. Some schemes used *ad hoc* ingredients, such as a (rather arbitrary) quantum-mechanical definition of the atomic energies [71]. Other schemes used a definition of the energy flux based on the normal-mode decomposition of the atomic coordinates and forces, which is only possible in crystalline solids [72]. Leveraging the *gauge invariance* [8, 58], Marcolongo, Umari, and Baroni (MUB) derived a general DFT expression for the adiabatic energy flux [8].The MUB expression of the heat current is implemented in `QEHeat` [43], an open-source code distributed with QUANTUM ESPRESSO project [73–75]. In this chapter we first illustrate the expression of the MUB adiabatic energy flux, then we give a detailed presentation of our implementation in `QEHeat`.

5.1 The MUB DFT adiabatic energy flux

In order to derive an expression for the adiabatic energy flux, we start from the standard expression for the DFT total energy in terms of Kohn-Sham (KS) eigenvalues ε_v, eigenfunctions $\phi_v(\boldsymbol{r})$, and density $n(\boldsymbol{r}) = \sum_v |\phi_v(\boldsymbol{r})|^2$ [76, 77]:

$$\begin{aligned} E^{DFT} =& \frac{1}{2}\sum_s M_s V_s^2 + \sum_s w_s^Z(\{\boldsymbol{R}\}) + \sum_v \varepsilon_v - \frac{e^2}{2}\int \frac{n(\boldsymbol{r})n(\boldsymbol{r}')}{|\boldsymbol{r}-\boldsymbol{r}'|} d\boldsymbol{r} d\boldsymbol{r}' \\ &+ \int d\boldsymbol{r}(\epsilon^{XC}[n](\boldsymbol{r}) - \mu^{XC}[n](\boldsymbol{r}))n(\boldsymbol{r}), \end{aligned} \tag{5.1}$$

where e is the electron charge, $\{\boldsymbol{R}\}$ and $\{\boldsymbol{r}\}$ indicate ionic and electronic positions, respectively, s represents an atom in the computational cell; w_s^Z is the atomic electrostatic energy; ϵ^{XC} is the local exchange-correlation (XC) energy per particle, defined by the relation:

$$E^{XC}[n] = \int \epsilon^{XC}[n](\boldsymbol{r})n(\boldsymbol{r})d\boldsymbol{r}, \tag{5.2}$$

E^{XC} being the total XC energy of the system, and μ^{XC} is the XC potential:

$$\mu^{XC}(\boldsymbol{r}) = \frac{\delta E^{XC}}{\delta n(\boldsymbol{r})} = \epsilon^{XC}(\boldsymbol{r}) + \int \frac{\delta \epsilon^{XC}(\boldsymbol{r}')}{\delta n(\boldsymbol{r})} n(\boldsymbol{r}') d\boldsymbol{r}', \tag{5.3}$$

The actual expression for ϵ^{XC} depend on the type of exchange correlation potential used. In `QEHeat` are implemented only the local density approximation (LDA) and the generalized gradient approximation (GGA). The DFT energy density can be defined from its integral [8, 78]:

$$E^{DFT} = \int e^{DFT}(\boldsymbol{r}) d\boldsymbol{r}, \tag{5.4}$$

$$e^{DFT}(\boldsymbol{r}) = e^{KS}(\boldsymbol{r}) + e^{0}(\boldsymbol{r}) + e^{H}(\boldsymbol{r}) + e^{XC}(\boldsymbol{r}) \tag{5.5}$$

where, denoting $\hat{H}^{KS}$ the instantaneous Kohn-Sham Hamiltonian:

$$e^{KS}(\boldsymbol{r}) = Re \sum_{v} \phi_v^*(\boldsymbol{r}) \hat{H}^{KS} \phi_v(\boldsymbol{r}) \tag{5.6}$$

$$e^{0}(\boldsymbol{r}) = \sum_{s} \delta(\boldsymbol{r} - \boldsymbol{R}_s) \left(\frac{1}{2} M_s V_s^2 + w_s^Z \right) \tag{5.7}$$

$$e^{H}(\boldsymbol{r}) = -\frac{1}{2} n(\boldsymbol{r}) v^H(\boldsymbol{r}) \tag{5.8}$$

$$e^{XC}(\boldsymbol{r}) = (\epsilon^{XC}(\boldsymbol{r}) - \mu^{XC}(\boldsymbol{r})) n(\boldsymbol{r}). \tag{5.9}$$

The expression for the DFT adiabatic energy flux, $\boldsymbol{J}^{MUB}$, can derived inserting Eq. (5.5) into Eq. (3.13) [8]:

$$\boldsymbol{J}^{MUB} = \boldsymbol{J}^{KS} + \boldsymbol{J}^{0} + \boldsymbol{J}^{n} + \boldsymbol{J}^{H} + \boldsymbol{J}^{XC}, \tag{5.10}$$

For a more extensive and detailed study of the implementations of the many components of the MUB current the reader is referred to Section 5.2.1.

$$\boldsymbol{J}^{KS} = \sum_{v} \left(\langle \phi_v | \hat{\boldsymbol{r}} \hat{H}^{KS} | \dot{\phi}_v \rangle + \varepsilon_v \langle \dot{\phi}_v | \hat{\boldsymbol{r}} | \phi_v \rangle \right), \tag{5.11}$$

$$\boldsymbol{J}^{0} = \sum_{s\boldsymbol{L}} \sum_{v} \left\langle \phi_v \left| (\hat{\boldsymbol{r}} - \boldsymbol{R}_s - \boldsymbol{L}) \left(\boldsymbol{V}_s \cdot \nabla_{s\boldsymbol{L}} \hat{v}^0 \right) \right| \phi_v \right\rangle, \tag{5.12}$$

$$\boldsymbol{J}^{n} = \sum_{s} \left[\boldsymbol{V}_s e_s^0 + \sum_{t \neq s} \sum_{L} (\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}) \left(\boldsymbol{V}_t \cdot \nabla_{t\boldsymbol{L}} w_s^Z \right) - \sum_{L \neq 0} \boldsymbol{L} \left(\boldsymbol{V}_s \cdot \nabla_{s\boldsymbol{L}} w_s^Z \right) \right] \tag{5.13}$$

$$\boldsymbol{J}^{H} = \frac{1}{4\pi e^2} \int \dot{v}^H(\boldsymbol{r}) \nabla v^H(\boldsymbol{r}) d\boldsymbol{r}, \tag{5.14}$$

$$\boldsymbol{J}^{XC} = \begin{cases} 0 & \text{(LDA)} \\ -\displaystyle\int n(\boldsymbol{r}) \dot{n}(\boldsymbol{r}) \boldsymbol{\partial} \epsilon^{GGA}(\boldsymbol{r}) d\boldsymbol{r} & \text{(GGA).} \end{cases} \tag{5.15}$$

In the following, these components are referred to as the *Kohn-Sham*, *Zero*, *Ionic*, *Hartree* and *Exchange-Correlation* fluxes, respectively. For the insulating systems of interest in this work, the ionic degrees of freedom completely define the state of the system and the electrons populate the ground state, according to the adiabatic approximation. Each time derivative, indicated with the usual *dot* operator, has then to be understood from the implicit dependence on the atomic positions.

Here and in the following, we indicate with $\boldsymbol{L}$ the lattice vector. We stress that periodic boundary conditions (PBC) are assumed here over the unit (simulation) cell. This implies that Kohn-Sham orbitals and energies are sampled at the Γ point of the Brillouin cell. The notation $\nabla_{s\boldsymbol{L}}$ is a shorthand for the gradient with respect to displacement of the atom at location $\boldsymbol{R}_s + \boldsymbol{L}$. A summation over s runs over all atoms belonging to the simulation cell. Unless otherwise specified, carets indicate quantum-mechanical operators, as in $\hat{H}^{KS}$ or $\hat{\boldsymbol{r}}$. The symbol ϵ^{GGA} stands for the generalized gradient approximation (GGA) exchange-correlation local energy per particle and its derivative with respect to density gradients is indicated with $\boldsymbol{\partial}\epsilon^{GGA}$, which is a vector whose component along direction $i \in \{x, y, z\}$ is given by $\partial \epsilon^{GGA}(n, \nabla n)/\partial(\nabla_i n)$. In the present version of `QEHeat` only the local density approximation (LDA) and the generalized gradient approximation given by the PBE [44] functional are implemented.

The formulas reported are compatible with PBC. Thus, they can be implemented for periodic systems, where some of the summations need to be extended to all the periodic replicas of the atoms. Nevertheless, the computation of the various contributions to the energy current, Eqs. (5.11-5.15), is plagued by the occurrence of several divergences, arising from the long range character of the Coulomb interaction. As it is the case for the total energies, atomic forces, and

stress, the individual electronic, ionic, and electron-ion contributions diverge and it is only their sum that is regular in the thermodynamic limit. In order to regularize the individual components of the MUB flux, we compute all the relevant terms by screening the Coulomb interaction with a Yukawa cutoff, $\frac{1}{x} \to \frac{e^{-\mu x}}{x}$. In Section 5.2.1, we check explicitly that the singular contributions to the various terms cancel each other in the $\mu \to 0$ limit, so that they can be consistently and safely neglected.

5.2 QEHeat: a DFT energy flux calculator

`QEHeat` is an open-source code that implements the computation of the MUB current, distributed within the QUANTUM ESPRESSO project. `QEHeat` can be easily interfaced to read a dynamical trajectory generated with a code of choice and compute the MUB flux for the corresponding steps. It is already delivered with an interface for the `cp.x` program of the QUANTUM ESPRESSO distribution, which is routinely used to perform Car-Parrinello molecular dynamics simulations [73–75]. The combination of `QEHeat`, `cp.x` and the post-processing tool `SPORTRAN` [68], which implements the *cepstral analysis*, provides a convenient framework to compute the heat conductivity of extended insulating systems-be they crystalline, amorphous, or liquid-entirely from first principles.

`QEHeat` computes the MUB energy flux as a function of the atomic positions, $\{\boldsymbol{R}_s\}$ and velocities, $\{\boldsymbol{V}_s\}$, *i.e.* for any selected snapshot of an *ab initio* molecular dynamics (AIMD) trajectory. Despite the complexity of the resulting formula for the energy current, from a practical point of view the use of `QEHeat` relies on a limited number of additional input parameters with respect to a standard QUANTUM ESPRESSO DFT computation. These are reported in the `energy_current` input namelist, which is shown in figure 5.1. The meaning of all the keywords is explained in more detail in section Section 5.2.3. The only additional parameters are `eta` and `n_max`, controlling the Ewald summations, which appear only in classical contributions to the energy current, and `delta_t`, a time-discretization parameter, used to perform numerical derivatives. The default values should work for most systems.

5.2.1 Numerical implementation of each current component

Kohn-Sham current

Starting from Eq. (5.11) and after some simple algebra [8] we end up with the following expression for the Kohn-Sham current. For every Cartesian component,

i, one has:

$$J_i^{KS} = \mathcal{R} \sum_v \langle \bar{\phi}_{v\ i}^{\mathrm{c}}(t) \left| \hat{H}^{KS} + \epsilon_v \right| \dot{\phi}_v^{\mathrm{c}}(t) \rangle, \tag{5.16}$$

where

$$|\bar{\phi}_{v\ i}^{\mathrm{c}}\rangle \equiv \hat{P}^{\mathrm{c}} \hat{r}_{\ i} |\phi_v\rangle , \tag{5.17}$$

$$|\dot{\phi}_v^{\mathrm{c}}\rangle, \equiv \hat{P}^{\mathrm{c}} \left| \dot{\phi}_v \right\rangle , \tag{5.18}$$

$$\hat{P}^{\mathrm{c}} \equiv 1 - \hat{P}^{\mathrm{v}} \tag{5.19}$$

$$\hat{P}^{\mathrm{v}} \equiv \sum_v |\phi_v\rangle \langle \phi_v| , \tag{5.20}$$

and $\hat{P}^{\mathrm{v}}$ and $\hat{P}^{\mathrm{c}}$ are the projectors over the occupied- (valence-) and empty- (conduction-) state manifolds respectively. $\bar{\phi}_v^{\mathrm{c}}$ is calculated by solving the linear system:

$$(\hat{H}^{KS} - \epsilon_v + \alpha \hat{P}^{\mathrm{v}}) \left| \bar{\phi}_{v\ i}^{\mathrm{c}} \right\rangle = \hat{P}^{\mathrm{c}} [\hat{H}^{KS}, \hat{r}_{\ i}] |\phi_v\rangle , \tag{5.21}$$

where $[\cdot, \cdot]$ indicates the commutator between quantum mechanical operators and α is a positive constant that removes the singularity of the linear system and forces the solution to be orthogonal to the valence manyfold, using standard techniques from density-functional perturbation theory (DFPT) [79]. In order to avoid alignment problems between wave-functions at different time steps, $\dot{\phi}_v^{\mathrm{c}}$ is calculated by moving the derivative to the projector [8, 74], using the relation:

$$\begin{aligned} \left| \dot{\phi}_v^{\mathrm{c}} \right\rangle &\equiv \hat{P}^{\mathrm{c}} \left| \dot{\phi}_v \right\rangle \\ &= \hat{P}^{\mathrm{c}} \dot{\hat{P}}^{\mathrm{v}} |\phi_v\rangle , \end{aligned} \tag{5.22}$$

which can be evaluated using a finite-difference scheme, as explained in detail in Sec. 5.2.2, reading:

$$\begin{aligned} \left| \dot{\phi}_v^{\mathrm{c}} \right\rangle &= \hat{P}^{\mathrm{c}} \dot{\hat{P}}^{\mathrm{v}} |\phi_v\rangle \\ &\sim \frac{1}{dt} (1 - \hat{P}^{\mathrm{v}}(t)) (\hat{P}^{\mathrm{v}}(t + dt/2) - \hat{P}^{\mathrm{v}}(t - dt/2)) |\phi_v(t)\rangle , \end{aligned} \tag{5.23}$$

where dt is a time-discretization parameter, input of a `QEHeat` computation, which is discussed in more detail in Section 5.2.2 and 5.2.6. After evaluating Eq. (5.23) and solving Eq. (5.21), the results can be inserted into (5.16). We note that the computational cost to evaluate all the components of the MUB current is dominated by the solution of the linear system, Eq. (5.21).

Zero current

The *Zero* current $\boldsymbol{J}^0$, Eq. (5.12), can be better treated by separating the local and non-local contributions from the ionic pseudo-potential, $\hat{v}_0$. Each contribution acts on a generic wave-function ϕ in the following way:

$$\hat{v}^0 = \sum_{s\boldsymbol{L}} \hat{v}^{LOC}_{s\boldsymbol{L}} + \hat{v}^{NL}_{s\boldsymbol{L}}, \tag{5.24}$$

$$\langle \boldsymbol{r}|\hat{v}^{LOC}_{s\boldsymbol{L}}|\phi\rangle = f^{LOC}_s\left(|\boldsymbol{r}-\boldsymbol{R}_s-\boldsymbol{L}|\right)\phi(\boldsymbol{r}), \tag{5.25}$$

$$\hat{v}^{NL}_{s\boldsymbol{L}}|\phi\rangle = \sum_{lm} D^s_l|\beta^{s\boldsymbol{L}}_{lm}\rangle\langle\beta^{s\boldsymbol{L}}_{lm}|\phi\rangle, \tag{5.26}$$

where the total pseudopotential has been separated into atomic contributions and into its local (LOC), long-tailed, and non-local (NL), short-range, components. The subscripts in the expressions $\hat{v}^{LOC}_{s\boldsymbol{L}}$ and $\hat{v}^{NL}_{s\boldsymbol{L}}$ indicate that the corresponding ionic pseudo-potential is centered at the atomic position $\boldsymbol{R}_s + \boldsymbol{L}$. Note that the atomic contributions $\hat{v}^{LOC/NL}_{s\boldsymbol{L}}$, when differentiated with respect to ionic positions, only depend on the position of the atom located at $\boldsymbol{R}_s + \boldsymbol{L}$.

The local and nonlocal potentials provide two contributions to the Zero current, which we discuss individually in the next sections. For the local contribution, $f^{LOC}_s(r)$ is the radial local pseudo-potential provided in the pseudo-potential datasets for each species. The centered beta functions $\beta^s_{lm}(\boldsymbol{r})$ (also called projectors) define the non-local component of the pseudo-potentials and are defined as $\beta^s_{lm}(\boldsymbol{r}) \equiv \beta^s_l(r)Y_{lm}(\hat{\boldsymbol{r}})$, where Y_{lm} are the real spherical harmonics with quantum numbers l, m and $\hat{\boldsymbol{r}}$ is the unit versor, not to be confused with the multiplicative position operator. We denote the radial components of the beta functions with the similar notation $\beta^s_l(r)$. These are the ones provided for each atomic species in the pseudo-potential datasets, alongside the constant D matrix. At every time step, the beta functions need to be centered on the instantaneous ionic positions and in Eq. (5.26) we indicated the translated beta function for atom at position $\boldsymbol{R}_s - \boldsymbol{L}$ with an apex. More explicitly :

$$\langle\boldsymbol{r}|\beta^{s\boldsymbol{L}}_{lm}\rangle \equiv \beta^s_{lm}(\boldsymbol{r}-\boldsymbol{R}_s-\boldsymbol{L}). \tag{5.27}$$

We fix some handy notation and define from a (real) localized function $\gamma(\boldsymbol{r})$ (*e.g.* a β function or the local pseudo-potential) its periodic counterpart as:

$$\overline{\gamma}(\boldsymbol{r}) \equiv \sum_{\boldsymbol{L}} \gamma(\boldsymbol{r}-\boldsymbol{L}), \tag{5.28}$$

whose Fourier components can be computed as:

$$\tilde{\overline{\gamma}}(\boldsymbol{G}) = \mathcal{F}[\gamma](\boldsymbol{G}) \equiv \frac{1}{\Omega}\int_{\mathbb{R}^3}\gamma(\boldsymbol{r})e^{-i\boldsymbol{G}\cdot\boldsymbol{r}}d\boldsymbol{r}, \tag{5.29}$$

where we introduced the symbol $\mathcal{F}[\gamma](\boldsymbol{G})$ to identify a standard Fourier transform over all the three dimensional space, here defined for localized functions. In the following we use the notation $\overline{\gamma}^{\boldsymbol{a}}$ to indicate the translated and periodic function built from its localized counterpart, analogously to the notation introduced for the projector. Note that the following relation is used in the code, $\tilde{\overline{\gamma}}^{\boldsymbol{a}}(\boldsymbol{G}) = e^{-i\boldsymbol{G}\cdot\boldsymbol{a}}\mathcal{F}[\gamma](\boldsymbol{G})$, thanks to standard properties of the Fourier transform. The remaining Fourier transforms are evaluated numerically by `QEHeat` whenever needed, as explained below.

Zero current: local contribution

The local pseudo-potential of the s-atom $f_s^{LOC}(r)$ behaves as $\sim -Z_s e^2/r$ for large r. We call this local long-range contribution to the flux $\boldsymbol{J}_0^{LR}$:

$$\boldsymbol{J}_0^{LR} = \sum_v \left\langle \phi_v \middle| \sum_{s\boldsymbol{L}} (\hat{\boldsymbol{r}} - \boldsymbol{R}_s - \boldsymbol{L}) \left(\boldsymbol{V}_s \cdot \nabla_{s\boldsymbol{L}} f_s^{LOC}(|\hat{\boldsymbol{r}} - \boldsymbol{R}_s - \boldsymbol{L}|) \right) \middle| \phi_v \right\rangle . \tag{5.30}$$

We use the chain rule and the definitions:

$$h_{ij}^s(\boldsymbol{r}) \equiv \sum_{\boldsymbol{L}} \frac{(\boldsymbol{r} - \boldsymbol{L})_i (\boldsymbol{r} - \boldsymbol{L})_j}{|\boldsymbol{r} - \boldsymbol{L}|} f_s^{\prime LOC}(|\boldsymbol{r} - \boldsymbol{L}|) \tag{5.31}$$

$$u_i(\boldsymbol{r}) \equiv -\sum_s \sum_{j\in\{x,y,z\}} V_{sj} h_{ij}^s(\boldsymbol{r} - \boldsymbol{R}_s), \tag{5.32}$$

where $f_s^{\prime LOC}(r)$ is the derivative of the local pseudo-potential. Note that both h and u are periodic functions. The current can then be rewritten as:

$$\begin{aligned} J_{0,i}^{LR} &= \int_\Omega n(\boldsymbol{r}) u_i(\boldsymbol{r}) d\boldsymbol{r} \\ &= \Omega \sum_{\boldsymbol{G}} \tilde{n}(\boldsymbol{G}) \tilde{u}_i(-\boldsymbol{G}), \end{aligned} \tag{5.33}$$

where:

$$\tilde{u}_i(\boldsymbol{G}) = -\sum_s \sum_{j\in\{x,y,z\}} V_{sj} \tilde{h}_{ij}^s(\boldsymbol{G}) e^{-i\boldsymbol{G}\cdot\boldsymbol{R}_s}. \tag{5.34}$$

The reciprocal Fourier components of $h_{ij}^s(\boldsymbol{r})$ are computed through the following procedure, which avoids an explicit numerical differentiation of the pseudo-potential. One writes:

$$\begin{aligned} h_{ij}^s(\boldsymbol{r}) &= \partial_j \left[\sum_{\boldsymbol{L}} (\boldsymbol{r} - \boldsymbol{L})_i f_s^{LOC}(|\boldsymbol{r} - \boldsymbol{L}|) \right] - \delta_{ij} \sum_{\boldsymbol{L}} f_s^{LOC}(|\boldsymbol{r} - \boldsymbol{L}|) \\ &\equiv \sum_{\boldsymbol{G}} \left(\tilde{h}_{ij}^{1,s}(\boldsymbol{G}) + \tilde{h}_{ij}^{2,s}(\boldsymbol{G}) \right) e^{i\boldsymbol{G}\cdot\boldsymbol{r}}. \end{aligned} \tag{5.35}$$

The resulting expressions of $\tilde{h}^1(\boldsymbol{G})$ and $\tilde{h}^2(\boldsymbol{G})$ can be evaluted using Eq. (5.29) and the standard expansion of $e^{-i\boldsymbol{G}\cdot\boldsymbol{r}}$ into spherical harmonics and Bessel functions J_l:

$$\tilde{h}^{1,s}_{ij}(\boldsymbol{G}) = \begin{cases} 0 & \text{for } \boldsymbol{G} = \boldsymbol{0} \\ \dfrac{4\pi}{\Omega}\dfrac{G_i G_j}{G^2} G \displaystyle\int_0^\infty r^3 f_s^{LOC}(r) J_1(Gr) dr & \text{for } \boldsymbol{G} \neq \boldsymbol{0}, \end{cases} \tag{5.36}$$

where the $\boldsymbol{G} = 0$ component vanishes thanks to the presence of a derivative in the definition of h^1. For h^2 the situation is different:

$$\tilde{h}^{2,s}_{ij}(\boldsymbol{G}) = \begin{cases} -\delta_{ij}\dfrac{4\pi}{\Omega}\displaystyle\int_0^\infty r^2 f_s^{LOC}(r) dr & \text{for } \boldsymbol{G} = \boldsymbol{0} \\ -\delta_{ij}\dfrac{4\pi}{\Omega}\displaystyle\int_0^\infty r^2 f_s^{LOC}(r) J_0(Gr) dr & \text{for } \boldsymbol{G} \neq \boldsymbol{0} \end{cases} \tag{5.37}$$

These expressions need still to be a bit elaborated before being evaluated by `QEHeat`. In order to evaluate integrals of the localized functions, one needs to add and subtract the asymptotic long-range tail of the local pseudo-potential. The long-range part can be integrated analytically for any finite value of the Yukawa screening parameter, μ, after plugging in the exact form of the spherical Bessel functions. For h^2, this permits to extract the divergent part in the $\boldsymbol{G} = \boldsymbol{0}$ contribution as well. The final results read:

$$\tilde{h}^{1,s}_{ij}(\boldsymbol{G}) = \begin{cases} 0 & \text{for } \boldsymbol{G} = \boldsymbol{0} \\ \dfrac{4\pi}{\Omega}\dfrac{G_i G_j}{G^2} G \left[\displaystyle\int_0^\infty r^3 \left(f_s^{LOC}(r) + \frac{e^2 Z_s}{r}\right) J_1(Gr) dr - \frac{2e^2 Z_s}{G^3}\right] & \text{for } \boldsymbol{G} \neq \boldsymbol{0} \end{cases} \tag{5.38}$$

and

$$\tilde{h}^{2,s}_{ij}(\boldsymbol{G}) = \begin{cases} -\delta_{ij}\dfrac{4\pi}{\Omega}\left[\displaystyle\int_0^\infty r^2 \left(f_s^{LOC}(r) + \frac{e^2 Z_s}{r}\right) dr - \frac{e^2 Z_s}{\mu^2}\right] & \text{for } \boldsymbol{G} = \boldsymbol{0} \\ -\delta_{ij}\dfrac{4\pi}{\Omega}\left[\displaystyle\int_0^\infty r^2 \left(f_s^{LOC}(r) + \frac{e^2 Z_s}{r}\right) J_0(Gr) dr - \frac{e^2 Z_s}{G^2}\right] & \text{for } \boldsymbol{G} \neq \boldsymbol{0} \end{cases} \tag{5.39}$$

We note here again that only from h^2 we get a Coulombian divergence when $\mu \to 0$. Overall, the divergent part of the Zero current is equal to

$$\boldsymbol{J}^0_{div} = -Z_{tot} e^2 \frac{4\pi}{\mu^2 \Omega} \sum_s \boldsymbol{V}_s Z_s, \tag{5.40}$$

where $Z_{tot} = \sum_s Z_s$.

Zero current: nonlocal contribution

The non-local part is inherently short range and we call it $\boldsymbol{J}_0^{SR}$:

$$\boldsymbol{J}_0^{SR} = \sum_v \left\langle \phi_v \left| \sum_{s\boldsymbol{L}} (\hat{\boldsymbol{r}} - \boldsymbol{R}_s - \boldsymbol{L}) \left(\boldsymbol{V}_s \cdot \nabla_{s\boldsymbol{L}} \hat{v}_{s\boldsymbol{L}}^{NL} \right) \right| \phi_v \right\rangle, \tag{5.41}$$

For a pair of two localized functions $(\gamma_1(\boldsymbol{r}), \gamma_2(\boldsymbol{r}))$ we introduce the notation, given a generic translation $\boldsymbol{a}$:

$$\mathcal{A}[\gamma_1, \gamma_2](\boldsymbol{a}) \equiv \sum_v \langle \overline{\gamma}_1^{\boldsymbol{a}} | \phi_v \rangle \langle \phi_v | \overline{\gamma}_2^{\boldsymbol{a}} \rangle = \langle \overline{\gamma}_1^{\boldsymbol{a}} | \hat{P}_v | \overline{\gamma}_2^{\boldsymbol{a}} \rangle, \tag{5.42}$$

where we recall that $\overline{\gamma}_1^{\boldsymbol{a}}$ and $\overline{\gamma}_2^{\boldsymbol{a}}$ are the translated and periodic counterpart of localized functions. We note that, once the Fourier components of ϕ, $\overline{\gamma}_1$ and $\overline{\gamma}_2$ are known, evaluating $\mathcal{A}[\gamma_1, \gamma_2](\boldsymbol{a})$ involves just scalar products between periodic functions and is straightforward. Using these expressions and expanding the projector operator in their integral form, the current can be rewritten as:

$$J_{0,i}^{SR} = \sum_s \sum_{lm} \sum_{j \in \{x,y,z\}} V_{sj} D_l^s \left(\mathcal{A}[-r_i \partial_j \beta_{lm}^s, \beta_{lm}^s](\boldsymbol{R}_s) + \mathcal{A}[r_i \beta_{lm}^s, -\partial_j \beta_{lm}^s](\boldsymbol{R}_s) \right). \tag{5.43}$$

In order to evaluate Eq. (5.43) via (5.42) the Fourier transforms of the following four localized functions are needed : $\beta_{lm}(\boldsymbol{r}), r_i \beta_{lm}(\boldsymbol{r}), -\partial_i \beta_{lm}(\boldsymbol{r}), -r_i \partial_j \beta_{lm}(\boldsymbol{r})$, where we dropped the atomic index s. We need only to evaluate the first two expressions thanks to the identities:

$$\mathcal{F}\left[-\partial_j \gamma\right](\boldsymbol{G}) = -iG_j \mathcal{F}\left[\gamma\right](\boldsymbol{G}) \tag{5.44}$$

$$\mathcal{F}\left[-r_i \partial_j \gamma\right](\boldsymbol{G}) = -iG_j \mathcal{F}\left[r_i \gamma\right](\boldsymbol{G}) + \delta_{i,j} \mathcal{F}\left[\gamma\right](\boldsymbol{G}), \tag{5.45}$$

which are valid for every localized function $\gamma(\boldsymbol{r})$. We show how to evaluate the Fourier transform of the second function, $r_i \beta_{lm}(\boldsymbol{r})$, for $i = x$, which is the most complex one. The procedure is similar for the remaining expressions. One starts by replacing the factor x using that $Y_{11}(\hat{\boldsymbol{r}}) = -\sqrt{\frac{3}{4\pi}} \frac{x}{r}$ (according to the convention followed by QUANTUM ESPRESSO for the sign of the spherical harmonics). By expanding $e^{-i\boldsymbol{G}\cdot\boldsymbol{r}}$ as well into spherical harmonics, one gets:

$$\mathcal{F}[x\beta_{lm}](\boldsymbol{G}) = -\frac{4\pi}{\Omega} \sqrt{\frac{4\pi}{3}} \sum_{l'm'} Y_{l'm'}\left(\hat{\boldsymbol{G}}\right) \times \left((-i)^{l'} \int_0^\infty r^3 \beta_l(r) J_{l'}(Gr) dr \right) \left(\int d\hat{\boldsymbol{r}}\ Y_{l'm'}(\hat{\boldsymbol{r}}) Y_{lm}(\hat{\boldsymbol{r}}) Y_{11}(\hat{\boldsymbol{r}}) \right), \tag{5.46}$$

where $d\hat{\boldsymbol{r}}$ indicates an integral over the solid angle, such that $d\boldsymbol{r} = r^2 d\hat{\boldsymbol{r}} dr$. In QUANTUM ESPRESSO the Clebsch-Gordan coefficients are stored in a array `ap` defined as:

$$Y_{lm} Y_{l'm'} = \sum_{LM} \mathtt{ap}\left(L, M, l, m, l', m'\right) Y_{LM} \tag{5.47}$$

Application of this relation to the product $Y_{lm} Y_{11}$ allows us to simplify Eq. (5.46) to the final form:

$$\begin{aligned} \mathcal{F}[x\beta_{lm}](\boldsymbol{G}) = -\frac{4\pi}{\Omega}\sqrt{\frac{4\pi}{3}} \sum_{LM} Y_{LM}\left(\hat{\boldsymbol{G}}\right) \times \\ \left((-i)^L \int_0^\infty r^3 \beta_l(r) J_L(Gr) dr\right) \mathtt{ap}\left(L, M, l, m, 1, 1\right). \end{aligned} \tag{5.48}$$

The remaining radial integral is performed numerically on a grid.

Ionic current

The so-called Ionic flux, $\boldsymbol{J}^n$, is the contribution to the total energy flux that depends only on the ionic positions, $\boldsymbol{R}_s$, and velocities, $\boldsymbol{V}_s$. First of all let's take, as reported in A.1, the definition of the ionic energy $e_s^0 = \frac{1}{2} M_s V_s^2 + w_s^Z$ and the electrostatic energy:

$$w_s^Z = \frac{e^2}{2} \sum_{t \neq s} \sum_{\boldsymbol{L}} \frac{Z_t Z_s}{|\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}|} + \frac{1}{2} e^2 Z_s^2 \sum_{L \neq 0} \frac{1}{L}, \tag{5.49}$$

where $L = |\boldsymbol{L}|$. We can separate the expression of the Ionic flux, Eq. (5.13), in two terms: one depending only on the mass and velocity of the ions, an other depending on w_s^Z and its gradient.

$$\nabla_{t\boldsymbol{L}} w_s^Z = -\frac{e^2}{2} Z_s Z_t \frac{\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}}{|\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}|} f'(|\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}|), \tag{5.50}$$

where $\nabla_{t\boldsymbol{L}}$ is the shorthand notation for the gradient with respect to displacement of the atom with position $\boldsymbol{R}_t + \boldsymbol{L}$, introduced in Section 5.1. We, also, introduced $f(x) = \frac{1}{x}$ to keep track of the Coulombian contributions when applying the Yukawa screening.

