

Università degli Studi di Trieste
Dipartimento di Fisica
Corso di Laurea Magistrale in Fisica della Materia Condensata

TEORIA E SIMULAZIONE
MOLECOLARE DI FLUIDI
VISCOSI CON
APPLICAZIONE
ALL'ACQUA

Laureando:
Cesare Malosso

Relatore:
Prof. Stefano Baroni

Matricola:
SM2300418

Anno accademico 2019/2020

CESARE MALOSSO

THEORY AND MOLECULAR SIMULATION
OF VISCOUS FLUIDS WITH
APPLICATIONS TO WATER



UNITS - UNIVERSITY OF TRIESTE
Piazzale Europa 1, 34127 Trieste, Italy

SISSA - SCUOLA INTERNAZIONALE SUPERIORE DI STUDI AVANZATI
Via Bonomea 265, 34136 Trieste, Italy

TITLEBACK:

This document was written with \LaTeX on Mac using *ArsClassica*, a reworking of the *ClassicThesis* style designed by André Miede, inspired to the masterpiece *The Elements of Typographic Style* by Robert Bringhurst.

CONTACTS:

cesare.malosso@gmail.com

*Par l'espace, l'univers me comprend et m'engloutit comme un point
par la pensée, je le comprends*

— Blaise Pascal

RINGRAZIAMENTI

Desidero ringraziare la mia famiglia per avermi dato l'opportunità di studiare, coltivare i miei interessi e le mie passioni, non facendomi mai mancare nulla. Un ringraziamento speciale va mia sorella Beatrice.

Vorrei ringraziare il Professor Stefano Baroni, relatore di questa tesi, fonte inesauribile di conoscenza e vulcanica di idee. La sua disponibilità e il suo entusiasmo mi hanno accompagnato durante l'intero progetto e nella comprensione degli argomenti trattati. Ringrazio lo studente di dottorato Davide Tisi per i consigli e il fondamentale aiuto che ha saputo darmi.

Infine vorrei ringraziare i miei amici e, in particolare, i miei coinquilini con i quali ho trascorso questi ultimi anni d'università.

SOMMARIO

La *viscosità di taglio* η , più comunemente chiamata viscosità, è un coefficiente di trasporto che descrive come un materiale resiste allo scorrimento. Dal punto di vista microscopico è legata all'attrito tra le molecole del materiale. La sua computazione numerica richiede simulazioni di dinamica molecolare molto lunghe al fine di ottenere un valore attendibile. Questo problema, ha posto un limite per molti anni sul calcolo di questo coefficiente, in particolare a partire da simulazione *ab initio* per via del grande costo computazionale che queste comportano. Recentemente, moderne tecniche di analisi dei dati hanno permesso il calcolo di coefficienti di trasporto, come la conducibilità termica, attraverso simulazioni relativamente corte, dell'ordine del centinaio di picosecondi.

I recenti sviluppi nell'ambito del *machine learning* e delle *reti neurali* hanno aperto una nuova frontiera in fisica della materia condensata, in particolare per quanto riguarda la dinamica molecolare. In particolare l'uso di cosiddetti "potenziali neural-network" per la modellizzazione delle interazioni tra le particelle sembra fornire un'alternativa interessante alla dinamica molecolare *ab initio*. Infatti, una volta allenati a partire da dati *ab initio*, i potenziali neural-network sono in grado di predire le forze agenti sugli ioni con un risultato indistinguibile da quello *ab initio*.

In questa tesi viene, per la prima volta, calcolata la viscosità dell'acqua a partire da simulazione *ab initio* e, successivamente, a partire da "simulazioni neural-network" dove la dinamica degli ioni è predetta da complesse reti neurali. A partire da queste simulazioni di dinamica molecolare, la viscosità viene calcolata mediante la teoria della risposta lineare di Green-Kubo, insieme alle più recenti tecniche di analisi di dati, in particolare la cosiddetta *analisi cepstrale*.

ABSTRACT

The *shear viscosity* η is a transport coefficient that describes how a material resists to deformation. Microscopically, it is related to the internal frictional forces that arises between the molecules of the material. The numerical computation of this coefficients needs very long molecular dynamics simulations in order to provide reliable values. For many years the evaluation of the shear viscosity has been a formidable task, almost unaffordable starting from *ab initio* molecular dynamics simulations, due to its high computational cost. Recently, modern data analysis techniques had provided new tools for the calculation of transport coefficients, as the thermal conductivity, with relative short molecular dynamics simulations.

The recent developments in *machine learnign* and *neural networks* provided new tools within condensed matter physics, in particular for what concern molecular dynamics. In particular, the use of neural-network potentials for

the modelization of the interactions between particles seems to provide a interesting alternative to the ab initio molecular dynamics. Indeed, once adequately trained with ab initio data, the neural-network potentials are able to predict the forces acting on the ions with the accuracy of ab initio calculations.

In this thesis we first evaluate the shear viscosity of liquid water from ab initio molecular dynamics simulations. Thus, we discuss the shear viscosity's results of "neural-network simulations" where the dynamics of the ions is given by complex neural networks. Starting from the molecular dynamics simulations, the shear viscosity evaluation is carried out through the Green-Kubo linear response theory, together with the most recent data analysis techniques, in particular the so-called *cepstral analysis*.