For the sake of simplicity, let's introduce the following four quantities:

$$\overline{S}^A_{ij} = \sum_{\boldsymbol{L}\neq\boldsymbol{0}} \frac{L_i L_j}{|\boldsymbol{L}|} f'(L), \tag{5.51}$$

$$S^B = \sum_{\boldsymbol{L}\neq\boldsymbol{0}} f(L), \tag{5.52}$$

$$S^C(\boldsymbol{R}_s - \boldsymbol{R}_t) = \sum_{\boldsymbol{L}} f(|\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}|), \tag{5.53}$$

$$\overline{S}^D_{ij}(\boldsymbol{R}_s - \boldsymbol{R}_t) = \sum_{\boldsymbol{L}} \frac{(\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L})_i (\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L})_j}{|\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}|} f'(|\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}|), \tag{5.54}$$

where $i, j \in \{x, y, z\}$ represent the Cartesian coordinates. Then, consider the following properties for the function $f(x)$:

$$\partial_i f(|\boldsymbol{x} - \boldsymbol{L}|) = \frac{(\boldsymbol{x} - \boldsymbol{L})_i}{|\boldsymbol{x} - \boldsymbol{L}|} f'(|\boldsymbol{x} - \boldsymbol{L}|), \tag{5.55}$$

and that $\overline{S}^A_{ij} = \lim_{x\to 0} \sum_{\boldsymbol{L}\neq\boldsymbol{0}} \frac{(\boldsymbol{x}-\boldsymbol{L})_i (\boldsymbol{x}-\boldsymbol{L})_j}{|\boldsymbol{x}-\boldsymbol{L}|} f'(|\boldsymbol{x} - \boldsymbol{L}|)$. The following relations between $\overline{S}^A_{ij}$, S^B, $S^C(\boldsymbol{R}_s - \boldsymbol{R}_t)$ and $\overline{S}^D_{ij}(\boldsymbol{R}_s - \boldsymbol{R}_t)$ hold:

$$\begin{aligned} \overline{S}^A_{ij} &= \lim_{x\to 0} \left[\partial_{x_j} \sum_{\boldsymbol{L}\neq\boldsymbol{0}} (\boldsymbol{x} - \boldsymbol{L})_i f(|\boldsymbol{x} - \boldsymbol{L}|) \right] - S^B \delta_{ij} \\ &\equiv S^A_{ij} - S^B \delta_{ij}, \end{aligned} \tag{5.56}$$

$$\begin{aligned} \overline{S}^D_{ij}(\boldsymbol{R}_s - \boldsymbol{R}_t) &= \left[\partial_{\boldsymbol{x}_j} \sum_{\boldsymbol{L}} (\boldsymbol{x} - \boldsymbol{L})_i f(|\boldsymbol{x} - \boldsymbol{L}|) \right]_{x=\boldsymbol{R}_s-\boldsymbol{R}_t} - S^C(\boldsymbol{R}_s - \boldsymbol{R}_t)\delta_{ij} \\ &\equiv S^D_{ij}(\boldsymbol{R}_s - \boldsymbol{R}_t) - S^C(\boldsymbol{R}_s - \boldsymbol{R}_t)\delta_{ij}, \end{aligned} \tag{5.57}$$

Finally, by combining Eqs. (5.49) to (5.54), (5.56) and (5.57) into Eq. (5.13) we can re-write $\boldsymbol{J}^n$:

$$\boldsymbol{J}^n = \boldsymbol{J}^{nA} + \boldsymbol{J}^{nB} + \boldsymbol{J}^{nC} + \boldsymbol{J}^{nD} \tag{5.58}$$

$$\boldsymbol{J}^{nA} = \frac{1}{2} \sum_s \boldsymbol{V}_s M_s V_s^2 \tag{5.59}$$

$$J^{nB}_i = \sum_s V_{si} Z_s^2 e^2 S^B - \frac{e^2}{2} \sum_s \sum_{j\in x,y,z} V_{sj} Z_s^2 S^A_{ij} \tag{5.60}$$

$$\boldsymbol{J}^{nC} = \sum_s \sum_{t\neq s} Z_t Z_s e^2 S^C(\boldsymbol{R}_s - \boldsymbol{R}_t) \boldsymbol{V}_s \tag{5.61}$$

$$J^{nD}_i = -\frac{e^2}{2} \sum_s \sum_{t\neq s} Z_t Z_s \sum_{j\in x,y,z} S^D_{ij}(\boldsymbol{R}_s - \boldsymbol{R}_t) V_{tj}. \tag{5.62}$$

Introducing the matrix $T_{ij} = S^B\delta_{ij} + S^A_{ij}$, which only depends on the cell and not on the individual atoms within it, we can rewrite Eq. (5.60) as a sum of terms that depends only on the atomic species and the cell:

$$J_i^{nB} = \frac{e^2}{2} \sum_{j\in\{x,y,z\}} T_{ij} \sum_S Z_S^2 \sum_{t\in S} V_{tj}, \tag{5.63}$$

where $\sum_S$ is a sum over the atomic species. Recent theoretical developments in the statistical analysis of fluxes [8, 9, 12, 58, 70], and in particular the *gauge* [8] and *convective* [12] *invariances* tell us that fluxes written as sum of terms that depends only on the atomic species do not contribute to the value of the thermal transport coefficient. Thus, $\boldsymbol{J}^{nB}$ can be neglected in the computation of $\boldsymbol{J}^n$, reducing the implemented formula to:

$$\boldsymbol{J}^n = \boldsymbol{J}^{nA} + \boldsymbol{J}^{nC} + \boldsymbol{J}^{nD}. \tag{5.64}$$

We still need to deal with the long-range Coulomb contributions in $\boldsymbol{J}^{nC}$, $\boldsymbol{J}^{nD}$ and $\boldsymbol{J}^{nB}$. Following the scheme explained in Section 5.1 we introduce a Yukawa screened interaction, $\frac{1}{x} \to \frac{e^{-\mu x}}{x}$, and we will study the limit $\mu \to 0$. We can, then, straightforwardly apply the Ewald techniques [80], with a convergence parameter η, for the computation of S^A_{ij}, S^B, $S^C(\boldsymbol{R}_s - \boldsymbol{R}_t)$ and $S^D_{ij}(\boldsymbol{R}_s - \boldsymbol{R}_t)$. Further details of the computation can be found in A.2:

$$\begin{aligned} S^A_{ij} = &- \sum_{\boldsymbol{L}\neq\boldsymbol{0}} \frac{L_iL_j}{L^2}\left[f(L)\mathrm{erfc}(\sqrt{\eta}L) + 2\sqrt{\frac{\eta}{\pi}}e^{-\eta L^2}\right] - 2\delta_{ij}\sqrt{\frac{\eta}{\pi}} \\ &+ \delta_{ij}\sum_{\boldsymbol{L}\neq\boldsymbol{0}} f(L)\mathrm{erfc}(\sqrt{\eta}L) + \sum_{\boldsymbol{G}\neq\boldsymbol{0}} \frac{4\pi}{\Omega}\frac{G_iG_j}{G^2}\frac{\exp(\frac{-G^2}{4\eta})}{G^2}\left[2 + \frac{G^2}{2\eta}\right], \end{aligned} \tag{5.65}$$

$$S^B = \sum_{\boldsymbol{L}\neq\boldsymbol{0}} f(L)\mathrm{erfc}(\sqrt{\eta}L) - 2\sqrt{\frac{\eta}{\pi}} + \frac{4\pi}{\Omega}\sum_{\boldsymbol{G}\neq\boldsymbol{0}} \frac{e^{-\frac{G^2}{4\eta}}}{G^2} + \frac{4\pi}{\Omega}\left(\frac{1}{\mu^2} - \frac{1}{4\eta}\right), \tag{5.66}$$

$$\begin{aligned} S^C(\boldsymbol{R}_s - \boldsymbol{R}_t) = &\sum_{\boldsymbol{L}} f(|\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}|)\mathrm{erfc}(\sqrt{\eta}|\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}|) + \\ &\frac{4\pi}{\Omega}\sum_{\boldsymbol{G}\neq\boldsymbol{0}} \frac{e^{\frac{-G^2}{4\eta}}}{G^2} e^{i\boldsymbol{G}(\boldsymbol{R}_s - \boldsymbol{R}_t)} + \frac{4\pi}{\Omega}\left(\frac{1}{\mu^2} - \frac{1}{4\eta}\right), \end{aligned} \tag{5.67}$$

$$S^D_{ij}(\boldsymbol{R}_s - \boldsymbol{R}_t) = \sum_{\boldsymbol{L}} (\sqrt{\eta} h(\sqrt{\eta}|\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}|)\delta_{ij} + \eta h'(\sqrt{\eta}|\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}|) \frac{(\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L})_i (\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L})_j}{|\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}|} \Bigg) + \\ + \sum_{\boldsymbol{G} \neq \boldsymbol{0}} e^{i\boldsymbol{G}(\boldsymbol{R}_s - \boldsymbol{R}_t)} \frac{4\pi}{\Omega} \frac{G_i G_j}{G^2} \frac{e^{-\frac{G^2}{4\eta}}}{G^2} \left(2 + \frac{G^2}{2\eta}\right), \tag{5.68}$$

where $\mathrm{erfc}(x) = 1 - \mathrm{erf}(x)$, $\mathrm{erf}(x)$ is the error function [80], and $h(x) = \frac{\mathrm{erfc}(x)}{x}$. Since Eqs. (5.65) and (5.68) do not diverge in μ, the divergent parts of the Ionic flux are given only by Eqs. (5.66) and (5.67):

$$\boldsymbol{J}^{nB}_{div} = e^2 \frac{4\pi}{\Omega \mu^2} \sum_s \boldsymbol{V}_s Z_s^2, \tag{5.69}$$

$$\boldsymbol{J}^{nC}_{div} = e^2 \frac{4\pi}{\mu^2 \Omega} \sum_s \sum_{t \neq s} Z_s Z_t \boldsymbol{V}_s, \tag{5.70}$$

$$\boldsymbol{J}^{n}_{div} = \boldsymbol{J}^{nB}_{div} + \boldsymbol{J}^{nC}_{div}, \\ = Z_{tot} e^2 \frac{4\pi}{\mu^2 \Omega} \sum_s Z_s \boldsymbol{V}_s. \tag{5.71}$$

Hartree and Exchange-correlation currents

A finite-difference scheme, explained in detail in section 5.2.2, can be directly implemented to evaluate the scalar fields $\dot{v}^H(\boldsymbol{r})$ and $\dot{n}(\boldsymbol{r})$, which are needed to evaluate $\boldsymbol{J}^H$ and $\boldsymbol{J}^{XC}$. For the Hartree current, the gradient $\boldsymbol{\nabla} v_H(\boldsymbol{r})$ is needed as well. Since the gradient operator in reciprocal space is multiplicative, this suggests to rewrite the entire expression in reciprocal space:

$$\boldsymbol{J}^H = -i \frac{\Omega}{4\pi e^2} \sum_{\boldsymbol{G}} \dot{v}^H(\boldsymbol{G}) v^H(-\boldsymbol{G}) \boldsymbol{G}, \tag{5.72}$$

which is the equation actually implemented. Analogously, for $\boldsymbol{J}^{XC}$ the gradient $\boldsymbol{\nabla} n(\boldsymbol{r})$ is first computed in reciprocal space and then Fourier transformed onto the real grid. In the PBE case, ϵ_{GGA} has an explicit analytical expression as a function of n and $|\boldsymbol{\nabla} n|$. The analytic expression of the derivative $\partial \epsilon_{GGA} / \partial(\boldsymbol{\nabla} n)$ is cumbersome but can be straightforwardly derived from the latter. The resulting expression is then evaluated at the local values of $n(\boldsymbol{r}_i)$ and $\boldsymbol{\nabla} n(\boldsymbol{r}_s)$ for each point $\boldsymbol{r}_s$ of the grid in real space. Finally, contributions from all grid points are summed up.

Electronic density current

As a by-product of the computation of the adiabatic energy flux, `QEHeat` also evaluates the adiabatic electron-number flux by implementing Thouless' expression [81] and using DFPT [79]. To this end, by leveraging the continuity equation, one first formally expresses the number flux as the first moment of the time derivative of the electron number density, to obtain for every Cartesian component i:

$$\begin{aligned} \boldsymbol{J}_i^{el} &= \int \dot{n}(\boldsymbol{r}) r_i d\boldsymbol{r} \\ &= \sum_v \left\langle \phi_v | \hat{r}_i | \dot{\phi}_v \right\rangle + \left\langle \dot{\phi}_v | \hat{r}_i | \phi_v \right\rangle \\ &= 2 \sum_v \left\langle \bar{\phi}_v^{\mathrm{c}}{}_i | \dot{\phi}_v^{\mathrm{c}} \right\rangle . \end{aligned} \tag{5.73}$$

All quantities needed to evaluate the Electronic density current have already been discussed in the section dedicated to the Kohn-Sham current. The electronic flux thus evaluated is interesting *per se*, *e.g.* to compute the electric conductivity in ionic conductors, and also as an ingredient to facilitate the statistical analysis of the energy-flux time series, using *multi-component* [12] or *decorrelation* [70] techniques.

Center-of-mass ionic current and Charge current

The code outputs a trivial but useful current defined for each atomic species as

$$\boldsymbol{J}_S^{CM} = \sum_{t \in S} \boldsymbol{V}_t, \tag{5.74}$$

where S is the atomic species index. The sum is over all atoms of kind S. This current can be used both for data analysis or for computing the Charge current together with Eq. (5.73):

$$\boldsymbol{J}^Q = -e\boldsymbol{J}^{el} + \sum_t \boldsymbol{v}_t e Z_t = -e\boldsymbol{J}^{el} + e \sum_S Z_S \boldsymbol{J}_S^{CM} \tag{5.75}$$

where eZ_S is the pseudo-potential charge of the atom of species S, and $\boldsymbol{J}^{el}$ is the Electronic density current computed by the code, defined in Eq. (5.73).

Divergences

In this section, we discuss the divergences arisen in the computations of Ewald sums in Section 5.2.1. First of all, we note that $\boldsymbol{J}_{div}^0 + \boldsymbol{J}_{div}^n = 0$, showing that the expression for the total MUB flux is free of any divergent term, as we already stated in Section 5.1. We highlight that any divergent term, being $\boldsymbol{J}_{div}^0$, $\boldsymbol{J}_{div}^{nB}$ or

```
    energy_current
    delta_t = 1.000,
    file_output = 'current_hz',
    eta = 0.100,
    n_max = 5,
    trajdir = 'traj/cp',
    first_step = 1,
    vel_input_units = 'CP'
/
```

Figure 5.1: Example of the energy_current namelist. delta_t is the time used for numerical derivatives. eta and n_max are the parameters used to converge Ewald sums. trajdir is the prefix of the trajectory files. In this example, the program reads the files "traj/cp.pos" and "traj/cp.vel". first_step tell the program the first step id to compute. The step ids are part of the trajectory file format. After this namelist the full pw.x input is required. A full example of the input and the documentation of the keywords can be found at `https://gitlab.com/QEF/q-e/-/tree/master/QEHeat/Doc`. See also Section 5.2.3.

$\boldsymbol{J}_{div}^{nC}$, it can be written as a sum of terms depending only on the atomic species, precisely as $\boldsymbol{J}_{div}^{nB}$. Thus, invoking the same *invariance principles* that allowed us to neglect $\boldsymbol{J}_{div}^{nB}$, we can state that any divergent contribution is non diffusive and would not contribute to the transport coefficient.

5.2.2 Code structure

We start by describing the strategy implemented to compute numerical derivatives of quantities appearing in Eqs. (5.11), (5.14), and (5.15), like $\dot{n}$ and $\dot{v}^H$, since they require a special treatment. Quite generally, one needs to evaluate terms of the type $\dot{f}(\{\boldsymbol{R}_s(t)\})$, where the function f can be a scalar function, which depends on time only through the set of the instantaneous ionic positions $\{\boldsymbol{R}_s(t)\}$, evolving according to Hamilton's equations of motion. `QEHeat` implements a finite-difference scheme, using by default a symmetric numerical differentiation formula:

$$\dot{f}(\{\boldsymbol{R}_s\}) \approx \frac{f(\{\boldsymbol{R}_s + \boldsymbol{V}_s dt/2\}) - f(\{\boldsymbol{R}_s - \boldsymbol{V}_s dt/2\})}{dt} \tag{5.76}$$

The small parameter dt is an input of the computation. In such a scheme quantities that are not differentiated are evaluated at time t, so three wave-functions are required to be kept in memory at the same time. `QEHeat` performs therefore for each step two additional self-consistent-field (SCF) DFT calculations, using the same DFT solver of the QUANTUM ESPRESSO distribution, at slightly displaced positions, i.e. $\{\boldsymbol{R}_s - \boldsymbol{V}_s dt/2\}$ and $\{\boldsymbol{R}_s + \boldsymbol{V}_s dt/2\}$ along the AIMD trajectory.

The wave-functions of the previous calculation are used as a starting point for the next one, which require much less iterations to converge. We note that `QEHeat` gives the user also the possibility to use a non-symmetric differentiation scheme, which is shown in 5.2.6. This scheme is computationally cheaper. Nevertheless, for differentiable functions, the order of convergence of the symmetric scheme is quadratic in dt, whereas the non-symmetric one is linear. Accordingly, the stability is improved with the default symmetric scheme. We recommend therefore the latter and use it for all calculations here presented. See also the dedicated 5.2.6.

The trajectory data is managed by the Fortran derived data type `cpv_trajectory` defined in the file `cpv_traj.f90`, while the orbitals and the associated atomic position are managed by the derived type `scf_result`, implemented in `scf_result.f90`. The most relevant subroutine that acts on this object is `scf_result_set_from_global_variable`, that copies the eigenfunctions, the eigenvalues, the potential and the atomic positions from the QUANTUM ESPRESSO's global variables to the instance of `scf_result`. The results for each of the three (or two) wave-functions that are required by the computation routines are stored in the variable `scf_all`, defined in the main program routine.

The code starts by reading the input "namelists": first the `energy_current` namelist, then all the `pw.x` namelists. Then it calls all the `pw.x`-related initialization routines. After eventually reading the previously generated output file that allows the program to set the correct starting timestep, it enters the main loop over the input trajectory timesteps. The trajectory files have the same format of QUANTUM ESPRESSO's `cp.x` code output files.

The most important routines, where the above mentioned data structures are used, are the following:

- `SUBROUTINE current_zero` (module `zero_mod`)
  Carries out the computation of Eq. (5.12). This routine is called in the middle of the computation using the same timestep t of the positions stored in the input trajectory, so that the result does not depend on dt.

- `SUBROUTINE current_ionic` (module `ionic_mod`)
  Computes all parts of Eq. (5.13), and it is called as `current_zero` at the same timestep of the input trajectory

- `SUBROUTINE current_hartree_xc` (module `hartree_xc_mod`)
  Computes (5.14) and (5.15). Since a numerical derivative is needed, this routine reads the wave-functions from the global type `scf_all` and it is run at the end of all necessary `run_pwscf` calls.

- `SUBROUTINE current_kohn_sham` (module `kohn_sham_mod`)

Computes (5.11). As `current_hartree_xc`, it needs all the wave-functions calculated by the DFT solver for this step.

- `SUBROUTINE run_pwscf`
  Uses QUANTUM ESPRESSO's routines to solve the DFT problem for the atomic positions stored in the global array `tau`. Equivalent (but the starting wave-function and potential, that can be the last computed one) to a standard call to the `pw.x` program with the input stripped of the `ENERGY_CURRENT` namelist. The result is stored in the Quantum Espresso's global arrays (`evc`)

- `SUBROUTINE prepare_next_step`
  This routine is used to change the global array `tau` to `tau` + `vel`$\cdot dt \cdot$`ipm`, where `ipm` is the argument of the subroutine that can be -1,0,1. After doing that it calls the necessary routines to prepare the potential for `run_pwscf`.

The 4 modules, one for each part of the MUB current, are completely independent of one another. The structure of the main loop over the trajectory's time steps is summarized in Algorithm 1.

Algorithm 1 Workflow of `all_currents.f90`.

1: QUANTUM ESPRESSO initialization (plane waves, pseudo-potentials,...)
2: Reading of Restart
3: **for** each snapshot **do**
4: call `run_pwscf` with positions displaced at $t - dt/2$
5: call `run_pwscf` with non-displaced positions at t
6: call `current_zero`, evaluate currents derived from the pseudo-potential
7: call `current_ionic`, evaluate the electrostatic and kinetic Ionic current
8: call `run_pwscf` with positions displaced at $t + dt/2$
9: call `current_hartree_xc`, evaluate Excange and Hartree currents
10: call `current_kohn_sham`, evaluate Kohn-Sham current

Steps 6 and 7 do not require any finite differences, while steps 9 and 10 do. Step 10 is the most expensive.

As every big computational code an extended test suite is needed to safeguard the correctness of the calculation after every source code modification. We implemented small tests that are able to run on a single core of a cheap laptop that check against changes in the numerical output of many parts of the code, using the standard QUANTUM ESPRESSO's test suite framework.

To conclude the section we want to do some remarks on the code and its interactivity with others typical *ab initio* simulations tools. In principle the wave-functions computed on-the fly by `cp.x` during the AIMD run could be used, but

we preferred to implement a workflow where the computation of the currents is completely decoupled from the AIMD engine, thus the wave-functions are always recomputed by `pw.x`. The chosen approach allows the user to run the calculation in post-processing mode, thus using the preferred code to generate the dynamics, not to be limited to those in the QE packages. This way it allows, also, a trivial and powerful per-snaphot parallelization.

5.2.3 Input description

The input is organized in a traditional Fortran namelist input file, similar to the input files of many Quantum Espresso's programs, and an optional trajectory file (that is a file for the atomic velocities and a file for atomic positions) if the user wants to compute the energy current for more than one snapshot with a single run. A full example of the input can be found at `https://gitlab.com/QEF/q-e/-/tree/master/QEHeat/examples`. Before running `QEHeat` it is necessary to obtain velocities and positions from a different code, for a complete description of the units of measure see Table 1. If the Quantum Espresso's `cp.x` program is used for this purpose, its output trajectory files can be recycled as input trajectory files without any modification. The program's mandatory input is organized into an `ENERGY_CURRENT` namelist and all the usual `pw.x` namelists. We remind the user that, up to the present version, only norm conserving pseupotentials and the PBE exchange correlation functional are supported. At the end of the input file the `ATOMIC_VELOCITIES` card is required. In the `IONS` namelists the value `ion_velocities = 'from_input'` is required, since the program must read the atomic velocities to compute the energy current. An extensive input description can be found in the documentation, inside the `Doc` folder of the code repository [82] . Here we remark the most important parameters of the `ENERGY_CURRENT` namelist:

- `delta_t` : time in PW's atomic unit used to compute all the numerical derivatives like the one in Eq. (5.76);
- `trajdir` : prefix of the cp-formatted trajectory. Optional: if not setted, only the positions and the velocities of the input file are read;
- `n_max` : number of periodic images along the directions of each basis cell vector to converge Ewald sums. This fixes the range of $\mathbf{L}$ in Eq. (5.13) ;
- `eta` : convergence parameter of the Ewald sums needed in the computation of $\boldsymbol{J}^n$, for more details see Section 5.2.1.

An example of the namelist is provided in Fig. 5.1. An additional output file is written and updated at the end of each step in the folder where the program is

	parameter	units
INPUT	dt	$\tau_{a.u.}$
	eta	$1/a_0^2$
	velocities	$a_0/\tau_{a.u.}$ (CP units can be specified)
	positions	a_0
OUTPUT	energy current	$Ry \cdot a_0/\tau_{a.u.}$
	electronic density current	$a_0/\tau_{a.u.}$
	center of mass currents	$a_0/\tau_{a.u.}$

Table 5.1: Units used for the input and the output, where $Ry = 2.1799 \cdot 10^{-18} J = 13.606 eV$, $a_0 = 5.2918 \cdot 10^{-11} m$, $\tau_{a.u.} = 4.8378 \cdot 10^{-17} s$ are the Rydberg units of energy, the Bohr radius and the time unit in Rydberg atomic units. The program assumes the input velocities to be in Rydberg atomic units, the standard for `pw.x`, unless specified otherwise with vel_input_units='CP' in the energy_current namelist. In that case it assumes Hartree atomic units, the standard for `cp.x`. `cp.x`'s unit of time is $2.4189 \cdot 10^{-17} s$.

run. All the currents are printed in a column format, ready to be analyzed by an external post-processing tool.

As discussed in Section 5.2.6, `CONTROL`'s `conv_thr` and `ENERGY_CURRENT`'s `delta_t` have a profound link and influence heavily each other, and despite we think the default value of `delta_t=1.0` is safe enough, they must be carefully tested, veryfing that the standard deviation of the result is low enough.

The standard deviation of the output energy current can be estimated by repeating the same calculation for every step, many different times, setting for each repetition a random starting potential and a random starting wave-function. The input options `re_init_wfc_1 = .true.` together with `n_repeat_every_step = 20`, for example, do 20 repetition of every timestep, resetting the starting wave-functions/potential before the first scf calculation. The `pw.x`'s input option `startingwfc = 'random'` is suggested, to obtain a faithful error estimation. If more reinitializations are desired, the options `re_init_wfc_2` and `re_init_wfc_3` can control the randomness of the starting wfc and potential of every of the 3 (or 2) wave-functions needed to perform the numerical derivatives, as explained in Section 5.2.2. Note that when the wave-function is reinitialized from scratch, the computation time raises since more scf cycles are required to reach the target convergence threshold. When `n_repeat_every_step` is greater than 1, an additional column formatted output file with the averages and the standard deviations is produced.

5.2.4 Implementation checks: Finite systems translating at constant speed

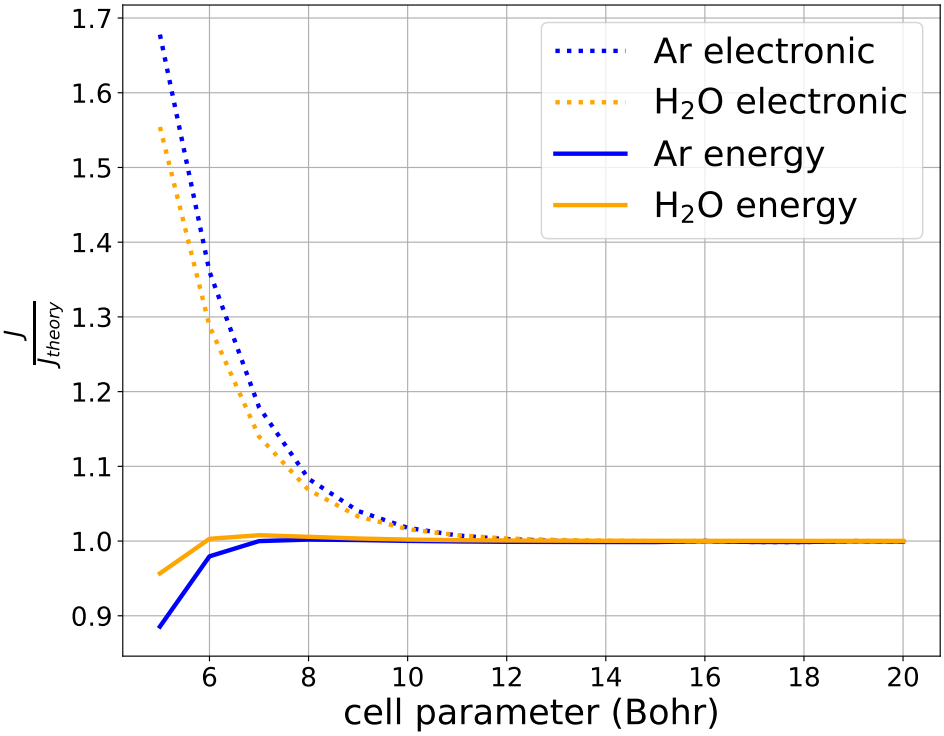


Figure 5.2: Proof that the behaviour of the code is correct for systems translating at equilibrium and constant velocity. In this setting, the output of QEHeat can be exactly compared with the known theoretical value, indicated with J_{theory}, in the large cell limit. Indeed the plot shows that the ratio J/J_{theory} goes to one increasing the simulation cell. This test can be performed for J equal to the electronic density current (dotted), that is used to calculate a part of the energy current, and for the energy current itself (not dotted). Tests are performed for a single relaxed water molecule (H_2O) and a single argon atom (Ar). The electronic current should be, in the infinite cell limit, $\boldsymbol{J}^{el}_{\text{theory}} = N_{el} \times \boldsymbol{v}$, where N_{el} is the number of electrons and the energy flux should be, in the infinite cell limit, equal to $\boldsymbol{J}^{ene}_{theory} = E^{tot} \times \boldsymbol{v}$, E^{tot} being the total energy.

The Green-Kubo current associated with a localized energy density $\epsilon(\boldsymbol{r}, t)$ rigidly translating with constant velocity $\boldsymbol{v}$, is equal to $E^{tot} \times \boldsymbol{v}$. One possible way to show this is to consider $\epsilon(\boldsymbol{r}, t) = \epsilon(\boldsymbol{r} - \boldsymbol{v}t, 0) \equiv \epsilon^0(\boldsymbol{r} - \boldsymbol{v}t)$, thus $J_a = \int \dot{\epsilon} r_a d\boldsymbol{r} = -v_b \int (\partial_b \epsilon^0) r_a d\boldsymbol{r} = v_a \int \epsilon^0 d\boldsymbol{r} = E^{tot} v_a$. Note that we used the fact that ϵ^0 can be taken identically equal to zero at the boundary of the integration volume, to remove boundary contributions from the integration by parts. The identity requires therefore the energy density to be localized and this condition can be mimicked in PBC considering a large enough supercell. We used this property to check the correctness of our implementation for each individual current in

Eq. (5.10). We simulate a single Argon atom and a water molecule at equilibrium, both translating at constant speed. We then compare $\boldsymbol{J}^{MUB}$ output from `QEHeat` and $E^{tot}\boldsymbol{v}$, where E^{tot} is evaluated using an independent computation from the QE code. As discussed, the resulting currents need to be equal only in the limit of large cells, where boundary effects can be neglected, i.e. the energy density is truly localized, and under tight convergence criteria. In Fig. 5.2, we report the ratio between the computed and theoretical values as a function of the cell parameter, showing that the correct limit behaviour is recovered. For this calculation we used a cutoff of 120 Ry and econv $= 10^{-14}$ Ry. In the same figure, using the same approach, we tested the electronic density current defined in Eq. (5.73) as well. In the infinite cell limit the electron density current of a system translating at a constant speed $\boldsymbol{v}$ is $\boldsymbol{J}^{el} = N_e\boldsymbol{v}$ where N_e is the number of electrons. It is possible to see in Fig. 5.2 that the correct limit is obtained, validating the implementation of the code.

5.2.5 Implementation check of individual currents

In the case of a finite system at equilibrium translating at constant speed $\boldsymbol{v}$, the current $\boldsymbol{J}^{MUB}$ must be equal to $E^{tot} \times \boldsymbol{v}$. In Fig. 5.3 we report the modulus of $\boldsymbol{J}^{MUB} - E^{tot}\boldsymbol{v}$, normalized by the module of $E^{tot}\boldsymbol{v}$ and indicated with REF_ERROR. In order to check the correct implementation of each individual current, we also report values of Δ_X , which represent the same quantity after the substitution $\boldsymbol{J}^{MUB} \to \boldsymbol{J}^{MUB} - \boldsymbol{J}^{X}, X \in \{\mathrm{XC}, \mathrm{IONIC}, \mathrm{ZERO}, \mathrm{KOHN}\}$. If the current X is correctly computed, the error with respect to the reference value $E^{tot}\boldsymbol{v}$ should increase, as is indeed observed. For Argon, we used the QE parameters ecut = 160 Ry and econv $= 10^{-16}$ Ry. For water ecut = 120 Ry, econv $= 10^{-14}$ Ry. We used a cubic simulation cell of 20 and 30 Å for Argon and water respectively.