CONTENTS

	INTRODUCTION	XI
1	THEORETICAL FOUNDATIONS	1
1.1	Hydrodynamic variables	1
1.1.1	Extensive variables	1
1.1.2	Local thermal equilibrium	2
1.1.3	Phenomenological equations	3
1.2	Linear-Response Theory	5
1.3	Momentum transport	7
1.4	Microscopic expression of the stress tensor	8
2	DATA ANALYSIS	11
2.1	Elements of statistics	12
2.1.1	Stochastic processes	12
2.1.2	The Wiener-Khintchine Theorem	13
2.2	Cepstral analysis	16
2.3	Applications to classical simulations	21
2.3.1	The TIP4P/2005 model of water	22
2.3.2	Validation of the technique	23
2.3.3	Size scaling	26
3	AB INITIO MOLECULAR DYNAMICS	31
3.1	Born-Oppenheimer approximation	31
3.2	Density-Functional theory	33
3.2.1	Kohn-Sham equations	34
3.2.2	Stress tensor in DFT	35
3.3	Car-Parrinello Molecular Dynamics	37
3.4	Shear viscosity	38
3.4.1	Results and data analysis	40
4	NEURAL NETWORK APPROACH TO MOLECULAR DYNAMICS	45
4.1	Machine Learning Basics	45
4.1.1	Example: Linear Regression	46
4.2	Deep feedforward networks	47
4.2.1	Stochastic gradient descent	49
4.2.2	Back-propagation algorithm	50
4.3	Deep Potential - Smooth Edition	52
4.4	Simulations	55
4.4.1	Validation of the neural network potential	57
4.5	Shear viscosity results	62
5	CONCLUSIONS	67
	BIBLIOGRAPHY	69

INTRODUCTION

Motivation

The numerical evaluation of transport coefficients, such as thermal conductivity and shear viscosity, in extended systems from *Ab Initio Molecular Dynamics* (AIMD) simulations has been considered, until recently, a formidable task. It is in fact well known that in order to estimate transport coefficients with satisfactory accuracy, very long simulations of order of tens nanoseconds are required. With modern empirical potentials, used to model the inter-atomic interactions, it is still computationally possible to simulate long trajectories, but their limited transferability and the lack of systematic techniques for their derivation set a limit on their use. On the other hand, AIMD simulations are computationally very demanding and even on the most advanced platforms ab initio simulations are limited to hundreds of picoseconds and few hundreds atoms.

Recently, approaches based on Machine Learning (ML) [Deringer & Csányi, 2017; Smith et al., 2017] and Neural Network (NN) [Behler & Parrinello, 2007; Zhang, Han, Wang, Car, et al., 2018] are growing in popularity. Machine learning or neural network potentials, after a proper training on accurate quantum mechanical data, can faithfully reproduce the ab initio results, with a computational cost comparable with that of classical simulations. With this techniques, "near ab initio" calculation of transport coefficients is possible, but a verification with AIMD is still needed in order to validate the results.

In the last few years, a new approach [Baroni et al., 2020; Ercole et al., 2017] was developed to avoid impractically long simulations, opening the way to the quantum simulation of transport coefficients. This new technique has already provided reasonably good results for thermal conductivity of liquid systems with relatively short molecular dynamics trajectories (about 100 ps) achievable with ab initio calculations. So far, there have been no attempts to apply this procedure for the evaluation of the shear viscosity coefficient.

This study aims to contribute to this growing area of research by exploring the calculation of shear viscosity of liquid water from ab initio molecular dynamics simulations through both AIMD and neural-network techniques. Among the many NN models, in the following work we will employ the DeepMD-Sc [Zhang, Han, Wang, Car, et al., 2018] model.

Shear viscosity

We now give a brief introduction on the notion of viscosity and momentum transport, following as reference [Ortiz de Zárate & Sengers, 2006].

Let us consider a one-component fluid with mass density $\rho(\mathbf{r}, t)$ and local velocity $\mathbf{v}(\mathbf{r}, t)$ at the point \mathbf{r} and at the time t . A generalized form of Newton's equation of motions reads:

$$\rho(\mathbf{r}, t) \frac{d}{dt} \mathbf{v}(\mathbf{r}, t) = -\nabla \cdot \mathbf{P}(\mathbf{r}, t) + \mathbf{f}_{\text{ext}}(\mathbf{r}, t) \quad (0.1)$$

where $\mathbf{f}_{\text{ext}}(\mathbf{r}, t)$ is the force density, which expresses the amount of external force per unit volume, and $\mathbf{P}(\mathbf{r}, t)$ is the *local pressure tensor*. The latter can be decomposed in

$$\mathbf{P}(\mathbf{r}, t) = p(\mathbf{r}, t) \mathbb{I} - \boldsymbol{\tau}(\mathbf{r}, t) \quad (0.2)$$

where $p(\mathbf{r}, t)$ is the local (hydrostatic) pressure and $\boldsymbol{\tau}(\mathbf{r}, t)$ is a symmetric tensor, usually called local deviatoric stress tensor or simply local stress tensor.

Eq. (0.1) can be written in the form¹

$$\frac{\partial}{\partial t} (\rho \mathbf{v}) = -\nabla \cdot [(\rho \mathbf{v}) \mathbf{v} - \boldsymbol{\tau}] + \mathbf{f}_{\text{ext}} - \nabla p \quad (0.3)$$

which is the the ordinary equation of motion of a fluid.

The expression of the deviatoric stress is given by the phenomenological Newton's viscosity law

$$(\boldsymbol{\tau})_{ij} = \tau_{ij} = \eta \left(\frac{\partial v_i}{\partial r_j} + \frac{\partial v_j}{\partial r_i} \right) - \left(\frac{2}{3} \eta - \eta_V \right) \delta_{ij} \frac{\partial v_l}{\partial r_l} \quad (0.4)$$

where η_V and η are two independent coefficients known respectively as *bulk viscosity* and *shear viscosity*. The first one represents extra forces required to compress volume elements, while the latter represents shearing forces between adjacent fluid layers.

Next, substituting Newton's viscosity law (0.4) into Eq. (0.3) and using the continuity equation for mass:

$$\frac{\partial}{\partial t} \rho(\mathbf{r}, t) = -\nabla \cdot (\rho(\mathbf{r}, t) \mathbf{v}(\mathbf{r}, t)) \quad (0.5)$$

one can obtain the *Navier-Stokes equation*:

$$\rho \left[\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} \right] = -\nabla p + \eta \nabla^2 \mathbf{v} + \left(\frac{1}{3} \eta + \eta_V \right) \nabla (\nabla \cdot \mathbf{v}) + \mathbf{f}_{\text{ext}} \quad (0.6)$$

which is one of the three hydrodynamic equations: the partial differential equations that determine the motion of a fluid. This equation presents a non-linear term $(\mathbf{v} \cdot \nabla) \mathbf{v}$ that leads to such complicated phenomena as turbulence, which lies outside the scope of this thesis.

The bulk viscosity and the shear viscosity, together with other coefficients such as the thermal conductivity, the diffusivity, etc., are generically referred to as *transport coefficients*. They are related to the *phenomenological coefficients*

¹ Here $\left[(\rho \mathbf{v}) \mathbf{v} \right]_{ij} = \rho v_i v_j$ is a second-order symmetric tensor.

(or *Onsager coefficients*) appearing in the so-called *phenomenological equations* [de Groot & Mazur, 1962; Kubo et al., 1991]. Non-equilibrium statistical physics provides a theoretical justification and, together with linear response theory [Kubo, 1957], offers some powerful tools to evaluate these coefficients with molecular dynamics simulations. All this theory end up with the *Green-Kubo relations* [1957] that give the exact mathematical expression for transport coefficients in terms of integrals of time-correlation functions of proper currents.

In particular for the shear viscosity, defining the *macroscopic stress tensor* as $\Pi(t) = \frac{1}{V} \int_V \tau(\mathbf{r}, t) d\mathbf{r}$, the equation reads [Kubo et al., 1991]

$$\eta = \frac{V}{k_B T} \int_0^\infty \langle \Pi^{\text{off}}(t) \Pi^{\text{off}}(0) \rangle_{\text{eq}} dt \quad (0.7)$$

where V and T are respectively the volume and the temperature of the system, while k_B is the Boltzmann constant. The label *off* indicates one of the three equivalent off-diagonal component of the macroscopic stress tensor ($\Pi_{xy}, \Pi_{xz}, \Pi_{yz}$) and the brackets $\langle \cdot \rangle_{\text{eq}}$ denote the ensemble average at equilibrium.²

Eq. (0.7) relates the shear viscosity to the integral of the time-correlation function of the off-diagonal elements of the stress tensor. Molecular dynamics allows to evaluate ensemble averages through the *ergodic hypothesis* [Frenkel & Smit, 2001] and finally to evaluate transport coefficients.

Outline

In this thesis we will evaluate the shear viscosity of water systems, exploiting the Green-Kubo relations. Molecular dynamics simulations are needed in order to deal with the ensemble average appearing in Eq. (0.7). Different approaches for the evaluation of the inter-atomic forces will be considered: the first based on empirical potentials, the second with quantum mechanical calculations based on Density-Functional theory and the last carried out using neural network potentials.

This thesis is structured as follows:

FIRST CHAPTER: The theory of transport will be reviewed, starting from the hydrodynamic theory, going through the linear response theory and finally ending up with the Green-Kubo formula for the shear viscosity; We will provide the microscopic expression for stress tensor both for classical and quantum systems.

SECOND CHAPTER: Starting from a brief introduction to some topics in statistics, we present the main techniques used to analyze the data obtained from the numerical simulations; a first application to an empirical water model is discussed in order to validate the techniques.

THIRD CHAPTER: After a brief introduction of Density-Functional theory, the Born-Oppenheimer approximation and the Car-Parrinello approach to molecular dynamics, we carry out the calculation of the shear viscosity of *ab initio* water.

² Since the liquid is isotropic, the off-diagonal component of the stress tensor are equivalent.

FOURTH CHAPTER: We give a basic overview on machine learning and we present the DeepMD-SE model for modelling inter-atomic. We end up showing the results of the numerical simulations.

The theory of hydrodynamic fluctuations [Forster, 2018] describes the momentum transport in classical fluids. The phenomenological theory of Onsager [Onsager, 1931a] provides the physical foundations, while the linear-response theory [Green, 1952; Kubo, 1957; Kubo et al., 1957] gives the mathematical formalization. The Green-Kubo formula relates the shear viscosity η , a non-equilibrium dissipative quantity, to the time integrated equilibrium auto-correlations of the off-diagonal elements of the macroscopic stress tensor.