5.2.6 Numerical stability of `QEHeat`

In the following subsection, we show the stability and convergence properties of a `QEHeat` calculation on a snapshot of 125 water molecules.

Fig. 5.4 reports a scaled version of the three Cartesian components of the MUB energy flux $J_i(dt), i \in \{x, z, y\}$, as a function of the time-discretization parameter dt. The $x-$axis is in units of the Car-Parinello MD simulation timestep, indicated with Δt. For each Cartesian coordinate, the behaviour of the error $(J_i(dt) - J_{\mathrm{REF},i})/J_{\mathrm{REF}}$, considering a reference and scale values, is reported, thus showing possible non-linear contributions due to the choice of a large dt. The reference value $J_{\mathrm{REF},i}$ is evaluated for each coordinate at the smallest value of dt available and $J_{REF} = |\boldsymbol{J}_{\mathrm{REF}}|$.

Fig. 5.4 shows the presence of small non-linear effects for higher values of dt,

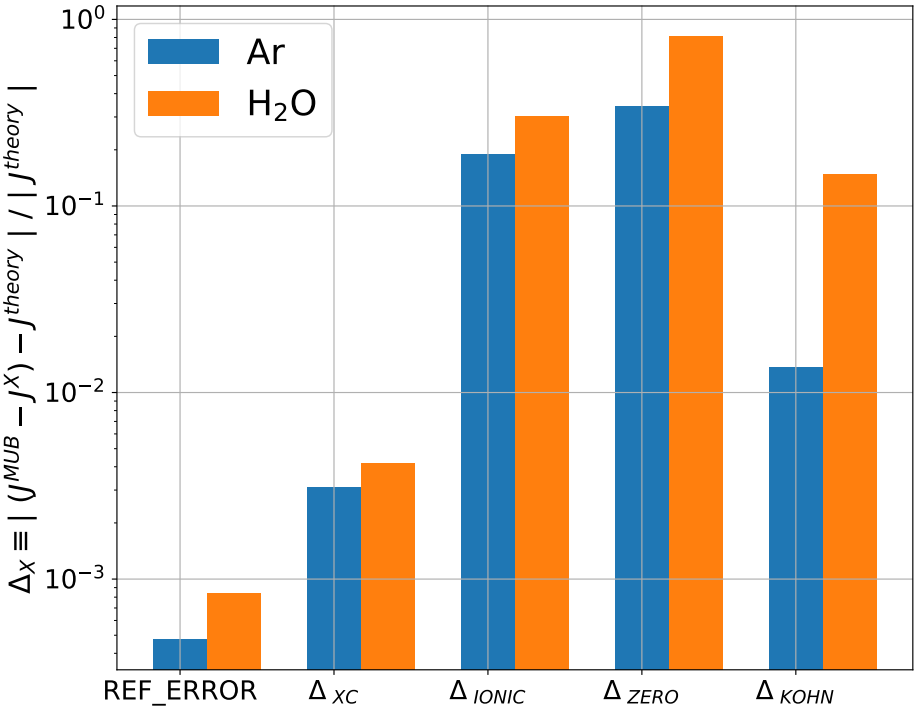


Figure 5.3: In the case of a finite system at equilibrium translating at constant speed $\boldsymbol{v}$, the current $\boldsymbol{J}^{MUB}$ must be equal to $\boldsymbol{J}^{theory} \equiv E^{tot} \times \boldsymbol{v}$. We report in the picture on the leftmost histogram REF_ERROR $= |\boldsymbol{J}^{MUB} - \boldsymbol{J}^{theory}|/|\boldsymbol{J}^{theory}|$ in the case of an Argon atom and a water molecule translating at constant speed. Ideally REF_ERROR should be identically zero, but for numerical reasons it can be just a small value. To check that REF_ERROR is indeed small and at the same time to validate the correct implementation of each individual current, Δ_X represents the same quantity after the substitution, in the numerator of the fraction, $\boldsymbol{J}^{MUB} \to \boldsymbol{J}^{MUB} - \boldsymbol{J}^X, X \in \{\mathrm{XC}, \mathrm{IONIC}, \mathrm{ZERO}, \mathrm{KOHN}\}$. Since removing a current component increases the error w.r.t. the expected value significantly, this proves the correct implementation of each individual current. See text for the parameters used in the computations.

and that, at least for the presented system, nonlinear effects do not take off substantially up to $dt = 2\Delta t$. At this value of dt, even reducing it by a factor of 10 would change the component of the current by less than 0.001%, a negligible error given that a typical value of the thermal transport coefficient has an accuracy of 10% [8, 12, 83]. Moreover $dt = 2\Delta t$ would be beneficial in an on-the-fly computation, allowing to reuse the same wavefunctions computed in the MD simulation, neglecting the need for the recomputation of the scf cycles. The errorbars in the figure are computed using the testing feature, provided with `QEHeat`, presented

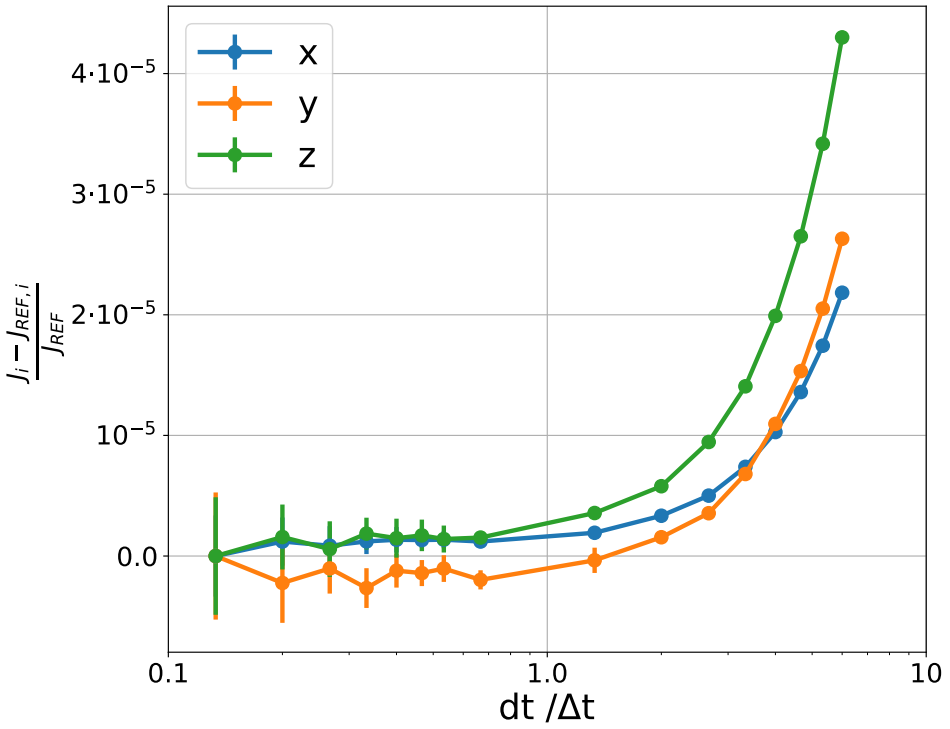


Figure 5.4: The behaviour of $(J_i(dt) - J_{\mathrm{REF},i})/J_{\mathrm{REF}}$ as a function of the timestep dt used in the symmetric derivative for a snapshot of H_2O, where $J_{\mathrm{REF}} = |\boldsymbol{J}_{\mathrm{REF}}|$ and $i \in \{x, y, z\}$. The symbol $J_i(dt)$ refers to a component of the MUB flux computed with a discretization step dt, as reported in the x-axis. The reference value $J_{\mathrm{REF},i}$ is evaluated for each coordinate at the smallest value of dt available, in the example taken to be with $dt/\Delta t = 0.13$. The x-axis is in units of the simulation step Δt. The picture shows that the code can handle correctly even small values of dt. At the same time, it shows that the non-linear effects due to high values of dt are negligible up to the range of dt explored, and in particular negligible at $dt = 2\Delta t$. The errorbar are an estimate of the statistical uncertainty computed using the testing feature presented in Section 5.2.3 and using 20 different initialization of the wavefunctions.

in Section 5.2.3 and averaging 20 fluxes obtained from different initial random wave-functions.

The previous calculations were performed with the self-consistent threshold `econv` equal to 10^{-14} Ry. The lower, the better the quality of the wavefunctions calculated. In order to further test and prove the stability of the MUB current, we used again the aforementioned testing feature to compute the dt dependence of the statistical uncertainty of the MUB current, at different values of `econv`. Fig. 5.5 shows the percentage error of the x-component of the flux for a specific snapshot of 125 molecule. Even though the dependence on dt is similar for the

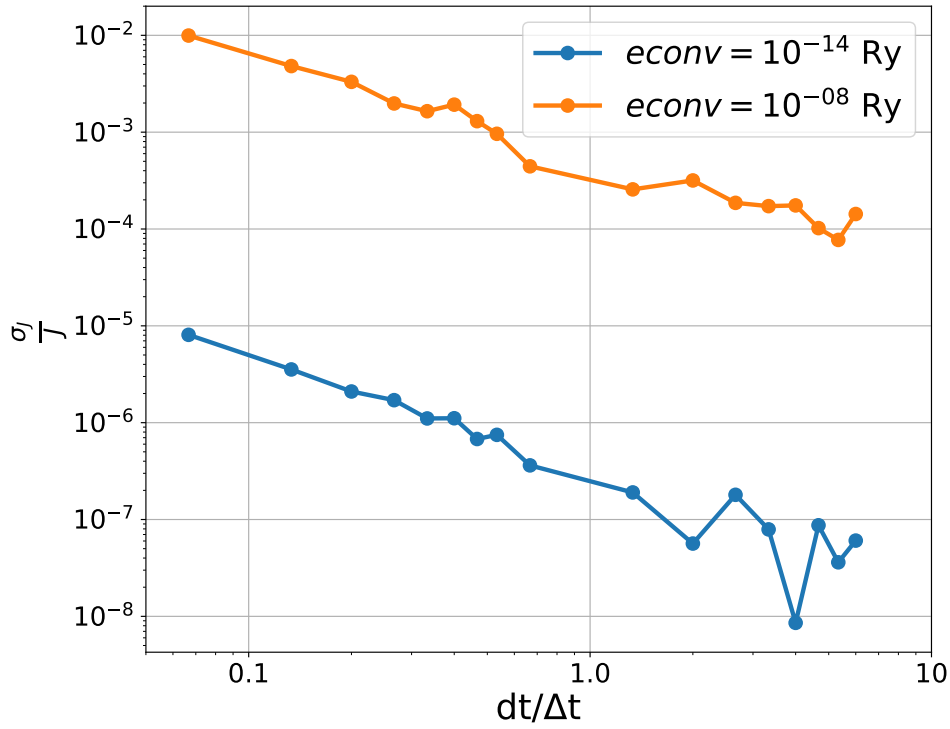


Figure 5.5: Percentage error on the x component of the total energy flux for a selected snapshot of H_2O . The source of error here considered is due to the inherent noise on the wave-functions computed by QUANTUM ESPRESSO and was estimated restarting `QEHeat` 20−times with different seeds. The error is plotted for different values of `econv` as function of the timestep dt used in the symmetric derivative. Note that the error decreases decreasing `econv`, as expected, but also increasing dt. Not shown here is the onset of the non-linear behavior due to a too large dt, visible, instead, in Fig. 5.4. (see also Section 5.2.6). Different colors represent different values of `econv`.

two values of `econv`, it is clear that increasing `econv`, at a fixed dt, increases the statistical error by orders of magnitude, three orders when `econv` is changed from 10^{-14} Ry to 10^{-08} Ry and at any fixed dt. Note also that the variance decreases when increasing dt, in an exponential way. It must be stated that the errors in Fig. 5.5 do not include the effect of the non-linearity due to a too large dt, bringing a (small) bias to the estimation of the MUB flux which can be seen, for example, in Fig. 5.4.

In Section 5.2.2 we showed how `QEHeat` implements numerical derivatives with the symmetric approach. The code allows also, simply changing the `three_point_derivative` keyword to `false`, to compute the derivative within

a non-symmetric approach. In this case the implemented expression is:

$$\dot{f}(\{\boldsymbol{R}_s\}) \approx \frac{f(\{\boldsymbol{R}_s\}) - f(\{\boldsymbol{R}_s - \boldsymbol{V}_s dt\})}{dt}, \tag{5.77}$$

removing one scf computation with respect to the symmetric derivative scheme. Quantities that are not numerically time-derived are evaluated with atoms at positions $\{\mathbf{R_s}\}$. For the same snapshot of water of Fig. 5.5 and *ecut* = 85 Ry, *econv* = 10^{-14} Ry, we computed the energy current with both the non and symmetric derivative approaches. Figure 5.6 shows that the latter returns an energy flux by far more numerically stable with *dt*. However, in both cases, the values of the current only slowly deviate from a constant behaviour, after increasing dt. Even the 2-point derivative gives results that differ of few percentage points.

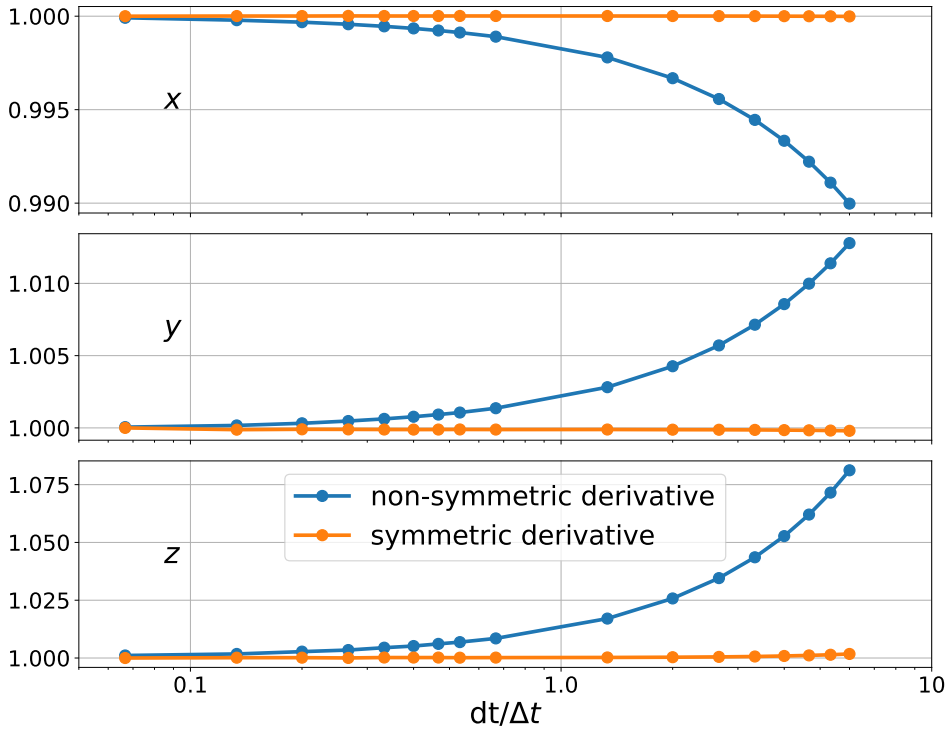


Figure 5.6: The behaviour of $J_i(dt)/J_{\mathrm{REF}}$ as function of the timestep *dt* used in the numerical derivative. The three panels compare the results obtained with the non and symmetric derivative, each panel shows a different component of the energy current J_i. J_{REF}, taken as reference, is the modulus of MUB flux computed for $dt = 0.66\Delta t$ and the symmetric derivative approach. The errorbars are obtained repeating the computation of the current with a different initial wave-function. The picture also shows that for the non-symmetric derivative there are only slightly stronger non-linear behavior on the current for large *dt*.

5.2.7 Example of a thermal conductivity calculation

Our system contains 64 molecules water system at 600 K and at PBE level of theory and we performed a *ab initio* MD trajectory using a cubic box with size of 12.43Å and an integration timestep $\Delta t = 3$ in Hartree atomic units, for a total simulation length of 240ps. For these calculations, we first investigated two values of `econv`, namely 10^{-8} and 10^{-11} Ry with a rather small $dt = 0.5$ Rydberg atomic units. The final result of the computation, i.e. the thermal conductivity coefficient, is affected very weakly by this parameter, that on the contrary changes in a noticeable way the computational cost. In this case both computations gave (1.07 ± 0.09)W/mK. Moreover we checked that using a dt that is equal to the integration timestep of the Car-Parrinello simulation and econv $= 10^{-8}$ produces the same spectrum with no noticeable differences and the same thermal conductivity coefficient: in this case we had (0.98 ± 0.09)W/mK.

According to our experience, the overall cost of a `QEHeat` calculation is often of the same order of magnitude of the full Car Parrinello molecular dynamics simulation even if the `QEHeat` computation can be trivially parallelized. The data analysis with `SPORTRAN` adds a negligible computational cost. The total computational time can be also reduced choosing an appropriate length of the simulation, according to the desired precision.

Fig. 5.7 show the dependence of κ from the simulation length both for our water system (upper panel) and a solid ice X structure from Ref. [83] (lower panel), where the oxygen are packed in a BBC lattice and tetrahedrally coordinated to hydrogen atoms located exactly midway between two neighbouring oxygen atoms. In the present didactic work we choose a very long simulation time of 240 ps but from the figure is it clear that a simulation length around 100 ps can provide a reasonable estimate. Depending on the system, even shorter simulations lengths could suffice: the lower panel of Fig. 5.7 presents the same data of ice X from Ref. [83], where a simulation of $20 - 30$ ps proved to be sufficient.

5.2.8 Computational Cost

In order to evaluate the computational cost of a QEHeat calculation consider a test systems of 64 molecules water system at 600 K and at PBE level of theory. We performed a molecular dynamics trajectory with the cp.x code using a cubic box with size of 12.43Å and an integration timestep $\Delta t = 3$ in Hartree atomic units, with a total length of 240ps.

The actual cost of a calculation depends on several factors that one should take into consideration, e.g.:

- *Typical decaying times.* Decaying times, defined as time lengths when the relevant autocorrelation functions become negligible, can vary a lot according to

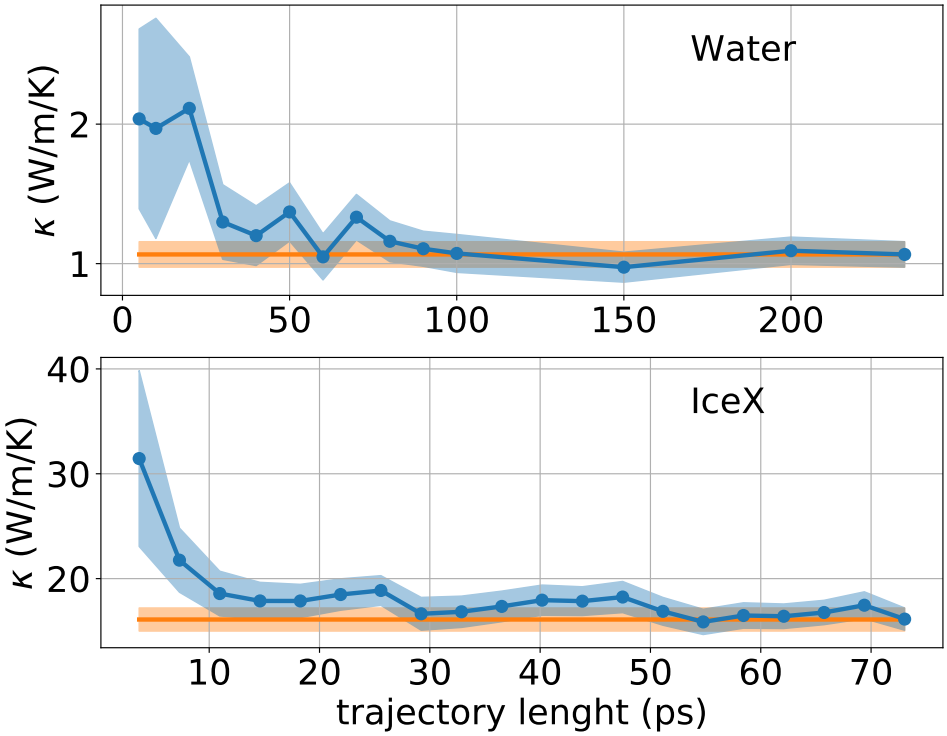


Figure 5.7: The dependence of κ with respect to the trajectory length for 64 molecules of water system at 600K and for the ice X system in [83]. The orange line correspond to the values with the longest trajectory possible, $\kappa = 1.07 \pm 0.09$ W/(mK) and $\kappa = 16.1 \pm 1.1$ W/(mK) for water and ice X respectively. The data for the ice X system are taken from the Material Cloud repository [84]

the system. Longer decaying times require longer simulations to acquire enough statistics. Typically, liquid materials show smaller decaying times. Other convenient situations are solids at high pressures and temperatures, or glasses.

- *Sampling frequency.* It is not necessary to compute the energy flux for every single step of the molecular dynamics trajectory, in order to avoid unneeded wastes of computer time. The optimal sampling frequency can depend on the chosen methodology to extract the thermal conductivity coefficient from the time series of the energy flux. More information about the data analysis in Chapter 4. We evaluated the heat flux every 60 units of time in Hartree atomic unit, i.e. 20 Car-Parrinello molecular dynamics steps in our simulation.

- *Simulation cell sizes.* One has to take into account that strongly harmonic systems, like crystalline solids at ambient temperature, may require large simulation cells to remove boundary effects. For high temperature or high pressure solids this should be a smaller issue. Anharhmonic effects in disordered sys-

tems or glasses can also reduce the typical path lengths and reduce therefore boundary effects.

- *Required accuracy.* All settings should be tuned according to the desired accuracy on the thermal conductivity coefficient, which may vary according to the application.

We provide here the overall computational cost of the computation presented in section 5.2.7 for liquid water, as a guideline, even if system specific setups are suggested. For this test, we chose conservative parameters and the default symmetric derivative scheme. We ran the MD calculations with 4 nodes, 192 processors, whereas the `QEHeat` calculations, being trivially parallelizable, where each run on 1 node and 48 processors. All times here reported are multiplied by the number of processors and identify therefore the total cost of the computation.
For the 64-water molecules system of Section 5.2.7, a general MD step costs $\approx 1.2 \times 10^2$s of cpu time (average over $\approx 2.5 \times 10^5$ steps, leading to a trajectory 240ps long). The cost of evaluating $\boldsymbol{J}^{MUB}$ for a single snapshot was $\approx 1.5 \times 10^3$s of cpu time, in which 84% of the time is reserved to the solution of the linear system in Eq. (5.11) during the evaluation of $\boldsymbol{J}^{KS}$. The 3 minimizations needed for the symmetric derivative scheme cost all together $\approx 2.2 \times 10^3$s (mean over all PW calculations). Evaluating $\boldsymbol{J}^{MUB}$ every 20 MD timesteps, the overall overhead becomes around 1.5 times the cost of the whole *ab initio* molecular dynamics simulation.
To conclude, at the present state our code recomputes the self-consistent cycles for each step, adding some extra time that we considered in this analysis. There are undergoing work trying to use a more on-the-fly approach, directly taking advantage of the wafefunctions provided during the Car-Parrinello MD simulation. A promising feature considering that our analysis on the dt dependence implies that the discretization step can be chosen equal to the MD time step, and it will be available in a future releases. All the computations are done on the Tier-0 system called Marconi CINECA, which have 3188 nodes equipped with 2 Intel Xeon 8160 (SkyLake) at 2.10 GHz with 24-cores each [85].

6

Ab initio *simulations of thermal transport: results*

In the following chapter, we will present the results for thermal conductivity of water from AIMD simulations from Ref. [86].

We performed four AIMD simulations of water, corresponding to different temperatures and phases, using the PBE functional approximation of DFT, the plane-wave pseudopotential method, and periodic boundary conditions. Hamann-Schlüter-Chiang-Vanderbilt norm-conserving pseudopotentials [87] were used with a kinetic-energy cutoff of 85 Ry. All the simulations were performed with the Car-Parrinello extended-Langrangian method [7] using the `cp.x` component of QUANTUM ESPRESSO [73–75] and setting the fictitious electronic mass to 25 physical masses and the timestep to $dt = 0.073$ fs. Liquid water simulations were done with 125 water molecules inside a cubic computational box of side $l = 15.52$ Å, hexagonal ice-Ih simulations used 128 water molecules inside an orthogonal cell, with sides: $l_1 = 18.084$ Å, $l_2 = 15.664$ Å and $l_3 = 14.724$ Å. It is known that within the PBE XC functional approximation, liquid water exhibits enhanced short-range order [35, 36] and a melting temperature that is more than 100 K higher than in experiment [88, 89], while solid ice has higher density than liquid water at co-existence. In order to compensate for this shortfall, it is customary to offset the simulation conditions by increasing the temperature by ≈ 100 K. We performed simulations of the liquid at three temperatures (521 K, 431 K and 409 K), and of ice in the hexagonal Ih structure at 260 K. Each simulation was 100 ps long. Then, using the `QEHeat` [43] code, we computed the MUB flux every 3.1 fs. The statistical noise affecting the estimates of the GK integrals is larger when the spectral power of the flux time series is larger. Because of gauge invariance, different representations of the energy current may carry a very different spectral power, and still yield the same conductivity, which is the zero-frequency limit of the flux power spectrum. The MUB energy flux turns out to carry an impractically large spectral power, cause by non-diffusive signals, which can be tamed to some extent by leveraging gauge and convective invariance. Gauge invariance is first exploited by the *velocity renormalization* technique of Ref. [70]. In a nutshell, it can be demonstrated that subtracting to each atomic velocity the average velocity of all the atoms of the same chemical species, results in a current with a much reduced spectral weight but the same conductivity. Further spectral weight can be subtracted by adding to the resulting effective flux any linear combination of non-diffusive fluxes. This

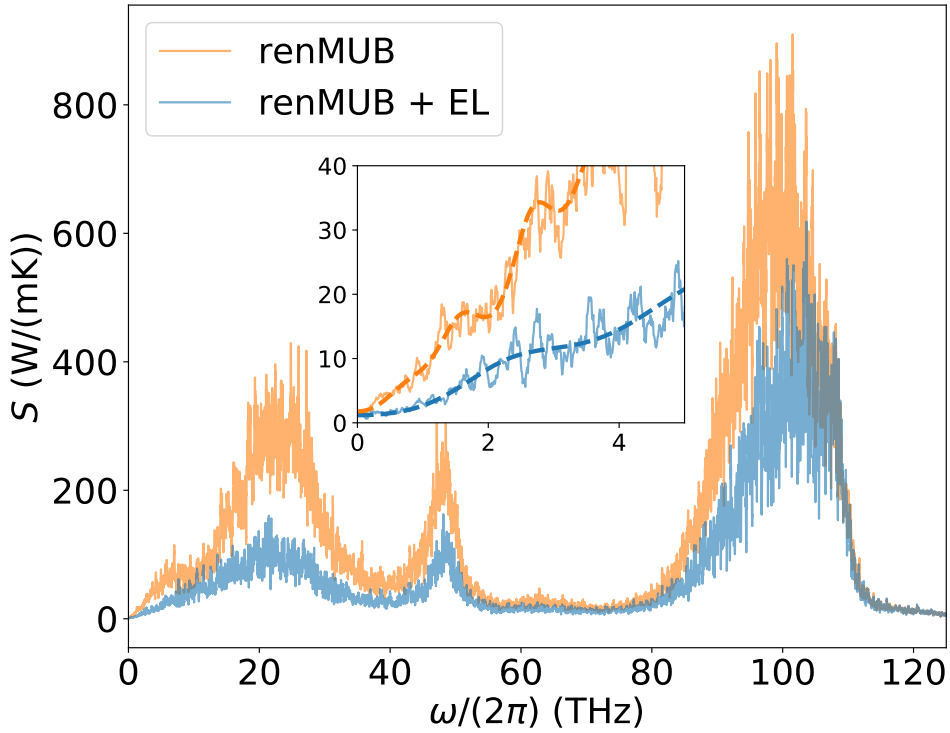


Figure 6.1: Comparison of the (window-filtered) spectrum of the velocity renormalized MUB flux (orange) and of the velocity renormalized MUB flux decorrelated with the adiabatic electronic flux (blue). Both spectrum are filtered with a moving average of 0.1 THz. The renormalized MUB flux has a higher power but close to zero the two spectra converge to the same value. The two dashed lines in the inset represent the cepstral filters of the power spectra.

can be effectively done by treating the (possibly renormalized) energy current as one component of an Q-component system, where all the other currents are non-diffusive ones [12], as in Fig. 4.3. Here, we choose $Q = 2$ and take the electronic adiabatic current as the auxiliary non-diffusive one. In all cases, the transport coefficients are obtained from the *cepstral analysis* of the power spectrum of the relevant currents, using the `SporTran` [68] code.

Fig. 6.1 displays the (window-filtered) power spectrum of the MUB flux from one of our Car-Parrinello MD simulations of liquid water at an average temperature of 431 K, using renormalized velocities (orange line), and further removing the contribution of the adiabatic electron current from the energy flux (blue line). In the inset we see that the two spectra converge to the same value when $\omega = 0$. The decorrelation decreases the power of the spectrum and flattens the spectrum near $\omega = 0$ facilitating data analysis by reducing the number of the required cepstral coefficients.

phase	T	κ	f^*	P^*
	K	W/(mK)	THz	
liquid	521	0.98 ± 0.19	20.7	55
liquid	431	1.06 ± 0.11	20.1	17
liquid	409	1.12 ± 0.17	45.9	33
ice Ih	260	1.8 ± 0.4	30.3	53

Table 6.1: Properties of water from AIMD. All liquid simulations used 125 H_2O molecules inside a cubic box of side $l = 15.52$ Å. The ice Ih simulations used 128 H_2O molecules inside an orthogonal cell with sides: $l_1 = 18.084$ Å, $l_2 = 15.664$ Å and $l_3 = 14.724$ Å. T is the mean temperature of the simulations and κ is the thermal transport coefficient. The lasts two columns contain the values of f^* and P^* used for the *ceptral analysis*.

Table 6.1 contains the thermal conductivities computed from AIMD using `QEHeat`and DPMD for all the simulations that we performed. The results differ substantially from experiment ($\kappa_{expt} \approx 0.6$ W/(mK) *vs.* $\kappa_{PBE} \approx 1$ W/(mK) for liquid water at near ambient conditions [90]). This discrepancy might be ascribed mostly to inaccuracies of the PBE functional, which is known to not provide a correctly structured water[35, 36, 88]. In Chapter 8 we will use neural-network potentials to extend our study with a more advanced functional approximation, the *strongly constrained and appropriately normed* (SCAN) meta-GGA [39].

7

Deep Potential model

The neural-network potentials (NNP) based on the deep potential (DP) framework uses a local decomposition of the total potential energy of the system in terms of atomic contributions, which makes straightforward to define the energy flux from which to compute the heat conductivity via the GK theory. Deep potential molecular dynamics (DPMD) simulations have been used successfully to study bulk thermodynamic properties beyond the reach of direct DFT calculations [19–25] as well as dynamic properties like mass diffusion in solid state electrolytes [26, 27], infrared spectra of water and ice [28] and Raman spectra of water [91].