1.1 HYDRODYNAMIC VARIABLES

1.1.1 Extensive variables

Extensive variables, such as energy or entropy, play a fundamental role in condensed matter physics. By definition, the values that these quantities assume for a system are equal to the sum of the values they has for any of its partitions into non-overlapping subsystems. This property allows to express an extensive variable, A , as the integral of a suitably defined density, $a(\mathbf{r})$, as:

$$A(V) = \int_V a(\mathbf{r}) d\mathbf{r}, \quad (1.1)$$

where V is the volume of the system. Equation (1.1) is not only the definition of an extensive variable but it also provides a mathematical definition to the density $a(\mathbf{r})$ at point \mathbf{r} of the space.

When an extensive quantity is locally conserved, i.e. there are neither sources nor sinks, a *continuity equation* can be established. Introducing a current density $\mathbf{j}(\mathbf{r}, t)$ associated to the extensive quantity considered, the continuity equation reads:

$$\frac{\partial a(\mathbf{r}, t)}{\partial t} = -\nabla \cdot \mathbf{j}(\mathbf{r}, t), \quad (1.2)$$

where $\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right)$ is the gradient vector and the dot \cdot is the usual scalar product. In the following, densities and current densities of conserved quantities will be referred as respectively *conserved densities* and *conserved currents* for short.

The continuity equation (1.2) fulfilled by the conserved currents and densities has far-reaching consequences on the dynamics of these variables. By Fourier transforming in space Eq. (1.2) one obtains

$$\dot{\tilde{a}}(\mathbf{q}, t) = -i\mathbf{q} \cdot \tilde{\mathbf{j}}(\mathbf{q}, t), \quad (1.3)$$

where the overdot indicates a time derivative, and the tilde the Fourier transform.

Eq. (1.3) states that the longer is the wavelength $\lambda \propto |\mathbf{q}|^{-1}$ of the Fourier mode of wave-vector \mathbf{q} , the slower is the dynamics of the corresponding Fourier component of the conserved density, $\tilde{a}(\mathbf{q}, t)$. We conclude that for long enough wavelengths, conserved densities are adiabatically decoupled from all the other fast atomic degrees of freedom.

The long-wavelength Fourier components of conserved densities are called *hydrodynamic variables*.

1.1.2 Local thermal equilibrium

Let S be the entropy of the system, which is a function of extensive conserved quantities:

$$S = S(E, V, N_1, N_2, \dots), \quad (1.4)$$

where E is the energy of the system, V is the volume and N_i is the number of particles of the species i . The corresponding conjugate intensive variables are temperature, pressure and the chemical potentials:

$$\begin{aligned} \left. \frac{\partial S}{\partial E} \right|_{V, N_i} &= \frac{1}{T}, \\ \left. \frac{\partial S}{\partial V} \right|_{E, N_i} &= \frac{p}{T}, \\ \left. \frac{\partial S}{\partial N_i} \right|_{V, E} &= -\frac{\mu_i}{T}. \end{aligned} \quad (1.5)$$

A system is said to be in thermodynamic equilibrium if all its macroscopic properties are stationary with respect to time. As a consequence of the principle of maximum entropy, one can easily prove that intensive thermodynamic properties (i.e. temperature, pressure and chemical potential) have uniform values along the system, regardless of the size of it.

Indeed, let $S(V_1, A_1)$ and let $S(V_2, A_2)$ be the entropies of subsystems 1 and 2, as functions of one of their respective extensive variable, A , and let $\alpha = \frac{\partial S}{\partial A}$ be the corresponding intensive variable. At equilibrium, the total entropy must be stationary with respect to the variation of A for one of its subsystems, say A_1 , with the constraint that $A_1 + A_2 = A$, namely:

$$\begin{aligned} \frac{\partial S(V, A, \dots)}{\partial A_1} &= \frac{\partial}{\partial A_1} [S(V_1, A_1) + S(V_2, A_2)] \\ &= \frac{\partial}{\partial A_1} [S(V_1, A_1) + S(V_2, A - A_1)] \\ &= \frac{\partial S(V_1, A_1)}{\partial A_1} - \frac{\partial S(V_2, A_2)}{\partial A_2} \\ &= \alpha_1 - \alpha_2 \\ &= 0, \end{aligned} \quad (1.6)$$

and we conclude that $\alpha_1 = \alpha_2$. In particular, the very possibility of defining an intensive variable is itself a consequence of equilibrium, which in turns implies the constancy of intensive variables across the system.

To extend these concepts to systems which are not in thermodynamic equilibrium (i.e. systems with in-homogeneous temperature, pressure, or chemical potential) one adopts the principle of *local thermodynamic states*. Specifically, it is assumed that one can imagine subsystems that are small with

respect to the macroscopic scale, but that still contain a sufficiently large number of molecules, so that one can define for them local values of the thermodynamic properties by the methods of equilibrium statistical physics. Thus, the total system can be seen as a continuum of local thermodynamic states at equilibrium with thermodynamic properties that now depend on the position \mathbf{r} and the time t , such as $\rho(\mathbf{r}, t)$, $T(\mathbf{r}, t)$, $p(\mathbf{r}, t)$, etc. Thermodynamic quantities are replaced by corresponding thermodynamic fields that are continuous functions of space and time. This assumption goes under the name of *local thermal equilibrium*.

1.1.3 Phenomenological equations

Let us consider a system with a number Q of conserved extensive variables $\{A^1, A^2, \dots, A^Q\}$. For instance, in the case of a one-component fluid $Q = 5$ corresponding to mass (i.e. particle number), energy and the three components of the momentum. In order to simplify the notation, we set the values of the conserved quantities equal to zero, $\langle A^i \rangle = 0$, so that their densities, $a^i(\mathbf{r}, t)$, directly describe the departure from equilibrium, and we indicate by $\mathbf{j}^i(\mathbf{r}, t)$ 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.

Restricting to the case of only one conserved quantity, for small enough deviations from equilibrium the time derivatives of conserved densities are linear combinations of the density fluctuations themselves

$$\dot{a}(\mathbf{r}, t) = \int \Lambda(\mathbf{r} - \mathbf{r}', t - t') a(\mathbf{r}', t') d\mathbf{r}' dt' \quad (1.7)$$

where $a(\mathbf{r}, t)$ is already the deviation from equilibrium of the conserved density and Λ is a suitable coefficient that depends only over $\mathbf{r} - \mathbf{r}'$ 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. (1.7) reads:

$$-i\omega \tilde{a}(\mathbf{q}, \omega) = \tilde{\Lambda}(\mathbf{q}, \omega) \tilde{a}(\mathbf{q}, \omega), \quad (1.8)$$

where $\tilde{\Lambda}$ is the space-time Fourier transform Λ .

The generalization of (1.8) for a system with Q conserved extensive variables reads:

$$-i\omega \tilde{a}^l(\mathbf{q}, \omega) = \sum_{j=1}^Q \tilde{\Lambda}^{lj}(\mathbf{q}, \omega) \tilde{a}^j(\mathbf{q}, \omega). \quad (1.9)$$

By combining the last set of equations with the time Fourier transform of Eq. (1.3), one obtains the *constitutive equations*:

$$-i\mathbf{q} \cdot \tilde{\mathbf{j}}^l(\mathbf{q}, \omega) = \sum_{j=1}^Q \tilde{\Lambda}^{lj}(\mathbf{q}, \omega) \tilde{a}^j(\mathbf{q}, \omega), \quad (1.10)$$

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

$$\tilde{\mathbf{j}}^l(\mathbf{q}, \omega) = i \frac{\mathbf{q}}{q^2} \sum_{j=1}^Q \tilde{\Lambda}^{lj}(\mathbf{q}, \omega) \tilde{a}^j(\mathbf{q}, \omega). \quad (1.11)$$

In isotropic media, the $\tilde{\Lambda}$'s are functions of $|\mathbf{q}|$, whereas their value at $|\mathbf{q}| = 0$ vanishes. In fact a non-vanishing value would imply a long-range dependence of the currents on density fluctuations, in contrast with our assumption of local thermodynamic equilibrium. Thus, the long-wavelength low-frequency limit of the coupling constants can thus be assumed to be

$$\tilde{\Lambda}^{lj}(\mathbf{q}, \omega) \sim q^2 \tilde{\lambda}^{lj}. \quad (1.12)$$

Then inserting this last expression in Eq. (1.11) one obtains

$$\tilde{\mathbf{j}}^l(\mathbf{q}, \omega) = i\mathbf{q} \sum_{j=1}^Q \tilde{\lambda}^{lj} \tilde{\mathbf{a}}^j(\mathbf{q}, \omega), \quad (1.13)$$

valid in the long-wavelength low-frequency regime.

Defining the macroscopic conserved current \mathbf{J}^l as:

$$\begin{aligned} \mathbf{J}^l &= \frac{1}{V} \int_V \mathbf{j}^l(\mathbf{r}) \, d\mathbf{r} \\ &= \lim_{q \rightarrow 0} \tilde{\mathbf{j}}^l(\mathbf{q}, \omega = 0), \end{aligned} \quad (1.14)$$

and the corresponding components of the macroscopic density gradients \mathbf{D}^l as:

$$\begin{aligned} \mathbf{D}^l &= \frac{1}{V} \int_V \nabla \alpha^l(\mathbf{r}) \, d\mathbf{r} \\ &= \lim_{q \rightarrow 0} i\mathbf{q} \tilde{\mathbf{a}}^l(\mathbf{q}, \omega = 0), \end{aligned} \quad (1.15)$$

we can relate explicitly these last two quantities through Eq. (1.13) and obtain:

$$\mathbf{J}^l = \sum_{j=1}^Q \tilde{\lambda}^{lj} \mathbf{D}^j. \quad (1.16)$$

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

$$\mathbf{F}^l = \frac{1}{V} \int_V \nabla \alpha^l(\mathbf{r}) \, d\mathbf{r}. \quad (1.17)$$

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

$$\mathbf{D}^l = \sum_{j=1}^Q \chi^{lj} \mathbf{F}^j, \quad (1.18)$$

and inserting this relation into Eq. (1.16), one gets the *phenomenological equations*:

$$\mathbf{J}^l = \sum_{j=1}^Q \mathbf{L}^{lj} \mathbf{F}^j, \quad (1.19)$$

where

$$L^{lj} = \sum_{k=0}^Q \lambda^{lk} \chi^{kj}, \quad (1.20)$$

are the *Onsager's coefficients* (or *phenomenological coefficients*).

Eq. (1.19) equation states a linear relation between the macroscopic conserved currents J 's and the thermodynamic forces F 's.

In particular the Onsager's coefficients satisfy the so-called Onsager's reciprocal relations [Onsager, 1931a; 1931b]

$$L^{lj} = L^{jl}, \quad (1.21)$$

that shows the equality of the cross-coefficients.