7.1 Deep Neural Networks

For a first introduction to the concept of deep neural networks (DNN) we follow chapter 6 of Ref. [92]. The goal of a neural network is to approximate some function $f^\star$ [92]. They are called networks because they are typically represented by composing together many different (non linear) functions. We can consider three functions $f^{(1)}, f^{(2)}, f^{(3)}$ connected in a chain, to form $f(\boldsymbol{x}) = f^{(3)}\left(f^{(2)}\left(f^{(1)}\left(\boldsymbol{x}\right)\right)\right)$. In this case, $f^{(1)}$ is called *first layer*, $f^{(2)}$ is called the *second layer* and so on. The number of functions chained gives the *depth* of the network. The final layer is called the *output layer*. During the training, we provide approximate examples of $f^\star((x)$ evaluated for different values of x, thus driving $f(x)$ to match $f^\star(x)$. The training data specify what the output layer must do for each input, but do not specify what any other intermediate layer should do, leaving this decision to the optimization algorithm. For that reason, they are called *hidden layers*. The elements of the layers are called *neurons* and, in most neural network, their functional form is a linear transformation defined by learned parameters followed by a fixed non linear function, called *activation function*:

$$f_j^{(1)} = \sigma\left(b_j^{(1)} + W_{i,j}^{(1)}\mathrm{I}_i\right), \tag{7.1}$$

where b_j are called biases, W_{ij} weights and σ, the activation function. A general neural network with L hidden layers can be written as:

$$\boldsymbol{y}(\boldsymbol{I};\boldsymbol{W}) = \boldsymbol{f}^{(L)}\left(\boldsymbol{f}^{(L-1)}\left(\cdots \boldsymbol{f}^{(1)}\left(\mathbf{I}\right)\right)\right) \tag{7.2}$$

$$= \sigma\left(\boldsymbol{b}^{(L)} + \boldsymbol{W}^{(L)} \cdot \sigma\left(\boldsymbol{b}^{(L-1)} + \boldsymbol{W}^{(L-1)} \cdot \sigma\left(\cdots \sigma(\boldsymbol{b}^{(1)} + \boldsymbol{W}^{(1)} \cdot \mathbf{I})\right)\right)\right), \tag{7.3}$$

where the symbol $\boldsymbol{W}$ denotes a dependence from both the weights $\boldsymbol{W}^{(l)}$ and the biases $\boldsymbol{b}^{(l)}$.

Fig. 7.1 shows a pictorial representation of a DNN, where in green is reported the input vector, in blue the output, the hidden layers are in orange and the grey lines represent the weights.

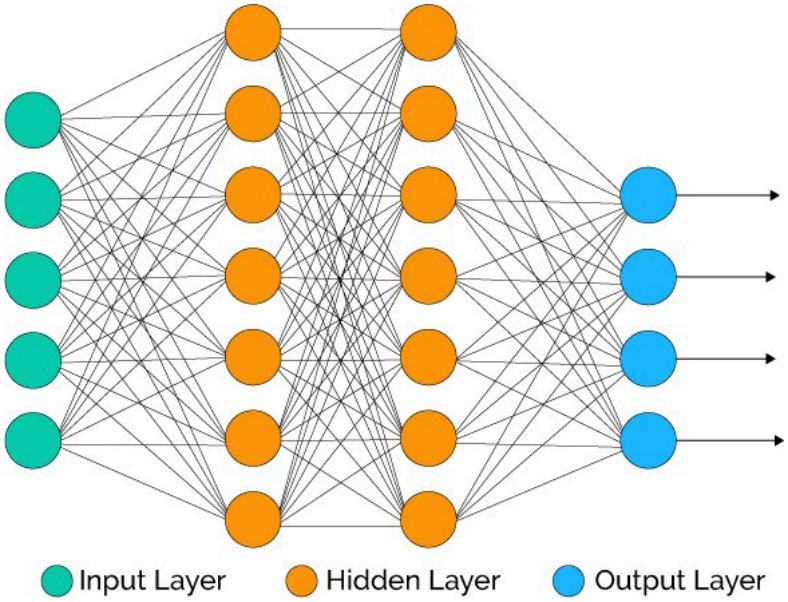


Figure 7.1: A pictorial representation of a DNN with two hidden layers, the grey lines correspond to the weight and biases that form the connection inside the network. The input $\boldsymbol{I}$ (green circles) is a vector of dimension 5, the output (blue) has dimension 4 and the 2 hidden layers (orange) have 7 neurons each.

The model is usually optimized minimizing a loss function L in the set of parameters $\boldsymbol{W}$, using standard algorithms like the the stochastic gradient descent [92] or Adam [93]. In many applications L takes the following form:

$$L(\boldsymbol{W}) = \frac{1}{B}\sum_{b=1}^{B} \left|\boldsymbol{y}(\boldsymbol{I}_b;\boldsymbol{W}) - \boldsymbol{f}^{\star}(\boldsymbol{I}_b)\right|^2, \tag{7.4}$$

where $\boldsymbol{f}^{\star}$ is the target function and b is summed over a batch of B training data set.

7.2 Construction of neural-network potentials

Descriptor

In order to construct a NNP, the first step is to build a descriptor that encodes each atomic environment. Consider a system of N atoms, whose configurations are represented by the set of atomic positions, $\{\boldsymbol{R}_1, \boldsymbol{R}_2, \ldots, \boldsymbol{R}_N\} \in \mathbb{R}^{3N}$. For each atom, n, we consider only the neighbours, $\{q\}$, such that $R_{qn} < R_c$, where R_{qn} is the modulus of the vector $\boldsymbol{R}_{qn} = [x_{qn}, y_{qn}, z_{qn}] \doteq \boldsymbol{R}_q - \boldsymbol{R}_n$, and R_c is a pre-defined cut-off radius. Denoting with N_n the number of neighbours of n within the cutoff radius, we define the *local environment matrices* $\tilde{K}_n \in \mathbb{R}^{N_n \times 4}$ to encode the local environment:

$$\tilde{K}_n = \begin{bmatrix} \frac{\sigma(R_{1n})}{R_{1n}} & \frac{\sigma(R_{1n})x_{1n}}{R_{1n}^2} & \frac{\sigma(R_{1n})y_{1n}}{R_{1n}^2} & \frac{\sigma(R_{1n})z_{1n}}{R_{1n}^2} \\ \frac{\sigma(R_{2n})}{R_{2n}} & \frac{\sigma(R_{2n})x_{2n}}{R_{2n}^2} & \frac{\sigma(R_{2n})y_{2n}}{R_{2n}^2} & \frac{\sigma(R_{2n})z_{2n}}{R_{2n}^2} \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}, \tag{7.5}$$

where $\sigma(R_{qn})$ is the smoothing function:

$$\sigma(R_{qn}) = \begin{cases} 1 & R_{qn} < R_{c1} \\ -6\Theta^5 + 15\Theta^4 - 10\Theta^3 + 1 & R_{c1} < R_{qn} < R_c \\ 0 & R_c < R_{qn} \end{cases} \tag{7.6}$$

where $\Theta = \frac{R_{qn} - R_{c1}}{R_c - R_{c1}}$ and R_{c1} is the smoothing cut-off radius, that switch the interaction smoothly to zero. The model encode the local environments within the cutoff R_c, thus it assumes that all the important interaction can be described within. We want our descriptor to preserve the natural symmetries of our system: rotation, translational invariance and invariance exchanging two atoms of the same species. We want to transform each $\tilde{K}_n$ in a symmetry-preserving input for neural network. The first point is to define a *local embedding network* $G^n = G(\frac{\sigma(R_{qn})}{R_{qn}})$ as a neural network mapping, depending only on the atomic species of atom n and its neighbour m, from a scalar value $\frac{\sigma(R_{qn})}{R_{qn}}$ to a vector of size M_1. As last ingredient, let's take the first M_2 columns of G^i and constructing the matrix $G^{n2} \in \mathbb{R}^{N_n \times M_2}$. Finally, the symmetry-preserving descriptors (called encoded feature matrix in Ref. [18]) $D^n \in \mathbb{R}^{M_1 \times M_2}$ of atom n can be defined:

$$D^n = (G^n)^T \tilde{K}^n (\tilde{K}^n)^T G^{n2}. \tag{7.7}$$

Every D^n is, then, reshaped into a vector and fed to a "fitting" DNN which returns the "atomic contributions", w_n, to the energy. A more direct derivation of the

symmetries of D^n can be found in the Supplemental material of Ref. [18]. The most distinctive feature of DP-based models reside within the embedding matrix. The symmetry preserving features functions are learned on-the-fly during the training process, while, for example, the popular Behler-Parrinello approach [13] requires a list of pre-defined parameters, that require human intervention.

Energy, forces and stress

We denote by $\boldsymbol{W}$ the full set of parameters that define the total potential energy, E. Thus, as illustrated in Ref. [18], the extensive property of E is ensured by its decomposition into "atomic contributions", w_n:

$$E^{\boldsymbol{W}}(\{\tilde{K}\}) = \sum_n w^{\boldsymbol{W}_{\alpha_n}}(\tilde{K}_n) \equiv \sum_n w_n \tag{7.8}$$

where α_n denotes the chemical species of atom n. We use the notation $(\dots)^{\boldsymbol{W}_{\alpha_n}}$ to indicate that the parameters used to represent the "atomic energy" only depend on the chemical species α_n of the n-th atom [18]. Being w_n a well defined and easy to compute function of the atomic positions, the atomic forces and their breakup into individual atomic contributions, $\frac{\partial w_m}{\partial \boldsymbol{R}_n}$ (needed in the definition of the energy flux), can be easily computed as the gradients of E and w_n, respectively. In particular, the computation of the latter can be divided into two contributions by applying the chain rule:

$$\boldsymbol{F}_n = \nabla_{\boldsymbol{R}_n} E^{\boldsymbol{W}}(\{\tilde{K}\}) = \sum_m \nabla_{\boldsymbol{R}_n} w_m \tag{7.9}$$

$$\nabla_{\boldsymbol{R}_n} w_m = \frac{\partial w_m}{\partial \boldsymbol{R}_n} = \sum_{i,j} \frac{\partial w_m}{\partial \tilde{K}_m^{ij}} \frac{\partial \tilde{K}_m^{ij}}{\partial \boldsymbol{R}_n} \tag{7.10}$$

where i, j identifies an element of the matrix $\tilde{K}_m$. The first terms can be easily computed with TensorFlow [94], while the second must be handled separately and coded explicitly [16, 18]. A more detailed description of the calculation can be found in Section 7.3. Beside energies and forces, the neural-network potential (NNP) predicts also the virial, $\Xi_{\alpha\beta}$, of the system defined as in Eq. (9.3). Using Eq. (7.8) one can write[95]:

$$\Xi_{\alpha\beta} = \sum_n R_{n\alpha} F_{n\beta} = -\sum_{n \neq m} R_{nm\alpha} \frac{\partial E^{\boldsymbol{W}}}{\partial R_{nm\beta}}, \tag{7.11}$$

where the second term can be further be split in two contribution as previously shown for the forces.

The $\boldsymbol{W}$ parameters contained in the networks are optimized minimizing the following family of loss functions using the Adam stochastic gradient descent method [93]:

$$L = p_E \Delta E^2 + \frac{p_f}{3N} \sum_n \Delta \boldsymbol{F}_n^2 \tag{7.12}$$

where ΔE^2 and $\Delta \boldsymbol{F}_n^2$ are the squared deviations of the potential energy and atomic forces, respectively, between the reference DFT model and the NNP predictions. The two prefactors, p_E and p_f, are needed to optimize the training efficiency and to account for the difference in the physical dimensions of energies and forces. The DeepMD-kit [95], also, allows to introduce inside the loss a term that depends on the virial, that, since we didn't used, is not reported in present work.

7.3 Heat trasport with DeepMD

The NNP, based on DP models, give an easy access to local energies, w_n, that directly enter in the equation of the energy current Eq. (3.27). An other key component of Eq. (3.27) is the derivative of the local energy, $\frac{\partial w_m}{\partial \boldsymbol{R}_n}$. It is composed of two terms, i.e., $\frac{\partial w_m}{\partial \tilde{K}_m}$ and $\frac{\partial \tilde{K}_m}{\partial \boldsymbol{R}_n}$. Since w_n is a well defined and easy to compute function of the *local environment matrices* $\tilde{K}_m$ [18], the first term can be easily obtained from TensorFlow [94] using the same back-propagation approach that is commonly used during the training of a NNP [92, 96]. The second term must, instead, be computed explicitly [16, 18]. Given the definition in Eq. (7.5) and the smoothing function in Eq. (7.6), applying the chain rule we get:

$$\frac{\partial \tilde{K}_m}{\partial R_n^\alpha} = \frac{\partial \tilde{K}_m}{\partial R_{ql}^\beta} \frac{\partial R_{ql}^\beta}{\partial R_n^\alpha} \tag{7.13}$$

where sums on repeated indices are implied, and $\alpha, \beta = 1, 2, 3 \equiv x, y, z$ denote Cartesian coordinates. We find:

$$\frac{\partial R_{ql}^\beta}{\partial R_n^\alpha} = \delta_{\beta,\alpha}(\delta_{n,q} - \delta_{n,l}) \tag{7.14}$$

$$\frac{\partial \tilde{K}_m}{\partial R_{ql}^\beta} = \frac{\partial \tilde{K}_m}{\partial R_{qm}^\beta} \delta_{l,m} + \frac{\partial \tilde{K}_m}{\partial R_{ml}^\beta} \delta_{q,m} \tag{7.15}$$

where δ_{nm} is the Kronecker delta.

Using i, j to represent line and column indices of the element of $\tilde{K}_m$ to be

differentiated, a general element of $\left[\frac{\partial \tilde{K}_m}{\partial R^\beta_{qm}}\right]_{ij}$ is non-zero only if atom q is the i-th neighbour of m in the matrix $\tilde{K}_m$:

$$\left[\frac{\partial \tilde{K}_m}{\partial R^\beta_{qm}}\right]_{i,j} = \begin{cases} \frac{R^\beta_{qm}}{R^2_{qm}}\left(\frac{\partial \sigma_{qm}}{\partial R_{qm}} - \frac{\sigma_{qm}}{R_{qm}}\right) & \text{if } j = 1 \\ \frac{\partial \sigma_{qm}}{\partial R_{qm}} \frac{R^\beta_{qm} R^{j-1}_{qm}}{R^3_{qm}} - 2\sigma \frac{R^\beta_{qm} R^{j-1}_{qm}}{R^4_{qm}} & \\ + \delta_{\beta,j-1} \frac{\sigma_{qm}}{R^2_{qm}} & \text{if } j \neq 1 \end{cases} \tag{7.16}$$

where $\sigma_{nm} = \sigma(R_{nm})$. With the same approach a similar expression for $\left[\frac{\partial \tilde{K}_m}{\partial R^\beta_{ml}}\right]_{i,j}$ can be obtained.

An implementation of the heat current with the DeepMD as presented in this chapter (and in [86]), has been released with the the latest versions of DeePMD-kit [95]. This code extends the LAMMPS [97–99] interface of DeePMD-kit allowing the computation of the heat current via the command `compute heat/flux`. For more info see the documentation on DeePMD-kit [100].

In our DNN model only the forces and the total potential energy are fixed by the training, the local distribution of the energy is freely left to the DNN to learn during the training. Different models might give the same energy and forces but different atomic energies. This ill-definition of atomic energy is not surprising and arises in all classical force field and the invariances principles [8, 9, 12] (see Section 3.3) insure us that the thermal transport coefficient is well defined. In the next section, we will better state the problem and we will perform a small numerical experiment to see the effects of having a batch of models trained over the same data but with different initial conditions.

7.4 Gauge invariance and neural-network potential

In a DNN, the loss function is a highly nonconvex function of the weights [101–103]. Its landscape presents many equivalent minima, thus models initialised with different initial conditions might reach different minima.

We trained four different models on the same training data set of DFT-PBE water configurations extracted from AIMD trajectories in the [400K – 600K] temperature range. We label the models as: REF, A, B and C. At the end of the training, we test them on a set of snapshots not present in the original training set and they all reach a root mean square error (RMSE) equal to 0.4 meV/atom. Using the REF-NNP, we performed a 100 ps NVE simulation of 125 molecule of

liquid water at 400 K. Then, for every snapshot in the trajectory, we computed the total potential energy, the local energies and energy fluxes with each of the four models. To estimate the relative accuracy of A-, B- and C-NNP with respect to the REF-NNP that generated the trajectory, we introduce:

$$\Delta_E^i = \max_t \left(\frac{\left| E_t^i - E_t^{\mathrm{REF}} \right|}{N_a} \right), \tag{7.17}$$

where N_a is the number of atoms, t is a snapshot in the trajectory, $i = A$, B, C labels the model and E represents the total potential energy. In our numerical experiment $\Delta_E^i < \mathrm{RMSE} = 0.4$ meV/atom for any i.

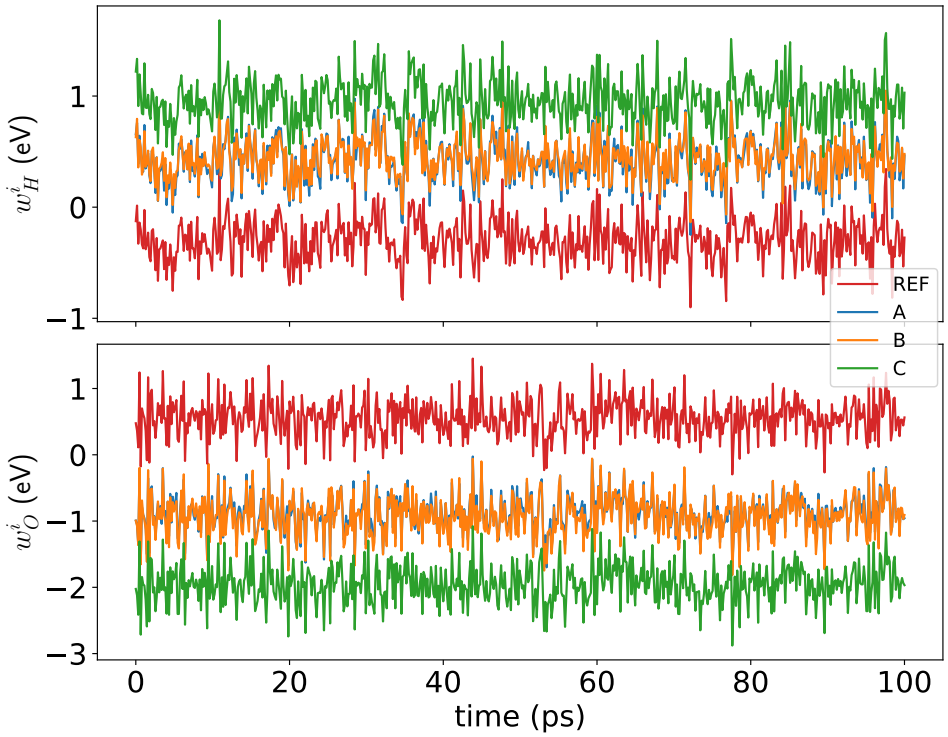


Figure 7.2: The time series of the atomic energy of two atoms, an hydrogen in the upper panel a oxygen in the lower, computed with the different models.

In Fig. 7.2, we can compare the time series of the local energy of two atoms, hydrogen in the upper panel and oxygen in the lower panel. The A- and B-NNP predict similar w, but with the C- and REF-NNP we obtain much larger absolute values. We can conclude that even though the four NNP predict very similar total potential energies, the local decomposition in atomic energies is different.

In Fig. 7.3, we show the power spectrum obtained with each model. Since the

atomic decomposition of the energy is not the same all the spectra are different for $\omega = 0$, but thanks to the invariance principles, the thermal transport coefficient is the same. In the inset the dashed lines represent the ceptral filters, that for all the models converge to consistent values of κ at $\omega = 0$: $\kappa^{REF} = 1.04 \pm 0.07$ W/(mK) , $\kappa^{A} = 1.05 \pm 0.07$ W/(mK), $\kappa^{B} = 1.06 \pm 0.07$ W/(mK), $\kappa^{C} = 1.05 \pm 0.07$ W/(mK).

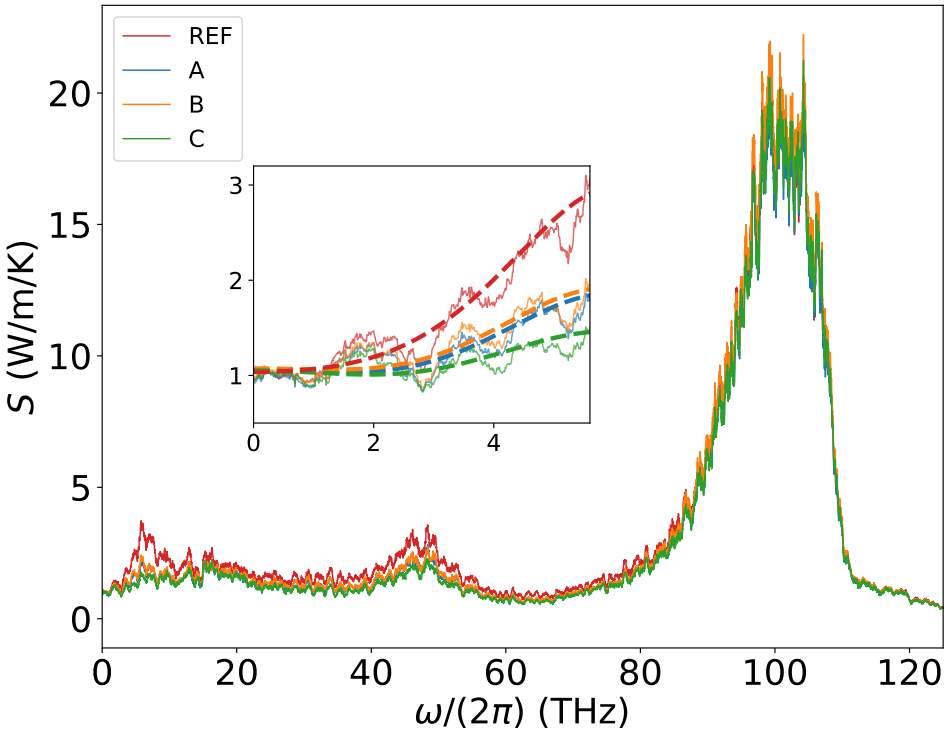


Figure 7.3: The (window-filtered) power spectra of the energy flux computed with the different models. In the inset the dashed lines represent the cepstral filter computed with the different models, that even if at $\omega > 0$ are not the same they all collapse at zero to the same transport coefficient.

8

Thermal transport in water via neural-network molecular dynamics

In this chapter we present the results of the computation of thermal transport coefficient from DPMD simulations. Section 8.2 contains the benchmark of our DNN methodology against the results from AIMD at PBE level. Having proved that DPMD trustfully reproduces *ab initio* results, in Section 8.3, we take advantage of the simple DNN expression for the heat current to compute the thermal transport coefficients of liquid water at the SCAN meta-GGA level of theory. All the results have been reported in Ref. [86]

8.1 Training parameters and test

We trained a DP model on a set of DFT-PBE data extracted from Car-Parrinello trajectories at different temperatures in the [400K – 1000K] temperature range. In the definition of the *local environment matrices*, the two radii inside the smoothing function in Eq. (7.6) are $R_{c1} = 3.50$ Å and $R_c = 7.00$ Å. The embedding network has three layers with 25, 50 and 100 neurons respectively, whereas the fitting network has three layers with 240 neurons each. The loss function is optimized using the Adam stochastic gradient descent method [93], with a learning rate starting at 0.005 and exponentially decaying, with a decay rate of 0.98, every 10^5 training step for a total of $1.5 \cdot 10^6$ training steps. In order to optimize training the coefficients p_E and p_f in Eq. (7.12) were adjusted, respectively, from 0.05 to 1, and from 1000 to 1, during training.

The PBE neural network was tested against a set of $N_v = 800$ independent snapshots of 125 molecules of water at temperatures in the range [400 K – 1000K], obtaining a root-mean-square error of the forces of 0.05 eV/Å. Fig. 8.1 shows a direct comparison between the α component of the *ab initio* force for the s-th atoms in the b-th snapshot and the corresponding NN prediction. The red dashed line correspond to $F^{\mathrm{NN}} = F^{\mathrm{DFT}}$, that fits the data with a *coefficient of determination*

$R^2 = 0.998$. R^2 is computed with the usual formula for linear regression [104]:

$$R^2 = 1 - \frac{\sum_i (F_i^{\mathrm{DFT}} - F_i^{NN})^2}{\sum_i (F_i^{\mathrm{DFT}} - \bar{F}^{DFT})^2}, \tag{8.1}$$

where $\bar{F}^{DFT}$ is the average of all the force components in the dataset.

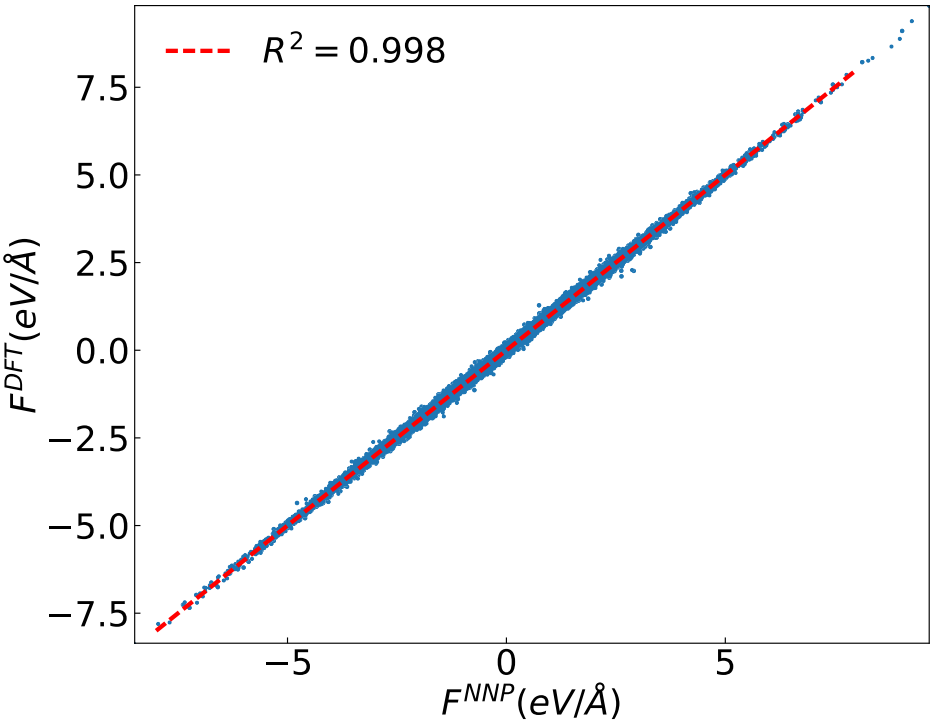


Figure 8.1: Direct comparison between the *ab initio* force components and the corresponding NN prediction. The indexes b, s, α (see main text) label, respectively, the snapshot, the atom, and the Cartesian coordinate of the force. The red dashed line represent $F^{\mathrm{DFT}}_{b,s,\alpha} = F^{\mathrm{NN}}_{b,s,\alpha}$, that fits the data with $R^2 = 0.998$.

Radial distribution function

To estimate the quality of the trained DP model we compared some simple static properties of the model with their *ab initio* counterparts. We ran DPMD simulations of water at the same thermodynamic conditions of the *ab initio* simulations reported in Chapter 6. Fig. 8.2 compares the oxygen radial distribution functions, $g(r)$, from DP and *ab initio* simulations of liquid water (at ≈ 430 K), and of ice-Ih (at ≈ 260 K). Both structures are well described by the DP model. This is true also for the ice-structure even though no ice-snapshots were included in the training data set, in agreement to recent discovery of the building blocks of ice structure inside liquid water [105, 106].

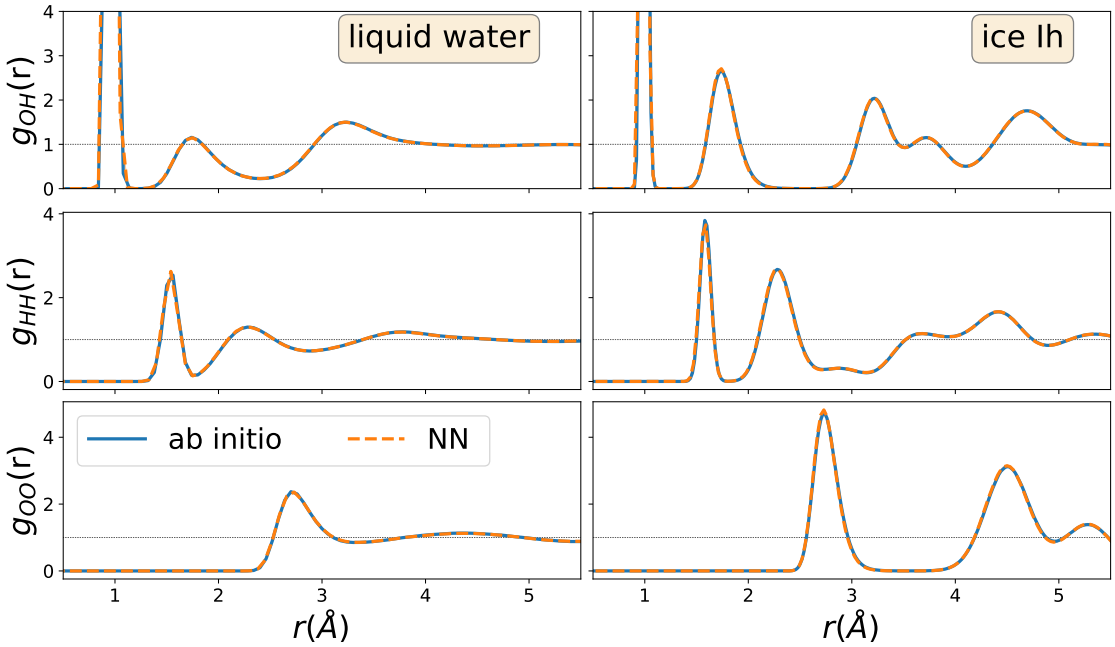


Figure 8.2: Comparison of the radial distribution functions of liquid water, ≈ 423 K, (left column) and ice Ih, ≈ 260 K, (right column) from *ab initio* (continuous blue line) and DP (dashed orange line) simulations, respectively. More details on the simulations can be found in the main text.

8.2 DPMD benchmark *vs.* GGA results: thermal conductivity

The energy flux can be computed using the standard equation for classical force field, Eq. (3.27), combined with Eq. (7.10) for the derivative of the local energies. One of the resulting energy-flux power spectra is displayed in Fig. 8.4 (orange), together with the corresponding *ab initio* spectrum (blue). The thermal conductivities corresponding to the two spectra are obtained as before through cepstral analysis. Notice that, in spite of the much larger weight of the *ab initio* spectrum relative to that of the DNN model, the two spectra have the same low-frequency limit, indicating that the two simulations predict the same conductivity within statistical errors. The difference between the two spectra stems much more from the different local representations of the potential energy than from a different dynamics. The latter is, in fact, very well mimicked by the DNN potential, which gives forces in close agreement with those of the *ab initio* model (see Section 8.1).

In Table 8.1 we display the thermal conductivities computed from *ab initio* MD and DPMD for all the simulations that we performed, together with the atomic diffusivities, D_H and D_O. The latter are computed from the $\omega = 0$ value of the

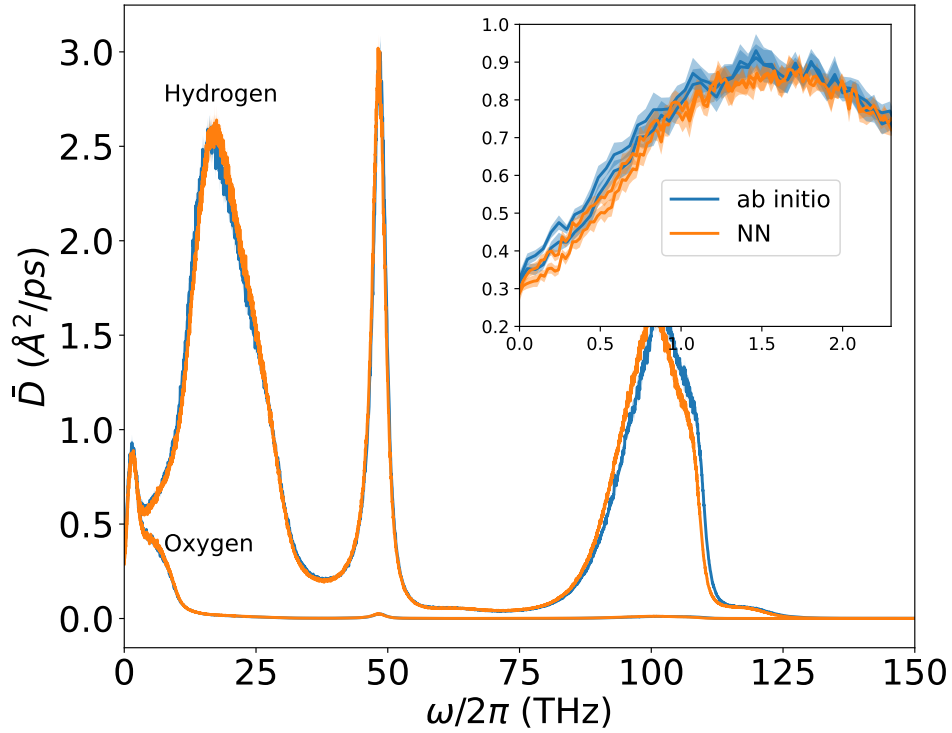


Figure 8.3: Comparison of the oxygen and hydrogen velocity power spectra of liquid water from *ab initio* (blue line) and DP (orange line) simulations, respectively. The simulations used the same periodic cubic cell with density $\rho = 1.00$ g/cm^3 containing 125 water molecules, at $\approx 410K$. The inset shows the region near $\omega = 0$ used to estimate the diffusivity.

power spectrum of the velocity:

$$\bar{D}_\alpha(\omega) = \frac{1}{6N_\alpha} \sum_n^{N_\alpha} \int_{-\infty}^{\infty} \langle \boldsymbol{V}_n(0) \cdot \boldsymbol{V}_n(t) \rangle \; e^{i\omega t} dt \tag{8.2}$$

where α represents the atomic species (oxygen and hydrogen here) and n runs over all the atoms of species α. The diffusivities are obtained from a block analysis of a 100 ps long trajectory and Fig. 8.3 shows the power spectra of liquid water systems at 410 K. The DP model was capable of reproducing accurately the three transport coefficients. In particular, it allowed us to perform longer simulations in order to reduce the statistical uncertainty on κ. While $\approx$ 100 ps long trajectories suffice for errors of about 10% in liquid water and of about 20% in ice Ih, we found that $\approx$ 360 ps long trajectories with the DP model reduced these errors to 5% and 8%, respectively. These errors could be reduced even further because trajectories lasting tens of ns or more would be possible with DPMD. All the thermal conductivities, and corresponding statistical uncertainties, have been computed via the cepstral

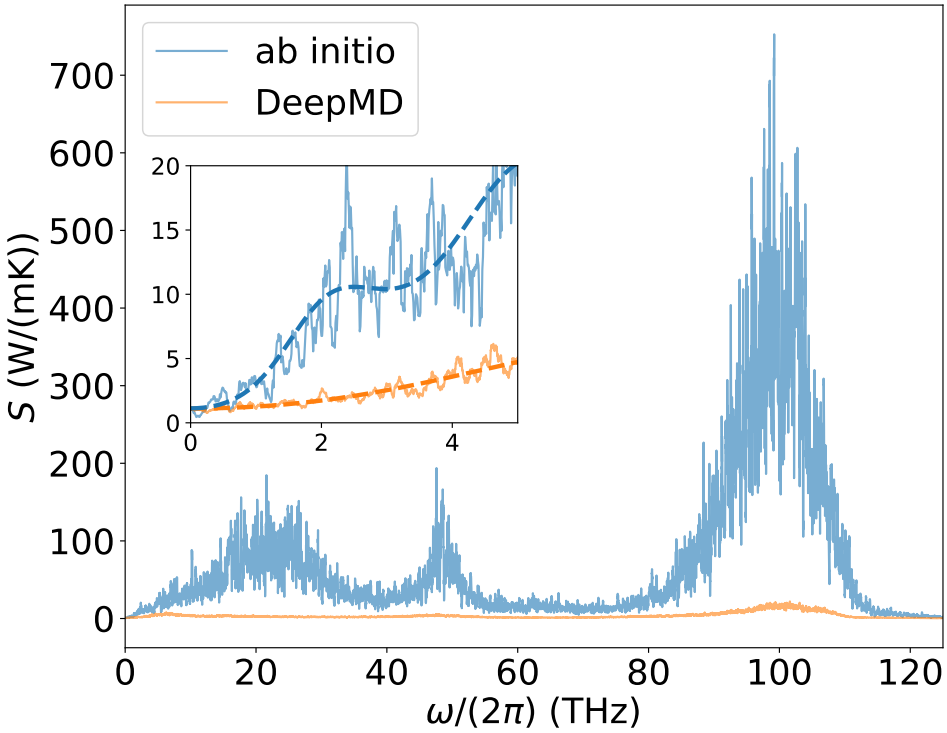


Figure 8.4: Power spectrum of a water simulation. The orange line is obtained from 360 ps of DPMD simulation of a periodic cubic cell containing 125 water molecules at 407 K. The blue line is obtained from an *ab initio* MD simulation of 125 water molecules with the same cubic box and an average temperature of 409 K. Both spectra are filtered with a moving average of 0.1 THz. The dashed lines in the inset represent the cepstral-filtered spectra. Even though the two spectra have very different intensities the values at zero frequency are the same.

analysis using the `SporTran` code [68]. The last two columns of Table 8.1 contains the values of the effective Nyquist frequency, f^*, used to limit the analysis to a properly defined low-frequency window, and the number P^* of cepstral coefficients.

The calculated heat conductivities with DPMD and *ab initio* MD, based on PBE-DFT, agree closely among them, but differ substantially from experiment ($\kappa_{expt} \approx 0.6$ W/(mK) *vs.* $\kappa_{PBE} \approx 1$ W/(mK) for water at near ambient conditions [90]), indicating that the distribution of the energy density resulting from the PBE functional adopted here is likely inadequate to accurately describe adiabatic energy transport in water. This prompted us to try more advanced functional approximations, like the meta-GGA SCAN framework, to cope with this shortcoming.

	phase	T	D_H	D_O	κ	f^*	P^*
		K	Å^2/ps	Å^2/ps	W/(mK)	THz	
DPMD	liquid	516	1.07 ± 0.05	1.08 ± 0.05	0.99 ± 0.05	9.9	11
	liquid	423	0.41 ± 0.02	0.42 ± 0.02	1.03 ± 0.05	17.8	12
	liquid	408	0.29 ± 0.02	0.32 ± 0.02	1.11 ± 0.05	36.7	17
	ice Ih	270	-	-	1.9 ± 0.2	25	93
ab initio	liquid	521	1.13 ± 0.05	1.11 ± 0.05	0.98 ± 0.19	20.7	55
	liquid	431	0.45 ± 0.03	0.45 ± 0.03	1.06 ± 0.11	20.1	17
	liquid	409	0.325 ± 0.018	0.29 ± 0.02	1.12 ± 0.17	45.9	33
	ice Ih	260	-	-	1.8 ± 0.4	30.3	53

Table 8.1: Comparison of some properties of water from *ab initio* MD and DPMD simulations based on PBE-DFT. All liquid simulations used 125 H_2O molecules inside a cubic box of side $l = 15.52$ Å. The ice Ih simulations used 128 H_2O molecules inside an orthogonal cell with sides: $l_1 = 18.084$ Å, $l_2 = 15.664$ Å and $l_3 = 14.724$ Å. T is the mean temperature of the simulations; D_H and D_O are the diffusivities of hydrogen and oxygen, respectively; while κ is the thermal transport coefficient. The diffusivities of ice Ih are compatible with zero and are not reported. The lasts two columns contain the values of f^* and P^* used for the *ceptral analysis*.

8.3 Extended simulations with a SCAN based deep potential model

Meta-GGA functionals like SCAN depend on the electronic kinetic energy density, in addition to the density and its gradient, making significantly more complicated than in the PBE case the derivation of an analytic expression for the energy flux to use in *ab initio* MD studies of heat transport. However, this is not necessary, as the DPMD methodology not only gives us a framework for molecular simulations having quantum-mechanical accuracy at a cost close to that of empirical force fields, but also offers us the capability of easily deriving a practical expression for the energy flux, in situations where it would be difficult to obtain it directly from first principles. To follow this route, we trained a DP model using the SCAN-DFT dataset of Ref. [108]. The thermal conductivity predicted by this model, at $T \approx 430$ K and at the same density used in our previous PBE simulations, is $\kappa = 0.88 \pm 0.05$W/(mK), which is closer to experiment, but still not in perfect agreement with it. Recent studies [19, 109] found that the melting temperature of SCAN-DP ice Ih models is around 310 K, a value very close to the corresponding DFT temperature, according to perturbative estimates [109]. While still not perfect, this result is far superior to PBE, whose estimated ice Ih melting temperature should be around

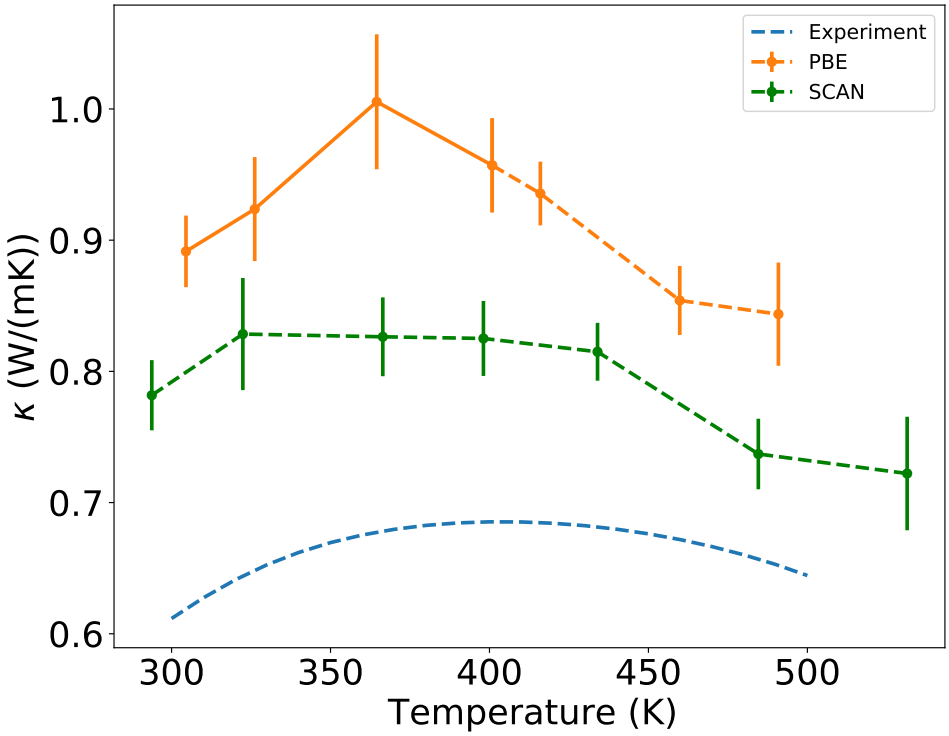


Figure 8.5: Temperature dependence of the thermal conductivity κ of water between 300 K and 500 K. The blue line represents the experimental data from the NIST website [107]. The orange and green lines result from (classical) DPMD simulations trained on PBE and SCAN data, respectively. The simulations use a periodically repeated cubic box with 128 water molecules. In the simulations the box size is fixed to the experimental density [107] at each given temperature. Relative to PBE, SCAN overestimates less the experimental values, and varies less with temperature, consistent with experiment. PBE exhibits a relatively sharp conductivity maximum at around 360 K, whereas experiment shows a broad maximum at $\approx$ 400 K. The sharp PBE maximum may be an artifact of imperfect equilibration in a metastable liquid. The continuous line connects data points below the freezing temperature at $\approx$ 400 K, where the PBE liquid is metastable.

400 K or higher [88, 89]. Thus, one might argue that the 100 K temperature offset used in our PBE-DFT simulations would be inappropriate here, but the rather broad temperature range displayed in Fig. 8.5 shows that the thermal conductivity of water is rather insensitive to temperature at near ambient pressure.

The simulations reported in Fig. 8.5 have been performed by fixing the size of the simulation-box in order to match the experimental density [107] at each reported temperature. At each temperature, we first performed an NVT simulation lasting for a few dozen ps, in which the system was coupled to a Nosé-Hoover thermostat, followed by a 880 ps long NVE simulation, in order to compute the thermal

transport coefficient. The solid line in Fig. 8.5 connects PBE data at temperatures below 400 K, i.e., below the estimated freezing temperature of this model [88, 89]. At these temperatures PBE water is sluggish and difficult to equilibrate.

SCAN overestimates κ less than PBE, consistent with the better representation of the covalent bond length of the water molecule in the liquid provided by this functional [39]. The experimental data show a broad maximum around 400 K, while PBE exhibits a sharp maximum around 360 K, i.e., below the estimated freezing point of this model. The SCAN results are closer to experiment and are consistent with a broad maximum of the thermal conductivity in the explored region. Whether the residual discrepancy between DFT-SCAN simulations and experiment is due to a residual inaccuracy of the XC functional or to neglect of quantum effects on the nuclear motion is an issue that would require further work to be clarified.

Size scaling for SCAN neural network potential

Size effects may affect the transport properties calculated in numerical simulations [110, 111]. In order to quantify these effects, we run 2 ns long NVE simulations at ≈ 407 K of SCAN-DP water at fixed density and increasingly larger cells (with up to 1000 molecules). The results, reported in Fig. 8.6, suggest that κ shows no size dependence within the error bars of the simulation.

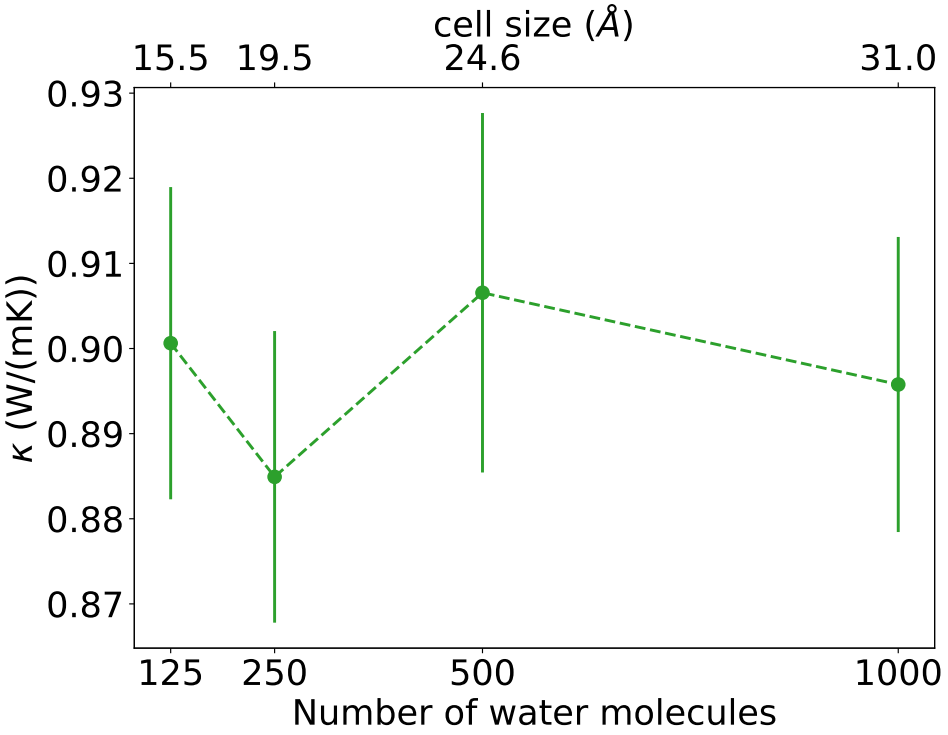


Figure 8.6: The size dependence of the thermal transport coefficient κ for simulation with the SCAN neural network potential. The test shows that no relevant size scale dependence is observed. All the quantities are evaluated from $\approx$ 2ns long trajectories.

9

Viscosity from ab initio *and neural-network simulations*

Shear viscosity is one of the most important transport properties governing the macroscopic flow of liquids. As such, it plays a fundamental role in various fields of science and technology, such as, *e.g.*, chemical and mechanical engineering or earth and planetary sciences, to name but a few. For instance, the viscosity of a solvent crucially affects the dynamics of solutes and the reactions rates, in biological processes and chemical reactions [112–114]. An accurate determination of the temperature and pressure profile of the viscosity is also essential for the correct modelling of tidal interactions in the planets' interior, in particular in the presence of icy layers [115, 116].

In spite of the great importance of water and the large number of studies based on density-functional theory and *ab initio* molecular dynamics (AIMD) devoted to it [34–42], all of these efforts have always dodged its viscous properties, because *an accurate computation of the viscosity of water would require exceedingly long first-principles simulations* [40]. A number of studies based on classical force fields exists [117–120], but the poor transferability of these models sets a limit to their predictive power. An attempt to estimate the viscosity of water from first principles was made with an indirect approach relying on the Stokes-Einstein relation [42], which, however, does not hold over all the phase diagram for liquid water, particularly in the supercooled regime [121–123].

A rigorous microscopic description of the shear viscosity of liquids, η, is provided by the GK theory of linear response, according to which its value is proportional to the integral of the time auto-correlation function (tACF) of the off-diagonal matrix elements of the stress tensor. In the following chapter, we will present the results of the viscosity of liquid water from Ref. [124].

9.1 Green-Kubo simulation of viscosity in liquids

In Chapter 2 we went through the GK theory of linear response [1, 2] obtaining general equations to relate the transport coefficient, in the form of Onsager's coefficients, to the correlation of corresponding current, Eqs. (2.28) and (2.30). In the

case of the viscosity, the transported quantity is the momentum and the "current" components are the off-diagonal element of the stress tensor.

$$\eta = \frac{L^{\sigma\sigma}}{T} = \frac{\Omega}{k_B T} \int_0^\infty \langle\, \sigma_s\left(\Gamma_t\right) \sigma_s\left(\Gamma_0\right) \rangle\, dt, \tag{9.1}$$

where, with the same notation of Chapter 2, Ω is the volume of the system, T its temperature, k_B the Boltzmann's constant, σ_s is any of the three independent off-diagonal elements of the stress tensor, ($\sigma_s \in \{\sigma_{xy}, \sigma_{xz}, \sigma_{yz}\}$), and Γ_t indicates the time evolution of a point in phase diagram from the initial condition Γ_0. In practice, the value of the integral in Eq. (9.1) is averaged over the three pairs of Cartesian indices. $L^{\sigma\sigma}$ is the Onsager's coefficients, as defined in Eq. (2.30), where the stress is the only relevant current.

Expression of stress tensor

The thermodynamic stress tensor is the equilibrium average, $\bar{\sigma}$, of a microscopic estimator, σ, defined as:

$$\sigma_{\alpha\beta} = \frac{1}{\Omega}\Big[\sum_{i=1}^{N} \frac{p_{i\alpha} p_{i\beta}}{m_i} + \Xi_{\alpha\beta}\Big], \tag{9.2}$$

where α, β represent Cartesian coordinates, $p_{i\alpha}$ is the α component of the momentum of the i-th atom, m_i its mass, while Ξ is the virial term, defined as the derivative of the system's potential energy, E, with respect to an uniform scale transformation of the system ($r_\alpha \to r_\alpha + \sum_\beta \epsilon_{\alpha\beta} r_\beta$, ϵ being the strain tensor):

$$\Xi_{\alpha\beta} = -\frac{1}{\Omega} \frac{\partial E}{\partial \epsilon_{\alpha\beta}}. \tag{9.3}$$

The expression of the virial term depends on the approach one adopts to perform the simulations: explicit formulas in the classical case are given, *e.g.*, in Refs. [125], for pair-wise potentials, and [59], for general many-body potentials, while the quantum-mechanical case is thoroughly covered within DFT in Refs. [126, 127]. The expression of the virial stress using *Deep Potential* models has been given in Eq. (7.11), and it is here reported for clarity:

$$\Xi_{\alpha\beta} = \sum_n R_{n\alpha} F_{n\beta} = -\sum_{n \neq m} R_{nm\alpha} \frac{\partial E^{\boldsymbol{W}}}{\partial R_{nm\beta}}, \tag{9.4}$$

Data Analysis

Even for the viscosity, we can apply the same data analysis techniques illustrated in Chapter 4, defining the *power spectrum* $S^\sigma(\omega)$ of the stress time series according to the Wiener-Khintchine theorem [61, 62] as:

$$S^\sigma(\omega) = \int_{-\infty}^{+\infty} \langle \sigma_s(t)\, \sigma_s(0) \rangle e^{i\omega t} dt. \tag{9.5}$$

Combining Eq. (9.5) with Eq. (9.1):

$$\eta = \frac{\Omega}{2k_B T} S^\sigma(\omega = 0). \tag{9.6}$$

$S^\sigma(\omega = 0)$ can be estimated with the *cepstral analysis*, using `SporTran` [68].

9.2 *Ab initio* Molecular Dynamics

We performed AIMD simulations of liquid water at near-ambient conditions using the PBE [44] XC functional, the plane-wave pseudopotential method, Hamann-Schluter-Chiang-Vanderbilt norm-conserving pseudopotentials [87], and a kinetic-energy cutoff of 85 Ry. The simulated system was made of 64 molecules at the standard density of 1 gr/cm^3, corresponding to a cubic box of edge $l = 12.43$ Å. All the simulations were carried out with the Car-Parrinello extended-Langrangian method [7] using the `cp.x` component of the QUANTUM ESPRESSO distribution [73–75] and setting the fictitious electronic mass to 25 physical masses and the timestep to $dt = 0.073$ fs. We performed two simulations aimed at ambient temperature and somewhat above. As PBE is known to enhance the short-range structure of water and to overestimate the melting temperature by ≈ 140 K [88, 89], we set the target temperatures of the two simulations to 450 and 600 K. Both trajectories where first equilibrated in the NVT ensemble using a Nosé-Hoover thermostat [128] at the target temperature, followed by long production NVE runs of 400 ps. Finally, the shear viscosity is obtained from the cepstral analysis of the power spectrum of the off-diagonal elements of the stress, using the `SporTran` [68] code.

In Fig. 9.1 we display the (moving averages [129] of the) power spectra of the stress-tensor time series resulting from our two simulations. While showing similar features at high frequency, the two spectra differ substantially approaching $\omega = 0$. In particular, lower temperatures see the appearance of sharp peaks near $\omega = 0$, which requires a greater care in the cepstral analysis of the data, which is based on a low-pass filter of the (logarithm of) the power spectra. In the inset we

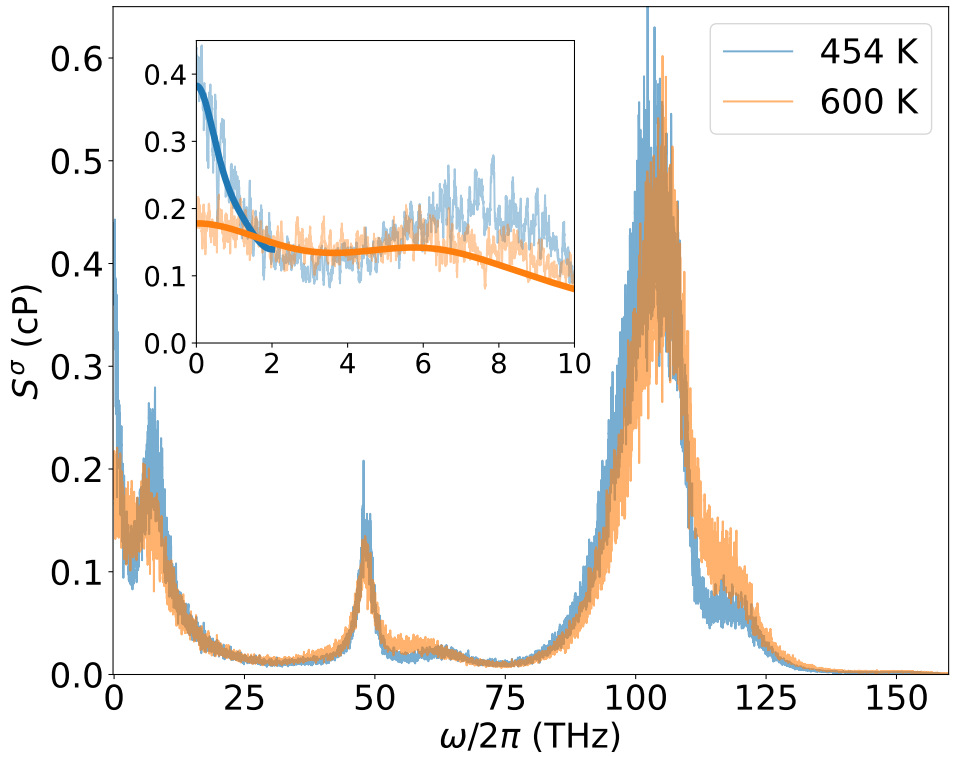


Figure 9.1: Power spectra of the off-diagonal elements of the stress in water at 454 K (blue) and 600 K (orange), obtained from AIMD simulations (see text). The spectra are filtered by a moving average with a window of 0.05 THz. The thick solid lines in the inset represent the cepstral-filtered spectra whose zero-frequency value gives an estimate of the shear viscosity.

display the low-frequency region of the spectra together with the results carried out by the cepstral analysis, *i.e.* by applying a low-pass filter to the logarithm of the raw spectra. The filtered spectra are represented by thick solid lines whose zero-frequency value is a fair and accurate estimate the shear viscosity we are after:

$$\eta = \begin{cases} 0.383 \pm 0.023 \text{ cP} & \text{at 454 K,} \\ 0.178 \pm 0.005 \text{ cP} & \text{at 600 K.} \end{cases} \qquad \text{(PBE)}$$

It is often assumed that the predictions of *ab initio* simulations should be compared with experiment upon shifting the simulated temperature by the offset between the theoretical and experimental melting temperatures, which, in the case of PBE, amounts to $T_m(\text{PBE}) - T_m(\text{expt}) \approx 140$ K [89]. We thus compare our value predicted by PBE at $T = 454$ K with the experimental value measured at $T = 313 \approx 454 - 140$ K, $\eta^{expt}(T = 313 \text{ K}) = 0.653$ cP. The agreement is fair, on account of both the uncertainties related to the empirical temperature shift and

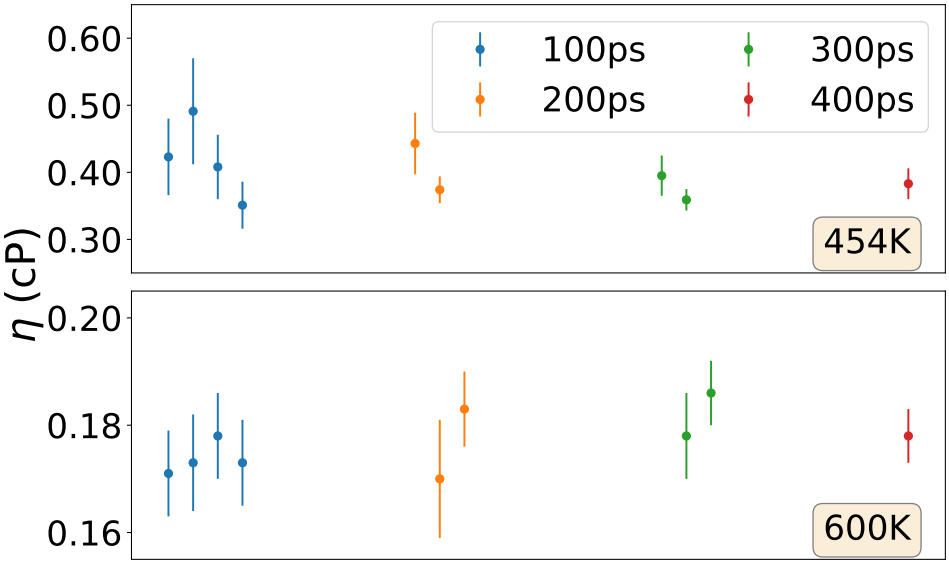


Figure 9.2: Dependency of the shear viscosity η on the length of the simulation, estimated by AIMD (a) at 454 K and (b) 600 K . Different colors refer to different simulation times.

the very sensitive dependence of the viscosity upon temperature near melting. For a more in depth study of the meaning of the residual disagreement see Section 9.5.

In Fig. 9.2 we display how the prediction of the shear viscosity in water depends on the length of the simulation. In order to highlight the impact of possibly long relaxation times on the estimate of the transport coefficient, we have split our 400-ps trajectories into segments of 100, 200, and 300-ps (in the latter case the two segments were overlapping). If the stress correlation time were much shorter that the length of the segments, the estimates from different segments would coincide within the statistical errors evaluated within each of them. By and large, this is so at 600 K, but not quite so at 454 K. This indicates that, as the freezing temperature is approached, the stress correlation time increases and make it necessary to rely on correspondingly longer trajectories. All these considerations suggest that near freezing the computation of the shear viscosity requires long simulation runs, and longer runs would be required for a fair evaluation of the statistical uncertainties, indicating that AIMD may not be the most efficient approach to explore a broad range of thermodynamic conditions. In the following we show that neural-network models of inter-atomic interactions trained on *ab initio* data provide a valid alternative to direct AIMD simulations, yielding results of similar quality at a much lower computational cost.

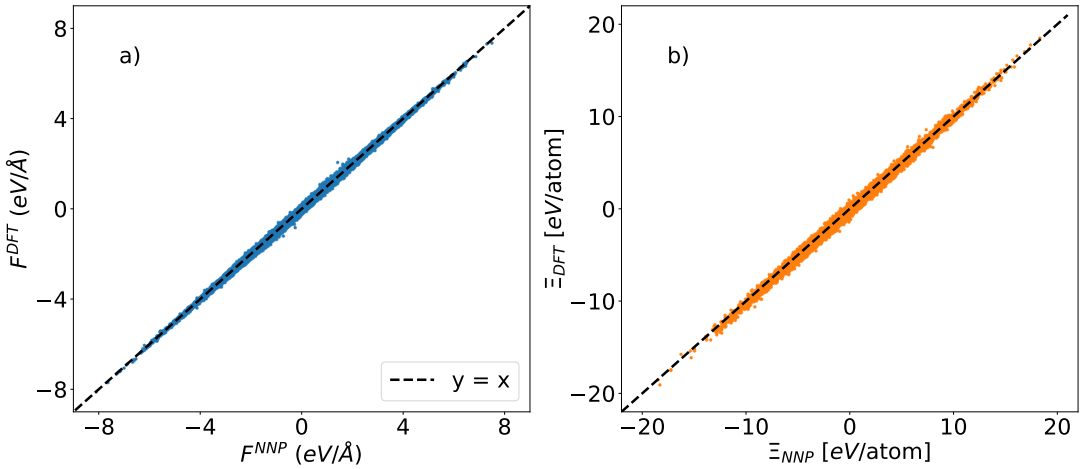


Figure 9.3: Scatter plot of the NNP forces (a) and off-diagonal elements of the virial per atom (b) *vs.* DFT data, for a dataset of 10000 configurations. The corresponding correlation coefficients are $R_a^2 = 0.998$ and $R_b^2 = 0.995$.

9.3 PBE NNP

In order to appraise the ability of NNP to accurately predict shear viscosity, we have generated one such model, by training it on a set of DFT-PBE data. The training dataset is prepared via a recently proposed "on-the-fly" learning procedure called Deep Potential Generator (DP-GEN) [103, 130] and it consists of the energies and atomic forces of 4000 configurations of water generated by the DP-GEN from NPT MD trajectories at different temperatures in the [300-700 K] range and for pressures up to 50 kbar. The PBE-NNP is then constructed and trained with the DeePMD-kit. The cutoff radius is set to 6 Å. The size of the embedding and fitting nets is (50, 50, 50) and (250, 250, 250), respectively. The model was trained by minimizing the loss function in Eq. (7.12) with 2 million steps of Adam stochastic gradient descent [93]. We tried to include the values of the virial in the definition of the loss function, but we found no improvement with respect to the standard definition of Eq. (7.12), and thus decided not to modify it.

Fig. 9.3 shows a scatter plot of the NNP predictions for atomic forces and stress *vs.* PBE-DFT data, evaluated over a set of 10000 configurations, not included in the training dataset. The average error on the forces and on the off-diagonal elements of the virial are $\sigma_F = 40$ meV/Å and $\sigma_\Xi = 1.4$ meV/atom, respectively, corresponding to correlation coefficients [104] of 0.998 and 0.995, respectively.