1.2 LINEAR-RESPONSE THEORY

In order to evaluate the phenomenological coefficients appearing in Eq. (1.19), we need linear-response theory. Let us consider a classical system of N interacting atoms described by the Hamiltonian

$$H^0(\Gamma) = \sum_{i=1}^N \frac{1}{2M_n} \mathbf{P}_n^2 + \Phi(\mathbf{R}_1, \mathbf{R}_2, \dots, \mathbf{R}_N), \quad (1.22)$$

where M_i , \mathbf{P}_n and \mathbf{R}_n are respectively the mass, the momentum and the position of the n -th particle, while $\Gamma = \{\mathbf{R}_n, \mathbf{P}_n\}$ 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 an external potential that couples to a linear combination of conserved densities, $\{a^l(\mathbf{r}; \Gamma)\}$, as:

$$\Phi'(\Gamma, t) = \sum_l \int_V v^l(\mathbf{r}, t) a^l(\mathbf{r}; \Gamma) d\mathbf{r}, \quad (1.23)$$

where $\{v^l(\mathbf{r}, t)\}$ are time-dependent fields that couple to the conserved densities, and $\{a^l(\mathbf{r}; \Gamma)\}$ are space-phase observables whose ensemble average is the conserved density

$$\begin{aligned} a^l(\mathbf{r}) &= \langle a^l(\mathbf{r}, \Gamma) \rangle_{eq} \\ &= \int a^l(\mathbf{r}; \Gamma) \mathcal{P}^0(\Gamma) d\Gamma, \end{aligned} \quad (1.24)$$

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

$$\mathcal{P}^0(\Gamma) \propto e^{-\frac{H^0(\Gamma)}{k_B T}}, \quad (1.25)$$

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

$$j^l(\mathbf{r}) = \langle j^l(\mathbf{r}, \Gamma) \rangle_{eq}. \quad (1.26)$$

Now, let $\Gamma'_t = \{\mathbf{R}_n(t), \mathbf{P}_n(t)\}$ be a dynamical trajectory of the system, driven by the perturbed hamiltonian $H'(t) = H^0 + \Phi'(t)$, through the well-known Hamilton equations of motions:

$$\begin{aligned}\dot{\mathbf{R}}_n(t) &= \left. \frac{\partial H'}{\partial \mathbf{P}_n} \right|_t, \\ \dot{\mathbf{P}}_n(t) &= - \left. \frac{\partial H'}{\partial \mathbf{R}_n} \right|_t,\end{aligned}\tag{1.27}$$

that 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, and on the initial conditions of the system. Averaging with respect to the initial conditions will result in a time-dependent expectation value for the conserved densities (or currents)

$$\begin{aligned}a^l(\mathbf{r}, t) &= \langle a^l(\mathbf{r}, \Gamma) \rangle \\ &= \int a^l(\mathbf{r}; \Gamma) \mathcal{P}^t(\Gamma) d\Gamma \\ &= \int a^l(\mathbf{r}; \Gamma_t) \mathcal{P}^0(\Gamma_0) d\Gamma_0 \\ &= \langle a^l(\mathbf{r}, \Gamma_t) \rangle_{eq},\end{aligned}\tag{1.28}$$

where the ensemble average over the distribution function $\mathcal{P}^t \propto e^{-\frac{H'(t)}{k_B T}}$ given by the hamiltonian H' was written as an average over the initial conditions weighted with \mathcal{P}^0 , the distribution function given by the unperturbed hamiltonian.

In accord to Green-Kubo theory [Green, 1952; Kubo, 1957] the linear response to a perturbation of the α component of a conserved current j^l is

$$\begin{aligned}j_\alpha^l(\mathbf{r}, t) &= \frac{1}{k_B T} \sum_j \int_{-\infty}^t dt' \int_V d\mathbf{r}' \langle j_\alpha^l(\mathbf{r}, \Gamma_t) \dot{a}^j(\mathbf{r}', \Gamma_{t'}) \rangle_{eq} v^j(\mathbf{r}', t') \\ &= -\frac{1}{k_B T} \sum_j \sum_\beta \int_{-\infty}^t dt' \int_V d\mathbf{r}' \langle j_\alpha^l(\mathbf{r}, \Gamma_t) \frac{\partial}{\partial r'_\beta} j_\beta^j(\mathbf{r}', \Gamma_{t'}) \rangle_{eq} v^j(\mathbf{r}', t') \\ &= \frac{1}{k_B T} \sum_j \sum_\beta \int_{-\infty}^t dt' \int_V d\mathbf{r}' \langle j_\alpha^l(\mathbf{r}, \Gamma_t) j_\beta^j(\mathbf{r}', \Gamma_{t'}) \rangle_{eq} \frac{\partial}{\partial r'_\beta} v^j(\mathbf{r}', t'),\end{aligned}\tag{1.29}$$

where r_β is the β component of the position vector, and where the continuity equation and the integration by parts have been used. Assuming space-time homogeneity as well as isotropy, last equation can be averaged to recover the macroscopic flux as in eq. (1.19) with

$$J_\alpha^l = \frac{1}{V} \int_V j_\alpha^l(\mathbf{r}) d\mathbf{r},\tag{1.30}$$

$$F_\alpha^l = \frac{1}{VT} \int \int \frac{\partial}{\partial r_\alpha} v^j(\mathbf{r}, t) d\mathbf{r} dt,\tag{1.31}$$

$$L_{\alpha\beta}^{lj} = \frac{V}{k_B} \int_0^\infty \langle J_\alpha^l(\Gamma_t) J_\beta^j(\Gamma_0) \rangle_{eq} dt.\tag{1.32}$$

This completes the derivation of the Green-Kubo formula for transport coefficients from classical linear-response theory, providing the tools in order to evaluate the Onsager's coefficients from equilibrium molecular dynamics.

1.3 MOMENTUM TRANSPORT

The continuity equation for the momentum density $\mathbf{g}(\mathbf{r}, t)$, in the absence of external forces, reads

$$\frac{\partial}{\partial t} \mathbf{g}(\mathbf{r}, t) = -\nabla \cdot \mathbf{P}(\mathbf{r}, t), \quad (1.33)$$

where $\mathbf{P}(\mathbf{r}, t) = p(\mathbf{r}, t) \mathbb{I} - \boldsymbol{\tau}(\mathbf{r}, t)$ is the *local pressure tensor* and it is a second-order tensor. It can be split in two contributions: the local *hydrostatic pressure* and the symmetric *local stress tensor*.

The phenomenology of momentum transport in an isotropic liquid system is described by the Newton's viscosity law

$$(\boldsymbol{\tau})_{ij} = \tau_{ij} = \eta \left(\frac{\partial v_i}{\partial r_j} + \frac{\partial v_j}{\partial r_i} \right) + \left(\eta_V - \frac{2}{3} \eta \right) \delta_{ij} \frac{\partial v_m}{\partial r_m}, \quad (1.34)$$

where η is the *shear viscosity* coefficient, while η_V is the *bulk viscosity* coefficient. For general system, under the assumption of local thermal equilibrium, Eq. (1.34) reads [de Groot & Mazur, 1954]

$$\tau_{ij}(\mathbf{r}, t) = \sum_{n,m} \eta_{ij, nm}(\mathbf{r}, t) \frac{\partial v_n(\mathbf{r}, t)}{\partial r_m}, \quad (1.35)$$

where $\eta_{ij, nm}$ is a fourth-order tensor: the *viscosity tensor*. This is the most general expression for the viscous stress tensor including anisotropic as well as isotropic systems for Newtonian fluids. For three-dimensional systems, the viscosity tensor has $3^4 = 81$ elements, however, since the stress tensor is symmetric, this number reduce to 36. Additionally, Onsager's reciprocal relations (1.21) further reduces the number of independent coefficients to 21. We are not giving more details, but it turns out that, because of other symmetries, for an isotropic fluid the number of independent phenomenological coefficients can be reduced to two: η and η_V , giving Eq. (1.34). We further assume that in practical application the space-time dependence of the phenomenological coefficients can be neglected.

Eq. (1.35) can be written in a macroscopic form, recasting to the form of the phenomenological equations (1.19), by averaging over the volume and assuming that the gradients of the velocity fields vary smoothly in space. We then obtain

$$\Pi_{ij} = \sum_{n,m} \eta_{ij, nm} \frac{\partial v_n}{\partial r_m}, \quad (1.36)$$

where $\boldsymbol{\Pi} = \frac{1}{V} \int_V \boldsymbol{\tau}(\mathbf{r}) d\mathbf{r}$ is the *macroscopic stress tensor*.

In the past section we presented how linear-response theory allows one to compute the linear response coefficients to a mechanical perturbation through the Green-Kubo equation (1.32). However a velocity gradient, as the one appearing in (1.36), is not due to any mechanical perturbation of

the hamiltonian. It is within the so-called *thermal perturbations*. Nevertheless, assuming the local thermal equilibrium hypotheses, a local value of the velocity vector field $\mathbf{v}(\mathbf{r}, t)$ can be defined providing us a way to treat this thermal perturbation as mechanical one.

When a field of non-uniform mass flow is present in a fluid, the local equilibrium phase-space distribution reads [Kubo et al., 1991]:

$$\mathcal{P}(\Gamma) = e^{-\frac{1}{k_B T} [H^0 - \int_V \mathbf{g}(\mathbf{r}, \Gamma) \cdot \mathbf{v}(\mathbf{r}, t) d\mathbf{r}]}, \quad (1.37)$$

where $\mathbf{g}(\mathbf{r}, t)$ is the momentum density and $\mathbf{v}(\mathbf{r})$ is the velocity vector field.

The analogous of the mechanical perturbation potential (1.23) is then given by

$$\Phi'(\Gamma) = \int_V \mathbf{g}(\mathbf{r}, \Gamma) \cdot \mathbf{v}(\mathbf{r}, t) d\mathbf{r}, \quad (1.38)$$

and we can write, in accord with (1.29), using the continuity equation (1.33)

$$\tau_{lj}(\mathbf{r}, t) = \frac{1}{k_B T} \sum_{n,m} \int_{-\infty}^t dt' \int_V d\mathbf{r}' \left\langle \tau_{lj}(\mathbf{r}, \Gamma_t) \tau_{nm}(\mathbf{r}', \Gamma_{t'}) \right\rangle_{eq} \frac{\partial}{\partial r'_m} v_n(\mathbf{r}', t'), \quad (1.39)$$

where, now, $v_n(\mathbf{r}, t)$ is one of the three components of the velocity vector field $\mathbf{v}(\mathbf{r}', t')$. Finally, in accord to Eqs. (1.30), (1.31) and (1.32) we can write

$$\eta_{lj, nm} = \frac{L_{lj, nm}}{T} = \frac{V}{k_B T} \int_0^\infty \left\langle \Pi_{ij}(\Gamma_t) \Pi_{nm}(\Gamma_0) \right\rangle_{eq} dt, \quad (1.40)$$

where Π_{ij} is the macroscopic stress tensor:

$$\Pi_{ij}(\Gamma) = \frac{1}{V} \int_V \tau_{ij}(\mathbf{r}, \Gamma) d\mathbf{r}. \quad (1.41)$$