In order to validate our neural-network methodology for the prediction of the shear viscosity, we performed DPMD simulations in the *NVE* ensemble for the

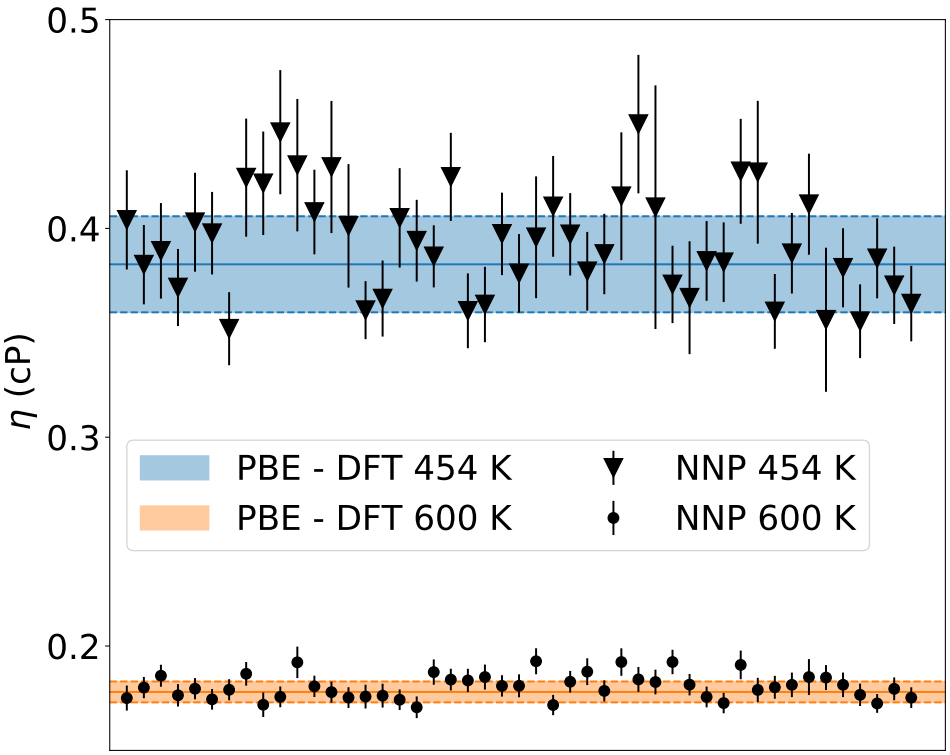


Figure 9.4: Comparison between the shear viscosity predicted by 400-ps long DFT AIMD simulations (horizontal blue and orange bands, 454 and 600 K, respectively) and by *ab initio* DPMD simulations of the same length (black symbols).

same model of liquid water described above. Simulations of 20-ns were run at two different temperatures, 454 K and 600 K, using our NNP trained to PBE water. All simulations were carried out using the LAMMPS code [98] interfaced with DeepMD-kit. In Figure 9.4 we display the results obtained by analysing independently each one of the 50 400-ps segments in which we have partitioned the whole 20-ns trajectory. The shear viscosity of each segment is obtained again by cepstral analysis using the `SporTran` and it is represented by black triangles and dots, for the 454 K and 600 K trajectories, respectively. Each result is displayed with its estimated statistical error. The blue and orange regions represent respectively the estimate of the shear viscosity given in Section 9.2 from *ab initio* MD simulations at 454 K and 600 K. We observe good compatibility between the two different approaches, concluding that our NNP is capable of predicting correctly the shear viscosity of water at the given pT conditions.

In Table 9.1 we report our results for the viscosity of water computed at two different temperatures with DPMD and NNP trained on PBE-DFT data, obtained

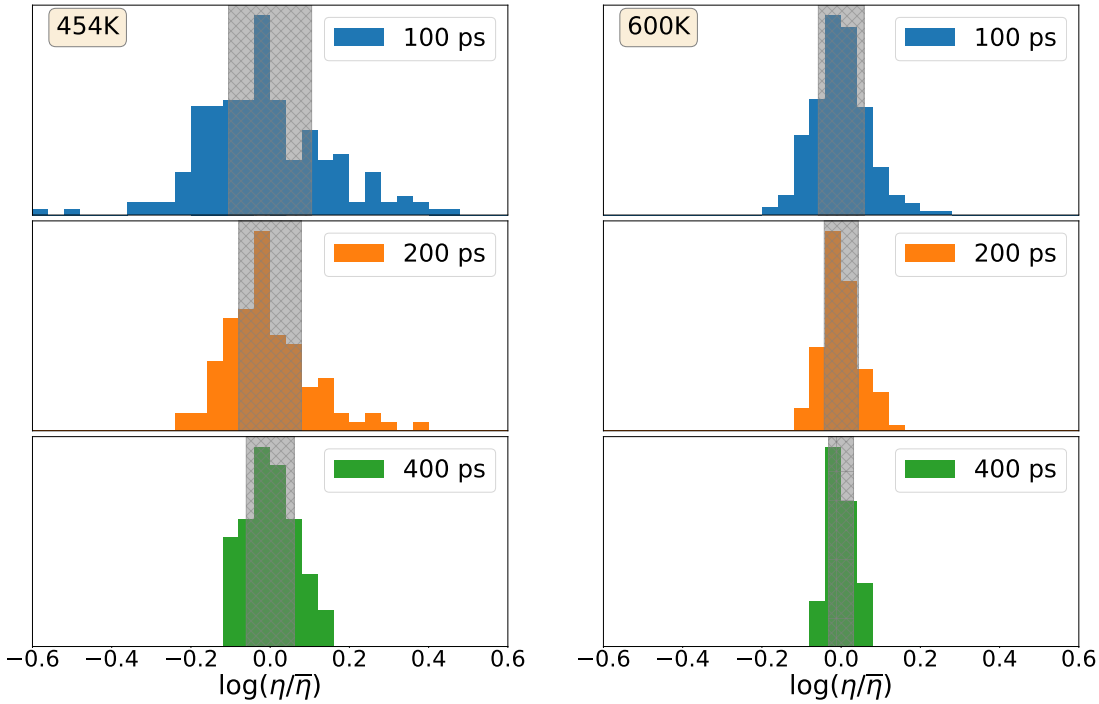


Figure 9.5: Normalized distributions of the logarithm of the shear viscosities, $\log(\eta)$, estimated over multiple MD segments (blue: 100 ps; orange: 200 ps; green: 400 ps) extracted from a 20-ns trajectory at 454 K (left) and 600 K (right). The reported data are referred to the average, $\bar{\eta}$. We remind that the absolute error on $\log(\eta)$ is the relative error on η. The shaded area denote the average standard deviation of the shear viscosities, as estimated by cepstral analysis within each individual segment.

from very long (20-ns) trajectories, and compare them with the AIMD data of Sec. 9.2.

	T = 454 K	T = 600 K
AIMD	0.383 ± 0.023	0.178 ± 0.005
DPMD	0.402 ± 0.005	0.184 ± 0.001

Table 9.1: Viscosity of water [cP] computed from AIMD or DPMD performed at two different temperatures using the PBE XC functional.

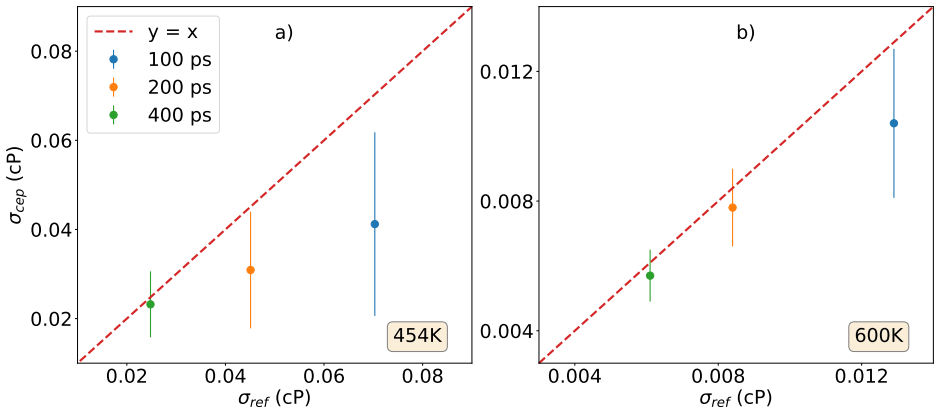


Figure 9.6: Correlation between the cepstral estimates of the standard deviations of the viscosity of water from trajectories of different lengths and temperatures, $\sigma_{\rm cep}$, *vs.* the spread of the distribution of their values resulting from different trajectories, $\sigma_{\rm ref}$ (see text).

9.4 Statistical analysis and finite-size scaling

We are now ready to investigate the statistical behaviour of the shear viscosity for different simulation lengths. To this end, we slice our 20-ns simulations in segments of smaller lengths (100, 200, and 400 ps) and analyze them independently. The distributions of the results are shown in Fig. 9.5 for both temperatures. We observe that: *i)* the width of the distributions of the viscosity estimated at different lengths is larger than the standard deviation estimated within each segment by cepstral analysis; *ii)* this difference decreases as the length of the segments increases, until it roughly vanishes at 400-ps; *iii)* this difference also decreases by increasing the temperature. This observation is made more quantitative in Fig. 9.6, which shows the correlation between the standard deviations of the cepstral estimates of the viscosity from trajectories of different lengths and temperatures, *vs.* the spread of the distribution of their values resulting from different trajectories. The former quantity is itself affected by a statistical uncertainty because cepstral analysis returns different standard deviations for different trajectories of a same length. Fig. 9.6 indicates that as the system approaches freezing from above and the viscosity increases, the low-frequency components of the virial fluctuations become increasingly important, and simulations of increasing length become necessary to cope with them. This is confirmed in Fig. 9.7 that displays the low-frequency portion of the power spectrum of the off-diagonal elements of the stress in water at different temperatures, and shows that as the system approaches freezing from above, a narrow peak develops at $\omega = 0$, as a consequence of the onset of long-lived

relaxation modes. In the present case, it appears that at 450 K trajectories as long as 400-ps are needed to get a reliable estimate of the statistical error affecting the estimate of the PBE-DFT viscosity. More generally, it seems that the flexibility offered by NNP and the long simulations they can afford are instrumental not only in exploring broad regions of the phase diagram of a material, but also in providing a reliable estimate of the statistical accuracy of individual simulations.

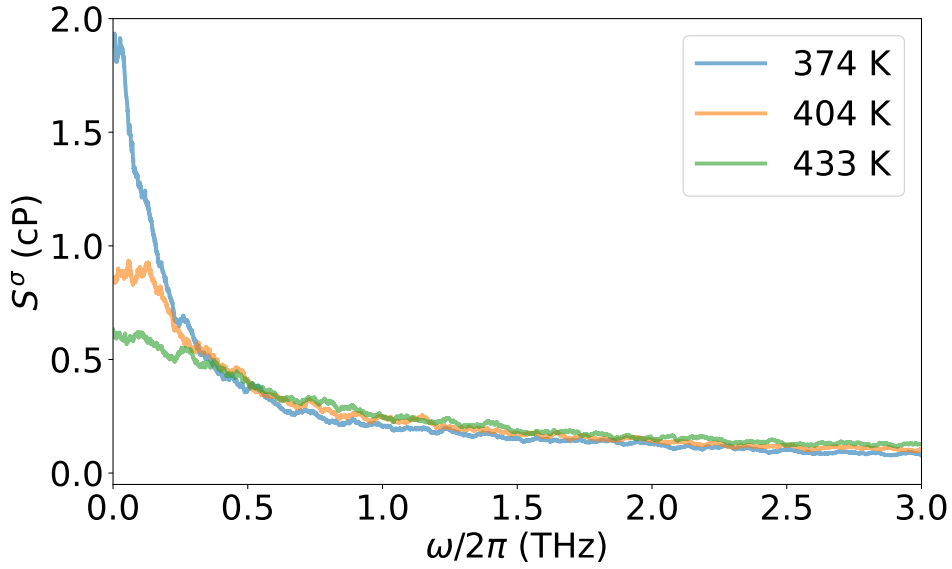


Figure 9.7: Moving average of the low-frequency region of the power spectrum of the off-diagonal elements of the stress tensor in water at different temperatures, as obtained from DPMD simulations trained on PBE DFT data. An averaging window of 0.05 THz was used. Simulations were run at the fixed density of 1 gr/cm^3.

Finite-size effects may affect the transport properties calculated in numerical simulations. In order to quantify these effects in the present case, we run up to 5 ns long NVE simulations at 454 K and 600 K of PBE-NNP water at fixed density and increasingly larger cells (with up to 4096 molecules). The results, reported in Tab. 9.2, indicate that η shows no evident size dependence within the error bars of our simulations.

	size (number of molecules)		
T [K]	64	512	4096
454	0.402 ± 0.005	0.402 ± 0.005	0.417 ± 0.007
600	0.184 ± 0.001	0.186 ± 0.002	0.186 ± 0.002

Table 9.2: Shear viscosity [cP] computed for water at different temperatures with PBE-NNP force fields and using simulation boxes of different size.

9.5 SCAN NNP

The SCAN meta-GGA XC functional has demonstrated the ability to predict well several properties of water over a broad range of thermodynamic conditions, whose exploration was made possible by NNP techniques [19, 23, 39, 40, 109]. In Section 8.3 we already saw how a combination of AIMD and NNP techniques, based on the SCAN XC functional, can successfully predict the heat transport properties of liquid water [86]. In the following section we report an extension to the computation of the shear viscosity.

Accurate DPMD simulations were performed using NNP force fields trained on both PBE and SCAN DFT data [86] and the same software setup as in Sec. 9.3. Our simulated systems consist of 512 water molecules. With systems of this size, temperature fluctuations are smaller than 1K. We first perform NVT simulations at the target temperature, followed by NVE production runs, up to 5-ns long.The volume was fixed to the value corresponding to the equilibrium densities evaluated in Ref. [109] via enhanced-sampling simulations for SCAN, while for PBE it is computed from direct DPMD NPT simulations at ambient pressure, whose results are in agreement with previous calculations [38, 131].

In the left panel of Fig. 9.8 we compare our SCAN-NNP and PBE-NNP results with each other and with experimental data [132, 133]. Results below the melting temperature, T_m, refer to the undercooled fluid, which becomes increasingly viscous as the temperature decreases. Remarkably, when temperatures are referred to the theoretical melting one, the SCAN predictions for the viscosity are in close agreement with experiment at melting (and above, as we will discuss shortly). This is not so for PBE. One could argue that PBE yields too low a viscosity as a consequence of the too low equilibrium density (0.77 *v.s.* ≈ 1 gr/cm^3 at melting). This is not the case, however, because repeating the simulations at the density of 1 gr/cm^3 (dashed lines) results in only a marginal increase in the predicted viscosity. We conclude that the common wisdom according to which the properties of PBE water would match those of real water at a simulation temperature $\gtrsim 100$ K above the experimental one is likely too simplistic: PBE water not only freezes at

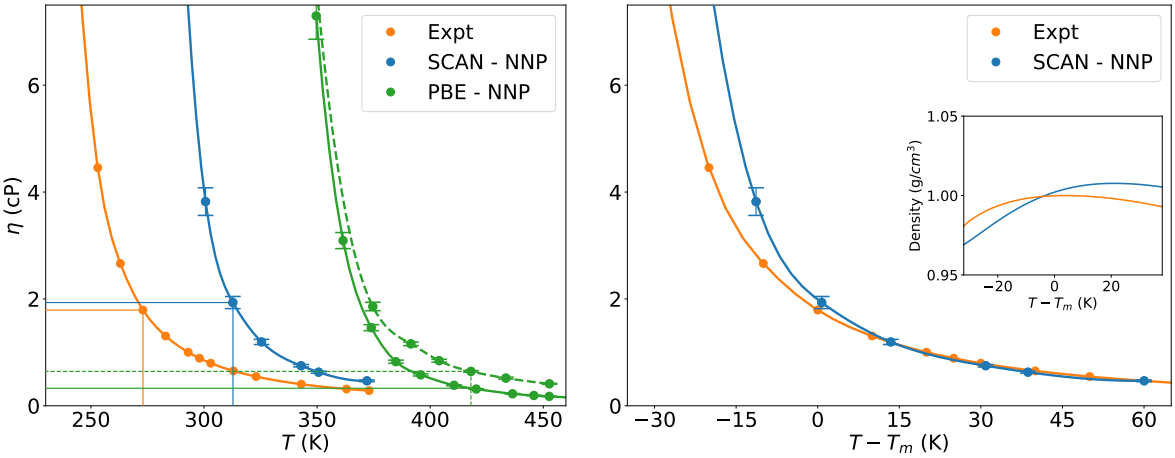


Figure 9.8: Comparison between the shear viscosities of water computed via DPMD simulations using NNP force fields trained to different DFT data sets and with experiments [132, 133]. When not visible, the error bars are smaller than the dots. Continuous lines refer to simulations performed at the equilibrium density corresponding to each temperature. PBE data marked with a dashed line are obtained at the density of 1 gr/cm^3. The thin vertical and horizontal lines mark the melting temperature and the corresponding viscosities. The right panel show only the SCAN predictions *v.s.* the experimental values. The temperature scale for SCAN data has been offset by the difference between the theoretical and experimental melting temperatures, T_m (see text).

too high temperature, but its dynamics is way too fast at melting, as confirmed by the too-high self-diffusivity predicted by PBE, with respect to SCAN and experiment, when all the simulations are performed at the same temperature offset from T_m as experiment. For instance, the self-diffusivity of PBE water computed at a temperature ≈ 20 K higher than T_m(PBE) ≈ 417K [89] ($T \approx 20 + 410 = 430$ K) is 0.45 Å^2/ps [86], to be compared with 0.19 Å^2/ps at $T \approx 20 + 310 = 430$ K for SCAN (T_m(SCAN) ≈ 312K [19, 109]) and a same value for experiment, measured at $T = 20$ C. In a model where the dependence of the self-diffusivity on temperature were Arrhenius-like, this behaviour would be consistent a too small pre-exponential factor as predicted by PBE with respect to SCAN and experiment. Further insight into the dynamics of the water hydrogen-bond network at melting certainly deserves further investigation.

In the right panel of Fig. 9.8 we compare with experiment the SCAN-NNP predictions for the viscosity of water, on a temperature scale that has been offset by the difference between the predicted melting temperature and the observed one, $\Delta T = 312 - 273 = 39$ K. One observes that, while the agreement between theory and experiment is excellent above the melting temperature, SCAN consistently overestimates the viscosity in the undercooled regime. Interestingly, a crossover

between the predicted and observed densities occurs at the melting temperature: SCAN slightly overestimates the density of water for $T > T_m$, while it underestimates it in the undercooled regime. We hypothesize that the too large SCAN predictions for the viscosity below freezing may be related to a propensity of SCAN to overestimate the strength of the hydrogen bonds. In turn, this would lead to overestimate low-density (LD) over high-density (HD) fluctuations upon cooling, corresponding to configurations that underlie the structure of amorphous ices and water. At very deep undercooling they may lead to phase separation between an LD and a HD liquid [23, 134, 135]. The stronger local structure of LD water with respect to HD water seems compatible with a more marked solid-like behavior [136–138] and, hence, with a larger viscosity.

10

Long-Range electrostatic interactions

Water has a rich and complex phase diagram [33]. Inside the ice giants, Uranus and Neptune, it is thought to appear at its superionic (SI) phase, where oxygen ions are arranged in a crystalline lattice and protons diffuse freely like in a fluid [33, 139]. Partially-dissociated, liquid (PDL) water may, instead, be confined to the outermost third of the interior, where the magnetic field is generated [140]. The transport properties of such phases of water are of great importance for any attempt at a quantitative planetary model of rich icy planets. Experimental investigation of transport properties of materials at such conditions is still challenging. Molecular simulations are the only tool to estimate the values of transport coefficients a such extreme conditions. Recently Grasselli et al. [83] computed the electrical of water at inner planets conditions via AIMD simulations. A neural-network potential would allow to extend such studies to a large number of temperatures and pressures. In this chapter we show our attempt to construct a NNP that can predict accurately the electrical conductivity of water at high pT condition.

10.1 Electrical conductivity

Levering the combination of gauge invariance of transport coefficients and topological quantization of adiabatic charge transport [141], the electrical conductivity (γ) can be computed using ions carry an integer charge whose magnitudes equal their formal oxidation numbers ($q_H = +1$ and $q_O = -2$ for water), obtaining the same conductivity obtained from the exact quantum-mechanical expression of the electric current, based on Born's effective charges. In particular, γ can be written as the Green-Kubo integral of the electrical flux, $\boldsymbol{J}^Z$:

$$\gamma = \frac{\Omega}{3k_B T} \int_0^{+\infty} \langle \boldsymbol{J}^Z (\Gamma_t) \cdot \boldsymbol{J}^Z (\Gamma_0) \rangle \, dt, \tag{10.1}$$

$$\boldsymbol{J}^Z = \frac{1}{\Omega} \left(q_H \sum_{n \in H} \boldsymbol{V}_n + q_O \sum_{n \in O} \boldsymbol{V}_n \right), \tag{10.2}$$

where Ω is the volume, k_B the Boltzmann's constant and $\boldsymbol{V}_n$ are the velocities of the atoms. γ can be computed with the data analysis techniques in Chapter 4 [9, 11].

10.2 Deep Potential model: first attempt

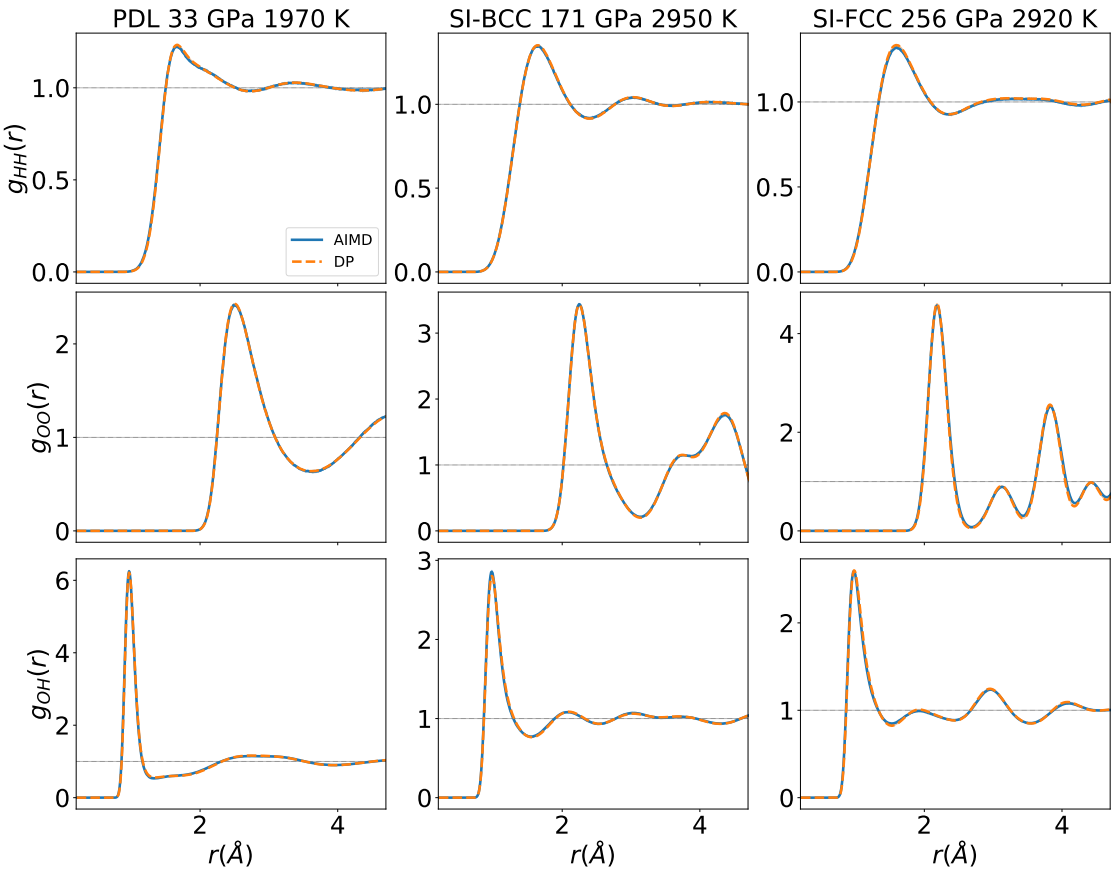


Figure 10.1: Radial distribution functions computed using the DP model (dashed orange line) and AIMD (blue line). Each column correspond to a different system: partially dissociated liquid (left), super ionic water with oxygen in a BCC lattice (center) and in a FCC lattice (right).

We trained a DP model over 3000 snapshots extracted from the simulations in Ref. [83]. For the descriptor in Eq. (7.7) we used a local embedding network with three layers of 25, 50, 100 neurons respectively and the while for cutoff radii of the smoothing function, Eq. (7.6), were respectively: $R_{c1} = 0.50$ Å and $R_c = 4.80$

Å. The fitting network had three layers of 240 neurons each. Once the model is trained, we can test it on a set of new snapshots that were not included in the training set. The obtained root mean square error (RMSE) for energy and forces is respectively: 3.1 meV per atom and 0.30 eV/Å. With such model, we performed DPMD simulations at the same conditions of Ref. [83]. For each system, we first perform a NVT simulation at the target temperature, followed by a NVE production run, up to 800 ps long. We consider the same simulation boxes as in Ref. [83], and we run few simulations for temperatures around the one of AIMD simulations. Figure 10.1 compare the radial distribution functions from DP and *ab initio* simulations. Our model is able to reproduce well the structures of PDL (first column) and SI, with the oxygen arranged in a BCC lattice (second colum) or FCC (third column). Then, we moved to study the electrical conductivity. For each system we performed few DPMD simulations in a range of temperature close to those in Ref. [83].

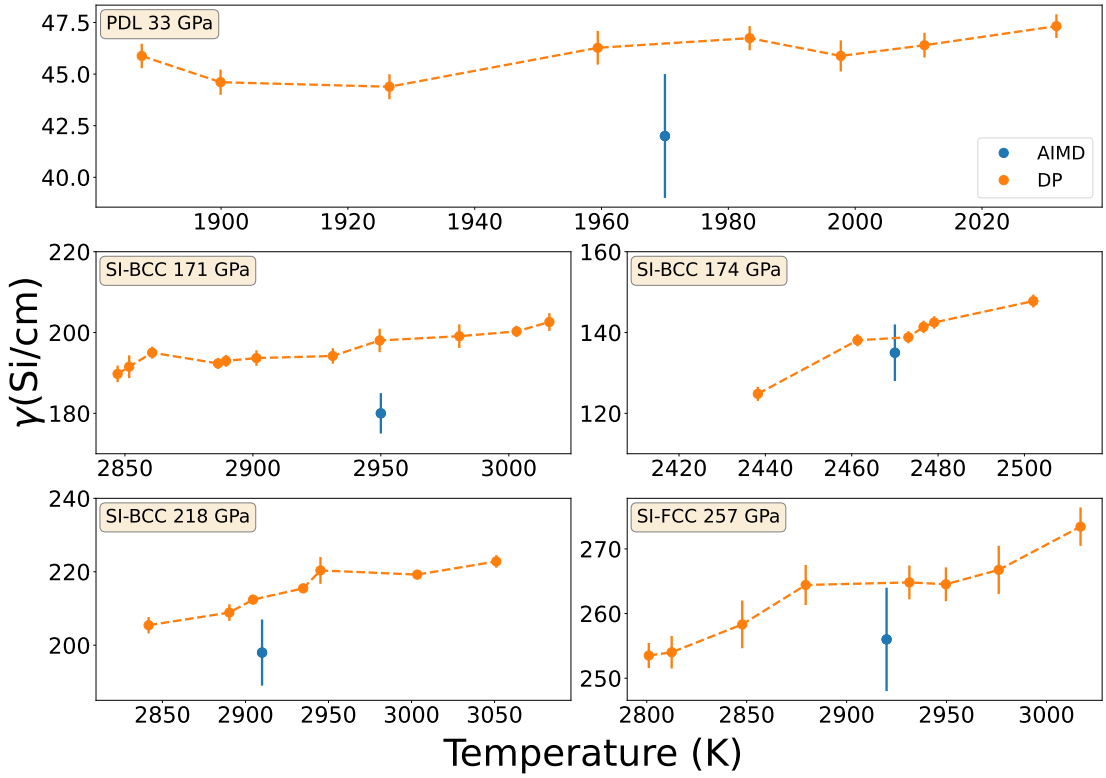


Figure 10.2: Comparison of the electrical conductivity, γ, at high-pT conditions obtained with the AIMD [83] and DPMD. All the simulations are run at the conditions of Ref. [83] with 128 water molecules for SI-BCC and PDL systems, and 108 molecules for SI-FCC system.

Fig. 10.2 shows the comparison between the γ obtained from DPMD (orange)

and AIMD (blue). The DP model fails to obtain compatible results for all the systems. We hypothesised that this discrepancy might be ascribed to long-range charge fluctuations that cannot be described within the cutoff of the DP descriptor.

10.3 Deep Potential Long-Range

L. Zhang et al. [142] have proposed a new approach to expand the DP model to incorporate explicit long-range interaction. In this new model, called deep potential long-range (DPLR), the short-range contribution retains the standard form of the DP model, while the long-range contribution is the electrostatic energy of a system of spherical Gaussian charges centered at the atomic positions and the position of the so-called wannier centroids (WC) [28]. The WCs are defined by average positions of the centers of the maximally localized Wannier functions (MLWF) [143, 144]. In a water system, we can always associate each oxygen to the four closest Wannier centers. The position of the WC is then defined by:

$$\boldsymbol{W}_I = \frac{1}{4}\sum_{i=1}^{4} \boldsymbol{S}_n^i \tag{10.3}$$

where $\boldsymbol{S}_n^i$ is the position of the i-th Wannier center associated to the oxygen n. For each water molecule, there is a one-to-one correspondence to a WC and its closest oxygen at $\boldsymbol{R}_n$. Due to the nearsightedness of the electronic matter [145, 146], the positions of the WCs can be computed from a local environment encoded in a descriptor like D^n in Eq. (7.7) [28]. Zhang et al. [28] developed a neural-network model, called Deep Wannier (DW), that, once properly trained, can predict the positions of the WCs in a water system.

Potential energy surface with DPLR

The total electrostatic energy can be approximated by the electrostatic energy of a system of spherical Gaussian charge distributions centered at $\boldsymbol{R}_n$ and $\boldsymbol{W}_I$ sites, respectively, and can be computed with standard techniques [147–151]:

$$E^G = \frac{1}{2\pi\Omega}\sum_{G\neq 0,|m|\geq L} \frac{exp(-\pi^2 G^2/\beta)}{G^2} f^2(G) \tag{10.4}$$

$$f(G) = \sum_n q_n e^{-2\pi i \boldsymbol{G}\cdot\boldsymbol{R}_n} + \sum_I q_I e^{-2\pi i \boldsymbol{G}\cdot\boldsymbol{W}_I} \tag{10.5}$$

where $f(G)$ is the structure factor, Ω is the volume, β is the spread of the Gaussian distributions and q_n and q_I are the charges associated, respectively, to the atoms

and to the WC. In water $q_n = 6e$ or $1e$ depending if the atom n is a oxygen or hydrogen, respectively, while $q_I = -8e$, with e equal the absolute value of the electron charge.

The total DFT energy, E^{DFT} directly includes the total electrostatic interactions in addition to all the short-range contributions. The DPLR total energy E^{DPLR} is composed of two terms:

$$E^{DPLR} = E^{sr} + E^{G}. \tag{10.6}$$

E^G takes into account the long-range contribution to the energy, while E^{sr} take into account the short range contribution. E^{sr} can be represented by a "short-range" DP model, like Eq. (7.8), with a target energy equal to $E^{DFT} - E^G$.

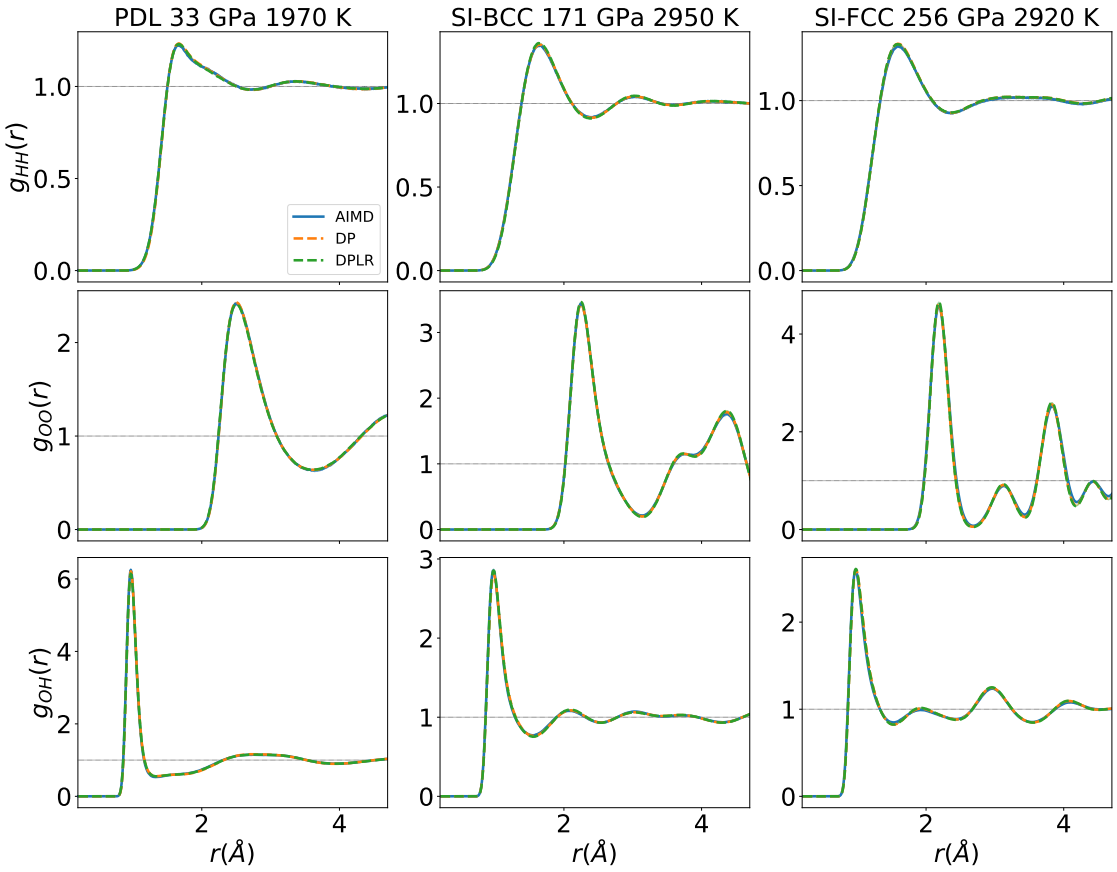


Figure 10.3: Radial distribution functions computed using the DP (orange dashed line), DPLR (green dashed line) models and AIMD (blue line). Each column correspond to a different system: partially dissociated liquid (left), super ionic water with oxygen in a BCC lattice (center) and in a FCC lattice (right).

10.4 Results with long-range interactions

In order to investigate the effects of long-range interactions in the computation of electrical conductivity, we trained a DPLR model. The DW model that computes the positions of the WC has an embedding network with three layers of 25, 50, 100 neurons respectively, while the fitting network has three layers of 100 neurons each. The cutoff radii in Eq. (7.6) are respectively: $R_{c1} = 0.50$ and $R_c = 4.80$. The short-range energy model has a local descriptor with the same parameters as the DW model but the three layers of the fitting network contain 240 neurons each, instead. We trained the DPLR model on the same dataset used for the DP model in Fig. 10.2. For each training snapshot, we computed the positions of the centers of the MLWF with `wannier90` [152–154] combined with QUANTUM ESPRESSO. The MLWF were generated with the selected columns of the density matrix algorithm [155, 156].

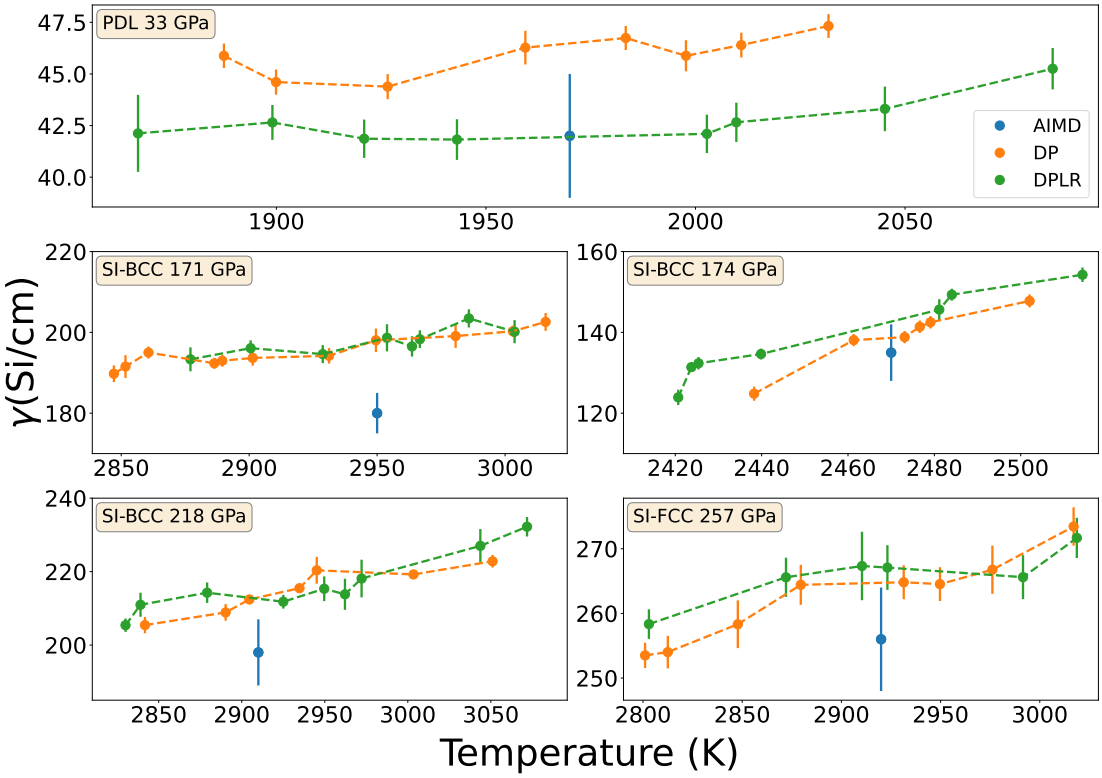


Figure 10.4: Comparison of the electrical conductivity, γ, at high-pT conditions obtained with the AIMD [83], DP and DPLR models. All the simulations are run at the conditions of Ref. [83] with 128 water molecules for SI-BCC and PDL systems, and 108 molecules for SI-FCC system.

We tested the DPLR model on the same test set used in Section 10.2. The new model performed slightly better than the previous one with a RMSE for energy and forces equal to 2.5 meV per atom and 0.24 eV/Å, respectively. In Fig. 10.3 we tested that our DPLR model could predict the structural properties of the water system, comparing the radial distribution function with those from Fig. 10.1.

At last, Fig. 10.4 presents the results for the electrical conductivity computed with the DP, the DPLR model and AIMD [83]. The DP and DPLR model seems to obtain similar results for all the SI systems. Only for the PDL system, the prediction capability of the NNP seems to be enhanced by the introduction of the explicit long-range interactions. We do not have a definitive answer for this behaviour and it requires further study.

11

Conclusion

In this work, we have shown that neural-network potentials (NNP) generated according to the DP framework and properly trained on DFT data are a powerful tool to study the transport properties of water, and likely of other material systems, with quantum-mechanical accuracy at a nearly empirical force field cost. Leveraging the invariance principles of the thermal transport coefficient we show that the values of κ is not influenced by the degree of arbitrariness of the atomic decomposition given by a NNP, which depends on the specific training procedure. An important byproduct of our neural-network approach is that it allows one to derive numerically practical expressions for the energy current, even in cases where analytical derivations from the DFT functional would be hard, as in the case of the SCAN functional. This is quite important since our results show that simpler approximations, like the PBE-GGA, cannot reproduce the transport properties of water accurately. For example, PBE overestimates the thermal conductivity by approximately 60%, while it deeply underestimates the viscosity close to the melting temperature. Both in the case of thermal conductivity and viscosity in water, we have confirmed the ability of the SCAN exchange-correlation density functional to predict a broad array of properties of water over a wide range of thermodynamic conditions. The residual discrepancies might be reduced in the future by training the NNP over more sophisticated functionals, *e.g.* hybrids. Further outlooks of the work might also include the study of the behaviour of the transport coefficients approaching the Widom line [32, 157].

In the last chapter, we showed that the DP model fails to correctly reproduce the transport properties of water at high pressures and temperatures. We tried to overcome these limitations by training a new model with explicit electrostatic long-range interactions. In our first simulations, the DPLR model improves the results only for the partially dissociated liquid and not for the super-ionic water. A definitive answer to this question requires further investigation.

A

Appendix

A.1 Notation for Chapter 5

Notations and definitions used throughout the text in particular Chapter 5 :

- e : electron charge ;
- e_s^0 : ionic energy, equal to $\frac{1}{2}M_s v_s^2 + w_s$, where w_s is the electrostatic energy.
- ϵ_{XC} : local XC energy per particle, defined by the relation: $E_{XC} = \int \epsilon_{XC}[n](\boldsymbol{r})n(\boldsymbol{r})d\boldsymbol{r}$. "LDA" and "GGA" in Eq. (5.15) indicate the local-density and generalized-gradient approximations to the XC energy functional ;
- ϵ_v : electronic eigenvalues ;
- $\tilde{f}(\boldsymbol{G})$: Fourier transform of periodic functions. Given a function $f(\boldsymbol{r})$ periodic with respect to the unit cell, the Fourier transform $\tilde{f}(\boldsymbol{G})$, evaluated at the reciprocal lattice vector $\boldsymbol{G}$ is defined by the convention:

$$\tilde{f}(\boldsymbol{G}) \equiv \frac{1}{\Omega}\int_{\Omega} f(\boldsymbol{r})e^{-i\boldsymbol{G}\cdot\boldsymbol{r}}d\boldsymbol{r}; \tag{A.1}$$

- $\mathcal{F}[f](\boldsymbol{G})$: Fourier transform of non periodic and localized functions. Given a function $f(\boldsymbol{r})$ smooth enough and localized (hence not periodic), its Fourier transform is defined for every value of $\boldsymbol{G}$ (hence not only for reciprocal lattice vectors), by the convention:

$$\mathcal{F}[f](\boldsymbol{G}) \equiv \frac{1}{\Omega}\int_{\boldsymbol{R^3}} f(\boldsymbol{r})e^{-i\boldsymbol{G}\cdot\boldsymbol{r}}d\boldsymbol{r}, \tag{A.2}$$

  where a convenient normalization factor Ω has been introduced ;
- $\hat{H}^{KS}$: instantaneous Kohn-Sham (KS) Hamiltonian ;
- $\boldsymbol{L}$: lattice vector, as defined by the unit cell ;
- M_s : atomic mass of atomic s ;
- $n(\boldsymbol{r})$: ground-state electron-density distribution, defined as $n(\boldsymbol{r}) = \sum_v |\phi_v|^2$;

- v^H: Hartree potential, defined as:

$$v^H(\boldsymbol{r}) = \int_{\boldsymbol{R}^3} \frac{e^2 n(\boldsymbol{r}')}{|\boldsymbol{r} - \boldsymbol{r}'|} d\boldsymbol{r}'. \tag{A.3}$$

  Note that if $n(\boldsymbol{r})$ is periodic, also the Hartree potential is periodic;
- $\hat{\boldsymbol{r}}$: multiplicative position operator or versor along direction pointed by vector $\boldsymbol{r}$. The meaning should be clear from the context.
- $\hat{\boldsymbol{v}}$: unit versor along the direction of $\boldsymbol{v}$
- $\hat{v}_0$: ionic (pseudo-) potential acting on the electrons ;
- w_s^Z : electrostatic energy, equal to $w_s^Z = \frac{e^2}{2} \sum_{t \neq s} \sum_{\boldsymbol{L}} \frac{Z_t Z_s}{|\boldsymbol{R}_s - \boldsymbol{R}_t - \boldsymbol{L}|} + \frac{1}{2} e^2 Z_s^2 \sum_{L \neq 0} \frac{1}{L}$;
- Z_s : atomic charge of atom s, expressed in units of the elementary electronic charge ;
- ϕ, $|\phi\rangle$: a generic normalized wave-function. With brackets, the same wave-function is considered as a vector of a Hilbert space with a scalar product $\langle\,\rangle$;
- Ω : volume of the unit cell ;
- $\boldsymbol{\partial}\epsilon_{GGA}$: derivative of the GGA XC local energy per particle with respect to density gradients. It is a vector whose component along direction i is explicitly given by $\partial\epsilon_{GGA}/\partial(\nabla n_i)$,
- ∇ : gradient with respect to the spatial coordinate $\boldsymbol{r}$;
- $\langle\,\rangle$: standard scalar product between wave-functions ;
- $\dot{[\,]}$: derivative with respect to time ;

A.2 Reciprocal space computation of the S_{ij}^A and S^B

In the following appendix we show explicitly the computation of S_{ij}^A, S^B. S^C and S_{ij}^D can be computed applying more straightforwardly standard Ewald techniques [80], then the full derivation is left to reader. Following the notation of Chapter 5 the indices i and j correspond to the Cartesian coordinates.

Recalling the definition of S_{ij}^A from Eq. (5.56):

$$
\begin{aligned}
S^A_{ij} &= \lim_{x\to 0}\partial_{\boldsymbol{x}_j}\sum_{\boldsymbol{L}\neq\boldsymbol{0}}(\boldsymbol{x}-\boldsymbol{L})_i f(|\boldsymbol{x}-\boldsymbol{L}|) \\
&= \lim_{x\to 0}\partial_{\boldsymbol{x}_j}\sum_{\boldsymbol{L}\neq\boldsymbol{0}}(\boldsymbol{x}-\boldsymbol{L})_i f(|\boldsymbol{x}-\boldsymbol{L}|)\Big(\mathrm{erf}(\sqrt{\eta}|\boldsymbol{x}-\boldsymbol{L}) + \mathrm{erfc}(\sqrt{\eta}|\boldsymbol{x}-\boldsymbol{L})\Big).
\end{aligned}
\tag{A.4}
$$

The term containing $\mathrm{erfc}(\sqrt{\eta}|\boldsymbol{x}-\boldsymbol{L}|)$ can be simply computed in real space:

$$
\begin{aligned}
\lim_{x\to 0}\partial_{\boldsymbol{x}_j}\sum_{\boldsymbol{L}\neq\boldsymbol{0}}(\boldsymbol{x}-\boldsymbol{L})_i f(|\boldsymbol{x}-\boldsymbol{L}|)\mathrm{erfc}(\sqrt{\eta}|\boldsymbol{x}-\boldsymbol{L}|) = -\sum_{\boldsymbol{L}\neq\boldsymbol{0}}\frac{L_iL_j}{L^2}\left(\frac{\mathrm{erfc}(\sqrt{\eta}L)}{L} + 2\sqrt{\frac{\eta}{\pi}}e^{-\eta L^2}\right) + \\
+ \delta_{ij}\sum_{\boldsymbol{L}\neq\boldsymbol{0}}\frac{\mathrm{erfc}(\sqrt{\eta}L)}{L},
\end{aligned}
\tag{A.5}
$$

while:

$$
\begin{aligned}
&\lim_{x\to 0}\partial_{\boldsymbol{x}_j}\sum_{\boldsymbol{L}\neq\boldsymbol{0}}(\boldsymbol{x}-\boldsymbol{L})_i f(|\boldsymbol{x}-\boldsymbol{L}|)\mathrm{erf}(\sqrt{\eta}|\boldsymbol{x}-\boldsymbol{L}|) = \\
&\lim_{x\to 0}\partial_{\boldsymbol{x}_j}\sum_{\boldsymbol{L}}(\boldsymbol{x}-\boldsymbol{L})_i f(|\boldsymbol{x}-\boldsymbol{L}|)\mathrm{erf}(\sqrt{\eta}|\boldsymbol{x}-\boldsymbol{L}|) \\
&-\lim_{x\to 0}\partial_{\boldsymbol{x}_j}\left(x_i\mathrm{erf}(\sqrt{\eta}x)f(x)\right).
\end{aligned}
\tag{A.6}
$$

The first term of Eq. (A.6) can be easily computed in reciprocal space, while it can be demonstrated that $\lim_{x\to 0}\partial_{\boldsymbol{x}_j}\left(x_i\right)\mathrm{erf}(\sqrt{\eta}x)f(x)\big) = 2\delta_{ij}\sqrt{\frac{\eta}{\pi}}$. Summing all these pieces together we get the expression in Eq. (5.65):

$$
\begin{aligned}
S^A_{ij} = &-\sum_{\boldsymbol{L}\neq\boldsymbol{0}}\frac{L_iL_j}{L^2}\left[\frac{\mathrm{erfc}(\sqrt{\eta}L)}{L} + 2\sqrt{\frac{\eta}{\pi}}e^{-\eta L^2}\right] - 2\delta_{ij}\sqrt{\frac{\eta}{\pi}} \\
&+ \delta_{ij}\sum_{\boldsymbol{L}\neq\boldsymbol{0}}\frac{\mathrm{erfc}(\sqrt{\eta}L)}{L} + \sum_{\boldsymbol{G}\neq\boldsymbol{0}}\frac{4\pi}{\Omega}\frac{G_iG_j}{G^2}\frac{\exp(\frac{-G^2}{4\eta})}{G^2}\left[2+\frac{G^2}{2\eta}\right].
\end{aligned}
\tag{A.7}
$$

S^B is, instead, defined as:

$$
S^B = \sum_{\boldsymbol{L}\neq\boldsymbol{0}} f(L),
\tag{A.8}
$$

introducing a functional dependence on x:

$$S^B = \lim_{x\to 0}\sum_{\boldsymbol{L}\neq\boldsymbol{0}} f(|\boldsymbol{x}-\boldsymbol{L}|) = \\ = \lim_{x\to 0}\sum_{\boldsymbol{L}\neq\boldsymbol{0}} f(|\boldsymbol{x}-\boldsymbol{L}|)\mathrm{erf}(\sqrt{\eta}|\boldsymbol{x}-\boldsymbol{L}|) + \lim_{x\to 0}\sum_{\boldsymbol{L}\neq\boldsymbol{0}} f(|\boldsymbol{x}-\boldsymbol{L}|)\mathrm{erfc}(\sqrt{\eta}|\boldsymbol{x}-\boldsymbol{L}|). \tag{A.9}$$

The second expression can be computed in direct space as it is, while the first requires some further work:

$$\lim_{x\to 0}\sum_{\boldsymbol{L}\neq\boldsymbol{0}} f(|\boldsymbol{x}-\boldsymbol{L}|)\mathrm{erf}(\sqrt{\eta}|\boldsymbol{x}-\boldsymbol{L}|) = \lim_{x\to 0}\sum_{\boldsymbol{L}} f(|\boldsymbol{x}-\boldsymbol{L}|)\mathrm{erf}(\sqrt{\eta}|\boldsymbol{x}-\boldsymbol{L}|) - \lim_{x\to 0} f(x)\mathrm{erf}(\sqrt{\eta}x) \\ = \lim_{x\to 0}\sum_{\boldsymbol{L}} f(|\boldsymbol{x}-\boldsymbol{L}|)\mathrm{erf}(\sqrt{\eta}|\boldsymbol{x}-\boldsymbol{L}|) - 2\sqrt{\frac{\eta}{\pi}}. \tag{A.10}$$

Now the first term of Eq. (A.10) can be computed in reciprocal space, thus, summing all the contributions, we can regain the expression in Eq. (5.66) for S^B:

$$S^B = \sum_{\boldsymbol{L}\neq\boldsymbol{0}} f(L)\mathrm{erfc}(\sqrt{\eta}L) - 2\sqrt{\frac{\eta}{\pi}} + \frac{4\pi}{\Omega}\sum_{\boldsymbol{G}\neq\boldsymbol{0}}\frac{\exp(\frac{-G^2}{4\eta})}{G^2} + \frac{4\pi}{\Omega}\left(\frac{1}{\mu^2}-\frac{1}{4\eta}\right). \tag{A.11}$$

Finally we remark that the two terms computed in this Appendix are general and work for generic cells. However, since $\boldsymbol{J}^{nB}$ does not contribute to the value of thermal transport coefficient, up to the present version of `QEHeat` we only implemented the simpler expression for cubic cells:

$$\boldsymbol{J}^{nB} = \sum_s \boldsymbol{V}_s Z_s^2 e^2 (S^B - \frac{1}{2}S^A) = \\ \sum_s \boldsymbol{V}_s Z_s^2 e^2 \left(\frac{2}{3}\sum_{\boldsymbol{L}\neq\boldsymbol{0}} f(L) erfc(\sqrt{\eta}L) + \frac{8\pi}{3\Omega}\sum_{\boldsymbol{G}\neq\boldsymbol{0}}\frac{\exp(\frac{-G^2}{4\eta})}{G^2}\right. \\ \left. -\frac{4}{3}\sqrt{\frac{\eta}{\pi}} - \frac{2\pi}{3\eta\Omega} + \frac{4\pi}{\Omega\mu^2}\right), \tag{A.12}$$

where we defined:

$$S^A \equiv \frac{1}{3}\mathrm{Tr}[S^A_{ij}], \tag{A.13}$$

Tr[·] indicating the trace of a matrix. The following identity can be used to recover the formula for cubic systems from the general one:

$$\sum_{\boldsymbol{L}} e^{-\eta L^2} = \frac{\sqrt{\pi^3}}{\Omega\sqrt{\eta}^3}\sum_{\boldsymbol{G}}\exp\left(\frac{-G^2}{4\eta}\right) \tag{A.14}$$

Bibliography

[1] M. S. Green, "Markoff random processes and the statistical mechanics of time-dependent phenomena, ii. irreversible processes in fluids", J. Chem. Phys. **22**, 398–413 (1954) (cit. on pp. 1, 5, 9, 11, 77).

[2] R. Kubo, "Statistical-mechanical theory of irreversible processes. i. general theory and simple applications to magnetic and conduction problems", J. Phys. Soc. Jpn. **12**, 570 (1957) (cit. on pp. 1, 5, 9, 11, 77).

[3] D. J. Evans and G. P. Morriss, *Statistical Mechanics of Nonequilibrium Liquids* (2007), 10.22459/smnl.08.2007 (cit. on pp. 1, 5, 11).

[4] M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids*, Vol. 1 (Oxford University Press, 2017), p. 442, 10.1093/oso/9780198803195.001.0001 (cit. on pp. 1, 5, 11).

[5] M. Born and R. Oppenheimer, "Zur Quantentheorie der Molekeln", Annalen der physik **389**, 457 (1927) 10.1002/andp.19273892002 (cit. on p. 1).

[6] D. Marx and J. Hutter, *Ab initio molecular dynamics: basic theory and advanced methods* (Cambridge University Press, 2009), 10.1017/CBO9780511609633 (cit. on p. 1).

[7] R. Car and M. Parrinello, "Unified Approach for Molecular Dynamics and Density-Functional Theory", Phys. Rev. Lett. **55**, 2471 (1985) (cit. on pp. 1, 55, 79).

[8] A. Marcolongo, P. Umari, and S. Baroni, "Microscopic theory and quantum simulation of atomic heat transport", Nature Physics **12**, 80 (2016) 10.1038/nphys3509 (cit. on pp. 1, 14, 25, 27, 28, 30, 31, 38, 48, 64).

[9] S. Baroni, R. Bertossa, L. Ercole, F. Grasselli, and A. Marcolongo, "Heat transport in insulators from ab initio Green-Kubo theory", in *Handbook of materials modeling: applications: current and emerging materials*, edited by W. Andreoni and S. Yip (Springer International Publishing, Cham, 2018), pp. 1–36, 10.1007/978-3-319-50257-1_12-1 (cit. on pp. 1, 14, 18, 20, 24, 25, 38, 64, 92).

[10] F. Grasselli and S. Baroni, "Invariance principles in the theory and computation of transport coefficients", European Physical Journal B (2021) 10.1140/epjb/s10051-021-00152-5 (cit. on pp. 1, 14, 18, 20, 25).

[11] L. Ercole, A. Marcolongo, and S. Baroni, "Accurate thermal conductivities from optimally short molecular dynamics simulations", Scientific Reports **7**, 15835 (2017) 10.1038/s41598-017-15843-2 (cit. on pp. 1, 19–22, 24, 25, 92).

[12] R. Bertossa, F. Grasselli, L. Ercole, and S. Baroni, "Theory and Numerical Simulation of Heat Transport in Multicomponent Systems", Physical Review Letters **122**, 255901 (2019) 10.1103/PhysRevLett.122.255901 (cit. on pp. 1, 18, 20, 24, 25, 38, 40, 48, 56, 64).

[13] J. Behler and M. Parrinello, "Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces", Physical Review Letters **98**, 146401 (2007) 10.1103/PhysRevLett.98.146401 (cit. on pp. 2, 62).

[14] R. Kondor, *N-body networks: a covariant hierarchical neural network architecture for learning atomic potentials*, 2018 (cit. on p. 2).

[15] J. S. Smith, O. Isayev, and A. E. Roitberg, "ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost", Chemical Science **8**, 3192 (2017) 10.1039/C6SC05720A (cit. on p. 2).

[16] L. Zhang, J. Han, H. Wang, R. Car, and W. E, "Deep Potential Molecular Dynamics: A Scalable Model with the Accuracy of Quantum Mechanics", Physical Review Letters **120**, 143001 (2018) 10.1103/PhysRevLett.120.143001 (cit. on pp. 2, 62, 63).

[17] A. P. Bartók, M. C. Payne, R. Kondor, and G. Csányi, "Gaussian Approximation Potentials: The Accuracy of Quantum Mechanics, without the Electrons", Physical Review Letters **104**, 136403 (2010) 10.1103/PhysRevLett.104.136403 (cit. on p. 2).

[18] L. Zhang, J. Han, H. Wang, W. Saidi, R. Car, and W. E, "End-to-end symmetry preserving inter-atomic potential energy model for finite and extended systems", in *Advances in neural information processing systems 31*, edited by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Curran Associates, Inc., 2018), pp. 4436–4446 (cit. on pp. 2, 61–63).

[19] L. Zhang, H. Wang, R. Car, and W. E, "Phase diagram of a deep potential water model", Phys. Rev. Lett. **126**, 236001 (2021) 10.1103/PhysRevLett.126.236001 (cit. on pp. 2, 59, 72, 87, 88).

[20] W. Jiang, Y. Zhang, L. Zhang, and H. Wang, "Accurate Deep Potential model for the Al–Cu–Mg alloy in the full concentration space*", Chinese Physics B **30**, 050706 (2021) 10.1088/1674-1056/abf134 (cit. on pp. 2, 59).

[21] C. Zhang, F. Tang, M. Chen, J. Xu, L. Zhang, D. Y. Qiu, J. P. Perdew, M. L. Klein, and X. Wu, "Modeling liquid water by climbing up jacob's ladder in density functional theory facilitated by using deep neural network potentials", The Journal of Physical Chemistry B **125**, PMID: 34533960, 11444 (2021) 10.1021/acs.jpcb.1c03884 (cit. on pp. 2, 59).

[22] J. Wu, Y. Zhang, L. Zhang, and S. Liu, "Deep learning of accurate force field of ferroelectric HfO2", Phys. Rev. B **103**, 024108 (2021) 10.1103/PhysRevB.103.024108 (cit. on pp. 2, 59).

[23] T. E. Gartner, L. Zhang, P. M. Piaggi, R. Car, A. Z. Panagiotopoulos, and P. G. Debenedetti, "Signatures of a liquid–liquid transition in an ab initio deep neural network model for water", Proceedings of the National Academy of Sciences **117**, 26040 (2020) 10.1073/pnas.2015440117 (cit. on pp. 2, 59, 87, 89).

[24] H. Niu, L. Bonati, P. M. Piaggi, and M. Parrinello, "Ab initio phase diagram and nucleation of gallium", Nature Communications **11**, 2654 (2020) 10.1038/s41467-020-16372-9 (cit. on pp. 2, 59).

[25] W. Jia, H. Wang, M. Chen, D. Lu, L. Lin, R. Car, W. E, and L. Zhang, "Pushing the limit of molecular dynamics with ab initio accuracy to 100 million atoms with machine learning", in *Proceedings of the international conference for high performance computing, networking, storage and analysis* (IEEE Press, 2020) (cit. on pp. 2, 59).

[26] A. Marcolongo, T. Binninger, F. Zipoli, and T. Laino, "Simulating diffusion properties of solid-state electrolytes via a neural network potential: Performance and training scheme", ChemSystemsChem **2** (2019) 10.1002/syst.201900031 (cit. on pp. 2, 59).

[27] J. Huang, L. Zhang, H. Wang, J. Zhao, J. Cheng, and W. E, "Deep potential generation scheme and simulation protocol for the Li10GeP2S12-type superionic conductors", The Journal of Chemical Physics **154**, 094703 (2021) (cit. on pp. 2, 59).

[28] L. Zhang, M. Chen, X. Wu, H. Wang, W. E, and R. Car, "Deep neural network for the dielectric response of insulators", Phys. Rev. B **102**, 041121(R) (2020) 10.1103/PhysRevB.102.041121 (cit. on pp. 2, 59, 94).

[29] K. Lodders, "Solar system abundances and condensation temperatures of the elements", The Astrophysical Journal **591**, 1220 (2003) 10.1086/375492 (cit. on p. 2).

[30] W. Good, "The role of water in the origin of life and its function in the primitive gene", Journal of Theoretical Biology **39**, 249 (1973) https://doi.org/10.1016/0022-5193(73)90097-0 (cit. on p. 2).

[31] "Prebiotic chemistry and origins of life research with atomistic computer simulations", Physics of Life Reviews **34-35**, 105 (2020) https://doi.org/10.1016/j.plrev.2018.09.004 (cit. on p. 2).

[32] P. Gallo, K. Amann-Winkel, C. A. Angell, M. A. Anisimov, F. Caupin, C. Chakravarty, E. Lascaris, T. Loerting, A. Z. Panagiotopoulos, J. Russo, J. A. Sellberg, H. E. Stanley, H. Tanaka, C. Vega, L. Xu, and L. G. M. Pettersson, "Water: a tale of two liquids", Chemical Reviews **116**, 7463 (2016) 10.1021/acs.chemrev.5b00750 (cit. on pp. 2, 99).

[33] C. Cavazzoni, G. L. Chiarotti, S. Scandolo, E. Tosatti, M. Bernasconi, and M. Parrinello, "Superionic and metallic states of water and ammonia at giant planet conditions", Science **283**, 44 (1999) 10.1126/science.283.5398.44 (cit. on pp. 2, 91).

[34] I.-F. W. Kuo, C. J. Mundy, M. J. McGrath, J. I. Siepmann, J. VandeVondele, M. Sprik, J. Hutter, B. Chen, M. L. Klein, F. Mohamed, M. Krack, and M. Parrinello, "Liquid water from first principles: investigation of different sampling approaches", The Journal of Physical Chemistry B **108**, 12990 (2004) 10.1021/jp047788i (cit. on pp. 2, 77).

[35] J. C. Grossman, E. Schwegler, E. W. Draeger, F. Gygi, and G. Galli, "Towards an assessment of the accuracy of density functional theory for first principles simulations of water", The Journal of Chemical Physics **120**, 300 (2004) 10.1063/1.1630560 (cit. on pp. 2, 55, 57, 77).

[36] E. Schwegler, J. C. Grossman, F. Gygi, and G. Galli, "Towards an assessment of the accuracy of density functional theory for first principles simulations of water. II", Journal of Chemical Physics **121**, 5400 (2004) 10.1063/1.1782074 (cit. on pp. 2, 55, 57, 77).

[37] T. Todorova, A. P. Seitsonen, J. Hutter, I.-F. W. Kuo, and C. J. Mundy, "Molecular dynamics simulation of liquid water: hybrid density functionals", The Journal of Physical Chemistry B **110**, 3685 (2006) 10.1021/jp055127v (cit. on pp. 2, 77).

[38] M. J. Gillan, D. Alfè, and A. Michaelides, "Perspective: How good is DFT for water?", The Journal of Chemical Physics **144**, 130901 (2016) 10.1063/1.4944633 (cit. on pp. 2, 77, 87).

[39] M. Chen, H.-Y. Ko, R. C. Remsing, M. F. Calegari Andrade, B. Santra, Z. Sun, A. Selloni, R. Car, M. L. Klein, J. P. Perdew, and X. Wu, "Ab initio theory and modeling of water", Proceedings of the National Academy of Sciences **114**, 10846 (2017) 10.1073/pnas.1712499114 (cit. on pp. 2, 57, 74, 77, 87).

[40] M. D. LaCount and F. Gygi, "Ensemble first-principles molecular dynamics simulations of water using the SCAN meta-GGA density functional", The Journal of Chemical Physics **151**, 164101 (2019) 10.1063/1.5124957 (cit. on pp. 2, 77, 87).

[41] L. Zheng, M. Chen, Z. Sun, H.-Y. Ko, B. Santra, P. Dhuvad, and X. Wu, "Structural, electronic, and dynamical properties of liquid water by ab initio molecular dynamics based on scan functional within the canonical ensemble", The Journal of Chemical Physics **148**, 164505 (2018) 10.1063/1.5023611 (cit. on pp. 2, 77).

[42] T. D. Kühne, M. Krack, and M. Parrinello, "Static and dynamical properties of liquid water from first principles by a novel Car-Parrinello-like approach", Journal of Chemical Theory and Computation **5**, 235 (2009) 10.1021/ct800417q (cit. on pp. 2, 77).

[43] A. Marcolongo, R. Bertossa, D. Tisi, and S. Baroni, "QEHeat: an open-source energy flux calculator for the computation of heat-transport coefficients from first principles", Computer Physics Communications **269**, 108090 (2021) https://doi.org/10.1016/j.cpc.2021.108090 (cit. on pp. 2, 27, 55).

[44] J. P. Perdew, K. Burke, and M. Ernzerhof, "Generalized gradient approximation made simple", Phys. Rev. Lett. **77**, 3865 (1996) 10.1103/PhysRevLett.77.3865 (cit. on pp. 3, 29, 79).

[45] L. Onsager, "Reciprocal relations in irreversible processes. i.", Phys. Rev. **37**, 405 (1931) 10.1103/PhysRev.37.405 (cit. on pp. 5, 8).

[46] L. Onsager, "Reciprocal relations in irreversible processes. ii.", Phys. Rev. **38**, 2265 (1931) 10.1103/PhysRev.38.2265 (cit. on pp. 5, 8).

[47] H. B. G. Casimir, "On onsager's principle of microscopic reversibility", Rev. Mod. Phys. **17**, 343 (1945) 10.1103/RevModPhys.17.343 (cit. on p. 8).

[48] J. B. J. Fourier, *The analytical theory of heat* (Cambridge University Press, 1878) (cit. on p. 8).

[49] R. Kubo, M. Yokota, and S. Nakajima, "Statistical-mechanical theory of irreversible processes. ii. response to thermal disturbance", Journal of the Physical Society of Japan **12**, 1203 (1957) 10.1143/JPSJ.12.1203 (cit. on p. 9).

[50] R. Peierls, "Zur kinetischen theorie der wärmeleitung in kristallen", Annalen der Physik **395**, Translated in: R. E. Peierls, *On the Kinetic Theory of Thermal Conduction in Crystals*, in R. H. Dalitz and R. E. Peierls (ed.), Selected Scientific Papers of Sir Rudolf Peierls with Commentary, vol. 19, World Scientific Publishing Co., Singapore, 1997., 1055 (1929) 10.1002/andp.19293950803 (cit. on p. 11).

[51] P. Klemens, "Thermal Conductivity and Lattice Vibrational Modes", in *Solid state physics - advances in research and applications*, Vol. 7 (1958), pp. 1–98, 10.1016/S0081-1947(08)60551-2 (cit. on p. 11).

[52] D. A. Broido, M. Malorny, G. Birner, N. Mingo, and D. A. Stewart, "Intrinsic lattice thermal conductivity of semiconductors from first principles", Applied Physics Letters **91**, 231922 (2007) 10.1063/1.2822891 (cit. on p. 11).

[53] F. Zhou, W. Nielson, Y. Xia, and V. Ozolinš, "Lattice anharmonicity and thermal conductivity from compressive sensing of first-principles calculations", Physical Review Letters **113**, 185501 (2014) 10.1103/PhysRevLett.113.185501 (cit. on p. 11).

[54] J. E. Turney, E. S. Landry, A. J. H. McGaughey, and C. H. Amon, "Predicting phonon properties and thermal conductivity from anharmonic lattice dynamics calculations and molecular dynamics simulations", Phys. Rev. B **79**, 064301 (2009) 10.1103/PhysRevB.79.064301 (cit. on p. 11).

[55] F. Müller-Plathe, "A simple nonequilibrium molecular dynamics method for calculating the thermal conductivity", The Journal of Chemical Physics **106**, 6082 (1997) 10.1063/1.473271 (cit. on p. 11).

[56] A. Tenenbaum, G. Ciccotti, and R. Gallico, "Stationary nonequilibrium states by molecular dynamics. Fourier's law", Physical Review A **25**, 2778 (1982) 10.1103/PhysRevA.25.2778 (cit. on p. 11).

[57] S. Stackhouse, L. Stixrude, and B. B. Karki, "Thermal Conductivity of Periclase (MgO) from First Principles", Physical Review Letters **104**, 208501 (2010) 10.1103/PhysRevLett.104.208501 (cit. on p. 13).

[58] L. Ercole, A. Marcolongo, P. Umari, and S. Baroni, "Gauge Invariance of Thermal Transport Coefficients", JLTP **185**, 79 (2016) 10.1007/s10909-016-1617-6 (cit. on pp. 14, 19, 27, 38).

[59] Z. Fan, L. F. C. Pereira, H.-Q. Wang, J.-C. Zheng, D. Donadio, and A. Harju, "Force and heat current formulas for many-body potentials in molecular dynamics simulations with applications to thermal conductivity calculations", Phys. Rev. B **92**, 094301 (2015) 10.1103/PhysRevB.92.094301 (cit. on pp. 16, 78).

[60] D. Surblys, H. Matsubara, G. Kikugawa, and T. Ohara, "Application of atomic stress to compute heat flux via molecular dynamics for systems with many-body interactions", Phys. Rev. E **99**, 051301 (2019) 10.1103/PhysRevE.99.051301 (cit. on p. 16).

[61] N. Wiener, "Generalized harmonic analysis", Acta Mathematica **55**, 117 (1930) 10.1007/BF02546511 (cit. on pp. 19, 79).

[62] A. Khintchine, "Korrelationstheorie der stationaren stochastischen Prozesse", Mathematische Annalen **109**, 604 (1934) 10.1007/BF01449156 (cit. on pp. 19, 79).

[63] R. J. Muirhead, *Aspects of multivariate statistical theory* (Wiley-Interscience, 2005) (cit. on p. 21).

[64] G. Arfken, *Mathematical methods for physicists*, Third (Academic Press, Inc., San Diego, 1985) (cit. on p. 21).

[65] D. Childers, D. Skinner, and R. Kemerait, "The cepstrum: a guide to processing", Proceedings of the IEEE **65**, 1428 (1977) 10.1109/PROC.1977.10747 (cit. on p. 22).

[66] M. Peligrad and W. B. Wu, "Central limit theorem for fourier transforms of stationary processes", Ann. Probab. **38**, 2009 (2010) 10.1214/10-AOP530 (cit. on p. 22).

[67] H. Akaike, "A new look at the statistical model identification", IEEE Transactions on Automatic Control **19**, 716 (1974) 10.1109/TAC.1974.1100705 (cit. on p. 24).

[68] L. Ercole, R. Bertossa, S. Bisacchi, and S. Baroni, *Sportran: a code to estimate transport coefficients from the cepstral analysis of (multivariate) current time series*, 2022 (cit. on pp. 24, 30, 56, 71, 79).

[69] G. Box, G. Jenkins, G. Reinsel, and G. Ljung, *Time series analysis: forecasting and control*, Wiley Series in Probability and Statistics (Wiley, 2015) (cit. on p. 25).

[70] A. Marcolongo, L. Ercole, and S. Baroni, "Gauge Fixing for Heat-Transport Simulations.", JCTC **16**, 3352 (2020) 10.1021/acs.jctc.9b01174 (cit. on pp. 25, 38, 40, 55).

[71] J. Kang and L.-W. Wang, "First-principles Green-Kubo method for thermal conductivity calculations", Physical Review B **96**, 020302(R) (2017) 10.1103/PhysRevB.96.020302 (cit. on p. 27).

[72] C. Carbogno, R. Ramprasad, and M. Scheffler, "Ab initio Green-Kubo approach for the thermal conductivity of solids", Phys. Rev. Lett. **118**, 175901 (2017) 10.1103/PhysRevLett.118.175901 (cit. on p. 27).

[73] P. Giannozzi, S. Baroni, N. Bonini, M. Calandra, R. Car, C. Cavazzoni, D. Ceresoli, G. L. Chiarotti, M. Cococcioni, I. Dabo, A. D. Corso, S. de Gironcoli, S. Fabris, G. Fratesi, R. Gebauer, U. Gerstmann, C. Gougoussis, A. Kokalj, M. Lazzeri, L. Martin-Samos, N. Marzari, F. Mauri, R. Mazzarello, S. Paolini, A. Pasquarello, L. Paulatto, C. Sbraccia, S. Scandolo, G. Sclauzero, A. P. Seitsonen, A. Smogunov, P. Umari, and R. M. Wentzcovitch, "QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials", Journal of Physics: Condensed Matter **21**, 395502 (2009) 10.1088/0953-8984/21/39/395502 (cit. on pp. 27, 30, 55, 79).

[74] P. Giannozzi, O. Andreussi, T. Brumme, O. Bunau, M. Buongiorno Nardelli, M. Calandra, R. Car, C. Cavazzoni, D. Ceresoli, M. Cococcioni, N. Colonna, I. Carnimeo, A. Dal Corso, S. de Gironcoli, P. Delugas, R. A. DiStasio, A. Ferretti, A. Floris, G. Fratesi, G. Fugallo, R. Gebauer, U. Gerstmann, F. Giustino, T. Gorni, J. Jia, M. Kawamura, H.-Y. Ko, A. Kokalj, E. Küçükbenli, M. Lazzeri, M. Marsili, N. Marzari, F. Mauri, N. L. Nguyen, H.-V. Nguyen, A. Otero-de-la Roza, L. Paulatto, S. Poncé, D. Rocca, R. Sabatini, B. Santra, M. Schlipf, A. P. Seitsonen, A. Smogunov, I. Timrov, T. Thonhauser, P. Umari, N. Vast, X. Wu, and S. Baroni, "Advanced capabilities for materials modelling with Quantum ESPRESSO", Journal of Physics. Condensed Matter **29** (2017) 10.1088/1361-648X/aa8f79 (cit. on pp. 27, 30, 31, 55, 79).

[75] P. Giannozzi, O. Baseggio, P. Bonfà, D. Brunato, R. Car, I. Carnimeo, C. Cavazzoni, S. de Gironcoli, P. Delugas, F. F. Ruffino, A. Ferretti, N. Marzari, I. Timrov, A. Urru, and S. Baroni, "Quantum ESPRESSO toward the exascale", The Journal of Chemical Physics **152**, 154105 (2020) 10.1063/5.0005082 (cit. on pp. 27, 30, 55, 79).

[76] R. M. Martin, *Electronic structure: basic theory and practical methods* (Cambridge University Press, 2004), 10.1017/CBO9780511805769 (cit. on p. 27).

[77] W. Kohn and L. J. Sham, "Self-consistent equations including exchange and correlation effects", Phys. Rev. **140**, A1133 (1965) 10.1103/PhysRev.140.A1133 (cit. on p. 27).

[78] N. Chetty and R. M. Martin, "First-principles energy density and its applications to selected polar surfaces", Phys. Rev. B **45**, 6074 (1992) 10.1103/PhysRevB.45.6074 (cit. on p. 28).

[79] S. Baroni, S. de Gironcoli, A. Dal Corso, and P. Giannozzi, "Phonons and related crystal properties from density-functional perturbation theory", Rev.

Mod. Phys. **73**, 515 (2001) 10.1103/RevModPhys.73.515 (cit. on pp. 31, 40).

[80] G. Grosso and G. Parravicini, *Solid state physics* (Elsevier Science, 2000) (cit. on pp. 38, 39, 102).

[81] D. J. Thouless, "Quantization of particle transport", Phys. Rev. B **27**, 6083 (1983) 10.1103/PhysRevB.27.6083 (cit. on p. 40).

[82] *Qeheat doc page*, https://gitlab.com/QEF/q-e/-/tree/master/QEHeat/Doc, 2021-2022 (cit. on p. 44).

[83] F. Grasselli, L. Stixrude, and S. Baroni, "Heat and charge transport in H2O at ice-giant conditions from ab initio molecular dynamics simulations", Nature Communications **11**, 3605 (2020) 10.1038/s41467-020-17275-5 (cit. on pp. 48, 52, 53, 91–93, 96, 97).

[84] F. Grasselli, L. Stixrude, and S. Baroni, *Heat and charge transport in h2o at ice-giant conditions from ab initio molecular dynamics simulations*, 10.24435/materialscloud:hn-6f (cit. on p. 53).

[85] *Marconi, the tier-0 system*, Accessed: 2021-04-09 (cit. on p. 54).

[86] D. Tisi, L. Zhang, R. Bertossa, H. Wang, R. Car, and S. Baroni, "Heat transport in liquid water from first-principles and deep neural network simulations", Phys. Rev. B **104**, 224202 (2021) 10.1103/PhysRevB.104.224202 (cit. on pp. 55, 64, 67, 87, 88).

[87] D. R. Hamann, "Optimized norm-conserving Vanderbilt pseudopotentials", Physical Review B **88**, 085117 (2013) 10.1103/PhysRevB.88.085117 (cit. on pp. 55, 79).

[88] P. H.-L. Sit and N. Marzari, "Static and dynamical properties of heavy water at ambient conditions from first-principles molecular dynamics", The Journal of Chemical Physics **122**, 204510 (2005) 10.1063/1.1908913 (cit. on pp. 55, 57, 73, 74, 79).

[89] S. Yoo, X. C. Zeng, and S. S. Xantheas, "On the phase diagram of water with density functional theory potentials: the melting temperature of ice Ih with the Perdew–Burke–Ernzerhof and Becke–Lee–Yang–Parr functionals", The Journal of Chemical Physics **130**, 221102 (2009) 10.1063/1.3153871 (cit. on pp. 55, 73, 74, 79, 80, 88).

[90] M. L. V. Ramires, C. A. Nieto de Castro, Y. Nagasaka, A. Nagashima, M. J. Assael, and W. A. Wakeham, "Standard Reference Data for the Thermal Conductivity of Water", Journal of Physical and Chemical Reference Data **24**, 1377 (1995) 10.1063/1.555963 (cit. on pp. 57, 71).

[91] G. M. Sommers, M. F. Calegari Andrade, L. Zhang, H. Wang, and R. Car, "Raman spectrum and polarizability of liquid water from deep neural networks", Phys. Chem. Chem. Phys. **22**, 10592 (2020) 10.1039/D0CP01893G (cit. on p. 59).

[92] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, Chapter 6 (MIT Press, 2016) (cit. on pp. 59, 60, 63).

[93] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization", in 3rd international conference on learning representations, ICLR2015,san diego, ca, usa, may 7-9, 2015, conference track proceedings, edited by Y. Bengio and Y. LeCun (2015) (cit. on pp. 60, 63, 67, 82).

[94] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, *TensorFlow: large-scale machine learning on heterogeneous systems*, Software available from tensorflow.org, 2015 (cit. on pp. 62, 63).

[95] H. Wang, L. Zhang, J. Han, and W. E, "DeePMD-kit: A deep learning package for many-body potential energy representation and molecular dynamics", Computer Physics Communications **228**, 178 (2018) 10.1016/j.cpc.2018.03.016 (cit. on pp. 62–64).

[96] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors", Nature **323**, 533 (1986) (cit. on p. 63).

[97] S. Plimpton, "Fast parallel algorithms for short-range molecular dynamics", Journal of Computational Physics **117**, 1 (1995) https://doi.org/10.1006/jcph.1995.1039 (cit. on p. 64).

[98] A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. Michael Brown, P. S. Crozier, P. J. in 't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen, R. Shan, M. Stevens, J. Tranchida, C. Trott, and S. J. Plimpton, "LAMMPS - A flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales", Computer Physics Communications, 108171 (2021) 10.1016/j.cpc.2021.108171 (cit. on pp. 64, 83).

[99] *Lammps website*, https://www.lammps.org (cit. on p. 64).

[100] a more detailed description of the use of the LAMMPS interface for the energy current is given the DeePMD-kit documentation https://github.com/deepmodeling/deepmd-kit/blob/master/doc/third-party/lammps-command.md (cit. on p. 64).

[101] R. Sun, D. Li, S. Liang, T. Ding, and R Srikant, *The global landscape of neural networks: an overview*, 2020 (cit. on p. 64).

[102] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the loss landscape of neural nets", in Advances in neural information processing systems, Vol. 31, edited by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (2018) (cit. on p. 64).

[103] L. Zhang, D.-Y. Lin, H. Wang, R. Car, and W. E, "Active learning of uniformly accurate interatomic potentials for materials simulation", Phys. Rev. Materials **3**, 023804 (2019) 10.1103/PhysRevMaterials.3.023804 (cit. on pp. 64, 82).

[104] E. W. Weisstein, Correlation Coefficient, From MathWorld – A Wolfram Web Resource https://mathworld.wolfram.com/CorrelationCoefficient.html, 2022 (cit. on pp. 68, 82).

[105] B. Monserrat, J. G. Brandenburg, E. A. Engel, and B. Cheng, "Liquid water contains the building blocks of diverse ice phases", Nature Communications **11**, 5757 (2020) 10.1038/s41467-020-19606-y (cit. on p. 68).

[106] C. Zeni, A. Anelli, A. Glielmo, and K. Rossi, *Machine learning potentials always extrapolate, it does not matter*, 2021 (cit. on p. 68).

[107] *Nist chemistry webbook*, https://webbook.nist.gov/chemistry/fluid/, Accessed: 2020 (cit. on p. 73).

[108] T. I. Gartner, L. Zhang, P. Piaggi, R. Car, A. Panagiotopoulos, and P. Debenedetti, *Data from "Signatures of a liquid-liquid transition in an ab initio deep neural network model for water"*, http://arks.princeton.edu/ark:/88435/dsp01b5644v47m, 2020, 10.34770/45m3-am91 (cit. on p. 72).

[109] P. M. Piaggi, A. Z. Panagiotopoulos, P. G. Debenedetti, and R. Car, "Phase equilibrium of water with hexagonal and cubic ice using the scan functional", Journal of Chemical Theory and Computation **17**, 3065 (2021) 10.1021/acs.jctc.1c00041 (cit. on pp. 72, 87, 88).

[110] I. C. Yeh and G. Hummer, "System-size dependence of diffusion coefficients and viscosities from molecular dynamics simulations with periodic boundary conditions", Journal of Physical Chemistry B **108**, 15873 (2004) 10.1021/jp0477147 (cit. on p. 74).

[111] M. Puligheddu and G. Galli, "Atomistic simulations of the thermal conductivity of liquids", Physical Review Materials **4**, 053801 (2020) 10.1103/PhysRevMaterials.4.053801 (cit. on p. 74).

[112] A. F. Olea and J. K. Thomas, "Rate constants for reactions in viscous media: correlation between the viscosity of the solvent and the rate constant of the diffusion-controlled reactions", Journal of the American Chemical Society **110**, 4494 (1988) 10.1021/ja00222a002 (cit. on p. 77).

[113] R. E. McKinnie and J. S. Olson, "Effects of solvent composition and viscosity on the rates of CO binding to heme proteins.", Journal of Biological Chemistry **256**, 8928 (1981) https://doi.org/10.1016/S0021-9258(19)52488-7 (cit. on p. 77).

[114] H. Kyushiki and A. Ikai, "The effect of solvent viscosity on the rate-determining step of fatty acid synthetase", Proteins: Structure, Function, and Bioinformatics **8**, 287 (1990) https://doi.org/10.1002/prot.340080310 (cit. on p. 77).

[115] Bolmont, E., Breton, S. N., Tobie, G., Dumoulin, C., Mathis, S., and Grasset, O., "Solid tidal friction in multi-layer planets: application to Earth, Venus, a super Earth and the TRAPPIST-1 planets - potential approximation of a multi-layer planet as a homogeneous body", A&A **644**, A165 (2020) 10.1051/0004-6361/202038204 (cit. on p. 77).

[116] C. Dumoulin, G. Tobie, O. Verhoeven, P. Rosenblatt, and N. Rambaux, "Tidal constraints on the interior of Venus", Journal of Geophysical Research: Planets **122**, 1338 (2017) https://doi.org/10.1002/2016JE005249 (cit. on p. 77).

[117] M. A. González and J. L. F. Abascal, "The shear viscosity of rigid water models", The Journal of Chemical Physics **132**, 096101 (2010) 10.1063/1.3330544 (cit. on p. 77).

[118] S. Tazi, A. Boţan, M. Salanne, V. Marry, P. Turq, and B. Rotenberg, "Diffusion coefficient and shear viscosity of rigid water models", Journal of Physics: Condensed Matter **24**, 284117 (2012) 10.1088/0953-8984/24/28/284117 (cit. on p. 77).

[119] D. M. Heyes, E. R. Smith, and D. Dini, "Shear stress relaxation and diffusion in simple liquids by molecular dynamics simulations: Analytic expressions and paths to viscosity", The Journal of Chemical Physics **150**, 174504 (2019) 10.1063/1.5095501 (cit. on p. 77).

[120] P. Montero de Hijes, E. Sanz, L. Joly, C. Valeriani, and F. Caupin, "Viscosity and self-diffusion of supercooled and stretched water from molecular dynamics simulations", The Journal of Chemical Physics **149**, 094503 (2018) 10.1063/1.5042209 (cit. on p. 77).

[121] P. Kumar, S. V. Buldyrev, S. R. Becker, P. H. Poole, F. W. Starr, and H. E. Stanley, "Relation between the widom line and the breakdown of the stokes–einstein relation in supercooled water", Proceedings of the National Academy of Sciences **104**, 9575 (2007) 10.1073/pnas.0702608104 (cit. on p. 77).

[122] L. Xu, F. Mallamace, Z. Yan, F. W. Starr, S. V. Buldyrev, and H. Eugene Stanley, "Appearance of a fractional stokes–einstein relation in water and a structural interpretation of its onset", Nature Physics **5**, 565 (2009) 10.1038/nphys1328 (cit. on p. 77).

[123] I. N. Tsimpanogiannis, S. H. Jamali, I. G. Economou, T. J. H. Vlugt, and O. A. Moultos, "On the validity of the stokes–einstein relation for various water force fields", Molecular Physics **118**, e1702729 (2020) 10.1080/00268976.2019.1702729 (cit. on p. 77).

[124] C. Malosso, L. Zhang, R. Car, S. Baroni, and D. Tisi, "Viscosity in water from first-principles and deep-neural-network simulations", arxiv:2203.01262 (2022) (cit. on p. 77).

[125] D. H. Tsai, "The virial theorem and stress calculation in molecular dynamics", The Journal of Chemical Physics **70**, 1375 (1979) 10.1063/1.437577 (cit. on p. 78).

[126] O. H. Nielsen and R. M. Martin, "First-principles calculation of stress", Phys. Rev. Lett. **50**, 697 (1983) 10.1103/PhysRevLett.50.697 (cit. on p. 78).

[127] O. H. Nielsen and R. M. Martin, "Quantum-mechanical theory of stress and force", Phys. Rev. B **32**, 3780 (1985) 10.1103/PhysRevB.32.3780 (cit. on p. 78).

[128] G. J. Martyna, M. L. Klein, and M. Tuckerman, "Nosé–Hoover chains: the canonical ensemble via continuous dynamics", The Journal of Chemical Physics **97**, 2635 (1992) 10.1063/1.463940 (cit. on p. 79).

[129] E. W. Weisstein, Moving Average, From MathWorld – A Wolfram Web Resource https://mathworld.wolfram.com/MovingAverage.html (cit. on p. 79).

[130] Y. Zhang, H. Wang, W. Chen, J. Zeng, L. Zhang, H. Wang, and W. E, "DP-GEN: a concurrent learning platform for the generation of reliable deep learning based potential energy models", Computer Physics Communications **253**, 107206 (2020) https://doi.org/10.1016/j.cpc.2020.107206 (cit. on p. 82).

[131] A. P. Gaiduk, F. Gygi, and G. Galli, "Density and compressibility of liquid water and ice from first-principles simulations with hybrid functionals", The Journal of Physical Chemistry Letters **6**, PMID: 26267178, 2902 (2015) 10.1021/acs.jpclett.5b00901 (cit. on p. 87).

[132] W. M. Haynes, D. R. Lide, and T. J. Bruno, *Crc handbook of chemistry and physics: a ready-reference book of chemical and physical data* (Florida: CRC Press, 2016) (cit. on pp. 87, 88).

[133] A. Dehaoui, B. Issenmann, and F. Caupin, "Viscosity of deeply supercooled water and its coupling to molecular diffusion", Proceedings of the National Academy of Sciences **112**, 12020 (2015) 10.1073/pnas.1508996112 (cit. on pp. 87, 88).

[134] P. H. Poole, F. Sciortino, U. Essmann, and H. E. Stanley, "Phase behaviour of metastable water", Nature **360**, 324 (1992) 10.1038/360324a0 (cit. on p. 89).

[135] B. Santra, R. A. D. Jr., F. Martelli, and R. Car, "Local structure analysis in ab initio liquid water", Molecular Physics **113**, 2829 (2015) 10.1080/00268976.2015.1058432 (cit. on p. 89).

[136] Y.-W. Kuo, P.-H. Tang, H. Wang, T.-M. Wu, and S. Saito, "Tetrahedral structure of supercooled water at ambient pressure and its influence on dynamic relaxation: comparative study of water models", Journal of Molecular Liquids **341**, 117269 (2021) https://doi.org/10.1016/j.molliq.2021.117269 (cit. on p. 89).

[137] R. Foffi, J. Russo, and F. Sciortino, "Structural and topological changes across the liquid–liquid transition in water", The Journal of Chemical Physics **154**, 184506 (2021) 10.1063/5.0049299 (cit. on p. 89).

[138] A. K. Soper and M. A. Ricci, "Structures of high-density and low-density water", Phys. Rev. Lett. **84**, 2881 (2000) 10.1103/PhysRevLett.84.2881 (cit. on p. 89).

[139] M. Millot, S. Hamel, J. R. Rygg, P. M. Celliers, G. W. Collins, F. Coppari, D. E. Fratanduono, R. Jeanloz, D. C. Swift, and J. H. Eggert, "Experimental evidence for superionic water ice using shock compression", Nature Physics **14**, 297 (2018) 10.1038/s41567-017-0017-4 (cit. on p. 91).

[140] S. Stanley and J. Bloxham, "Numerical dynamo models of uranus' and neptune's magnetic fields", Icarus **184**, 556 (2006) https://doi.org/10.1016/j.icarus.2006.05.005 (cit. on p. 91).

[141] F. Grasselli and S. Baroni, "Topological quantization and gauge invariance of charge transport in liquid insulators", Nature Physics **15**, 967 (2019) https://doi.org/10.1038/s41567-019-0562-0 (cit. on p. 91).

[142] L. Zhang, H. Wang, M. C. Muniz, A. Z. Panagiotopoulos, R. Car, and W. E, *A deep potential model with long-range electrostatic interactions*, 2021 (cit. on p. 94).

[143] N. Marzari, A. A. Mostofi, J. R. Yates, I. Souza, and D. Vanderbilt, "Maximally localized wannier functions: theory and applications", Rev. Mod. Phys. **84**, 1419 (2012) 10.1103/RevModPhys.84.1419 (cit. on p. 94).

[144] N. Marzari and D. Vanderbilt, "Maximally localized generalized wannier functions for composite energy bands", Phys. Rev. B **56**, 12847 (1997) 10.1103/PhysRevB.56.12847 (cit. on p. 94).

[145] E. Prodan and W. Kohn, "Nearsightedness of electronic matter", Proceedings of the National Academy of Sciences **102**, 11635 (2005) 10.1073/pnas.0505436102 (cit. on p. 94).

[146] W. Kohn, "Density functional and density matrix method scaling linearly with the number of atoms", Phys. Rev. Lett. **76**, 3168 (1996) 10.1103/PhysRevLett.76.3168 (cit. on p. 94).

[147] P. P. Ewald, "Die Berechnung optischer und elektrostatischer Gitterpotentiale", Annales Physics **369**, 253 (1921) 10.1002/andp.19213690304 (cit. on p. 94).

[148] T. Darden, D. York, and L. Pedersen, "Particle mesh ewald: an $N \cdot \log(N)$ method for ewald sums in large systems", The Journal of Chemical Physics **98**, 10089 (1993) 10.1063/1.464397 (cit. on p. 94).

[149] R. W. Hockney and J. W. Eastwood, *Computer simulation using particles* (Taylor & Francis, Inc., USA, 1988) (cit. on p. 94).

[150] U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee, and L. G. Pedersen, "A smooth particle mesh ewald method", The Journal of Chemical Physics **103**, 8577 (1995) 10.1063/1.470117 (cit. on p. 94).

[151] M. Deserno and C. Holm, "How to mesh up ewald sums. i. a theoretical and numerical comparison of various particle mesh routines", The Journal of Chemical Physics **109**, 7678 (1998) 10.1063/1.477414 (cit. on p. 94).

[152] A. A. Mostofi, J. R. Yates, Y.-S. Lee, I. Souza, D. Vanderbilt, and N. Marzari, "Wannier90: a tool for obtaining maximally-localised wannier functions", Computer Physics Communications **178**, 685 (2008) https://doi.org/10.1016/j.cpc.2007.11.016 (cit. on p. 96).

[153] A. A. Mostofi, J. R. Yates, G. Pizzi, Y.-S. Lee, I. Souza, D. Vanderbilt, and N. Marzari, "An updated version of wannier90: a tool for obtaining maximally-localised wannier functions", Computer Physics Communications **185**, 2309 (2014) https://doi.org/10.1016/j.cpc.2014.05.003 (cit. on p. 96).

[154] G. Pizzi, V. Vitale, R. Arita, S. Blügel, F. Freimuth, G. Géranton, M. Gibertini, D. Gresch, C. Johnson, T. Koretsune, J. Ibañez-Azpiroz, H. Lee, J.-M. Lihm, D. Marchand, A. Marrazzo, Y. Mokrousov, J. I. Mustafa, Y. Nohara, Y. Nomura, L. Paulatto, S. Poncé, T. Ponweiser, J. Qiao, F. Thöle, S. S. Tsirkin, M. Wierzbowska, N. Marzari, D. Vanderbilt, I. Souza, A. A. Mostofi, and J. R. Yates, "Wannier90 as a community code: new features and applications", Journal of Physics: Condensed Matter **32**, 165902 (2020) 10.1088/1361-648x/ab51ff (cit. on p. 96).

[155] A. Damle, L. Lin, and L. Ying, "Compressed representation of kohn–sham orbitals via selected columns of the density matrix", Journal of Chemical Theory and Computation **11**, 1463 (2015) 10.1021/ct500985f (cit. on p. 96).

[156] A. Damle and L. Lin, "Disentanglement via entanglement: a unified method for wannier localization", Multiscale Modeling & Simulation **16**, 1392 (2018) 10.1137/17M1129696 (cit. on p. 96).

[157] L. Xu, P. Kumar, S. V. Buldyrev, S.-H. Chen, P. H. Poole, F. Sciortino, and H. E. Stanley, "Relation between the widom line and the dynamic crossover in systems with a liquid-liquid phase transition", Proceedings of the National Academy of Sciences **102**, 16558 (2005) 10.1073/pnas.0507870102 (cit. on p. 99).