For a isotropic liquid, Eq. (1.40) for the shear viscosity reads

$$\eta = \frac{V}{k_B T} \int_0^\infty \left\langle \Pi^{\text{off}}(\Gamma_t) \Pi^{\text{off}}(\Gamma_0) \right\rangle_{eq} dt, \quad (1.42)$$

where Π^{off} is equivalently one of the three off-diagonal elements of the stress tensor.

1.4 MICROSCOPIC EXPRESSION OF THE STRESS TENSOR

In order to evaluate the macroscopic stress tensor Π through molecular dynamic simulations, one needs to express it as a function of atomic quantities. A simple classical derivation can be obtained from statistical mechanics [Frenkel & Smit, 2001]. Considering a classical system made of N atoms of mass m_n in a volume V , for pairwise additive interactions the macroscopic stress is given by¹

$$\Pi_{ij} = -\frac{1}{V} \sum_{n=1}^N \left\langle \left(\frac{p_i^n p_j^n}{m_i} + \frac{1}{2} \sum_{m(\neq n)=1}^N r_i^{nm} f_j^{nm} \right) \right\rangle \quad (1.43)$$

¹ For a generalized expression of (1.43) compatible with many-body potentials and periodic boundary conditions, one can look at [Thompson et al., 2009].

where p_i^n is the i -th component of momentum of the n -th particle, r_i^{nm} is the i -th component of the distance vector between particle n and m and f_j^{nm} represents the j -th component of the force exerted on particle n by the particle m . The $\langle \cdot \rangle$ brackets denote the usual ensemble average.

An analogue expression of the macroscopic stress for a quantum system is given by [Nielsen & Martin, 1985a]

$$\Pi_{ij} = -\frac{1}{V} \sum_{n=1}^N \langle \Psi | \left(\frac{p_i^n p_j^n}{m_i} - \frac{1}{2} r_i^n \nabla_j^n U_{\text{int}} \right) | \Psi \rangle \quad (1.44)$$

where U_{int} is the interacting potential between the particles and r_i^n is the i -th component of the position vector of particle n . In this case the positions and the momenta of the particles has to be treated as quantum operators, and the expectation value is over the many-body wave-function of the system. Equation (1.44) is not in the most desirable form for application to large macroscopic systems because of the presence of the position coordinate of each particle. However, it is possible, in particular within Density-Functional theory, to derive a well-defined expression for the stress tensor useful in simulations with periodic boundary conditions [1985a; 1985b].

2 | DATA ANALYSIS

The aim of this chapter is to introduce and briefly discuss the specific techniques and the methods used in the present work in order to deal with the calculation of the shear viscosity transport coefficient. In the past chapter, we saw that the Green-Kubo formalism gives a direct relationship between the viscosity and the expectation value at equilibrium of the off-diagonal elements of the stress tensor. This relation reads:

$$\eta = \frac{V}{k_B T} \int_0^\infty \left\langle \Pi^{\text{off}}(\Gamma_t) \Pi^{\text{off}}(\Gamma_0) \right\rangle_{\text{eq}} dt, \quad (2.1)$$

where V is the volume of the system, k_B is the Boltzmann constant and T is the temperature of the system. The brackets $\langle \cdot \rangle$ denotes the *ensemble average*.

In order to deal with the evaluation of the ensemble average of a general observable, one can proceed with standard molecular dynamics techniques [Frenkel & Smit, 2001]. In practice, the ensemble average, that consists in averaging the observable over all the possible states of the system, is often not easily accessible. Instead one can perform, in most of experiments, a series of measurements during a certain time interval and then determine the average of these measurements. This averages are equivalent to the ensemble average if the system satisfies the so-called "*ergodic hypothesis*". The latter can be expressed by the following relation

$$\langle \hat{A} \rangle = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T \mathcal{A}(\Gamma(t)) dt, \quad (2.2)$$

where $\hat{A} \equiv \mathcal{A}(\Gamma)$ is a general observable that depends over the phase-space variables of the system Γ , while $\mathcal{A}(\Gamma(t))$ is a general observable evaluated along a precise trajectory $\Gamma(t)$ in phase-space.

Thus assuming the validity of the ergodic hypothesis, true at least for the liquid systems, one can calculate the viscosity coefficient in Eq. (2.1) from the time-series of the off-diagonal elements of the stress tensor obtained from molecular dynamics simulations. Then these time-series are directly connected to the viscosity through the relation

$$\eta = \frac{V}{k_B T} \int_0^\infty \lim_{T \rightarrow \infty} \frac{1}{T-t} \int_0^{T-t} \Pi^{\text{ij}}(\Gamma(t+\tau)) \Pi^{\text{ij}}(\Gamma(\tau)) d\tau dt, \quad (2.3)$$

where T is the time-length of the simulation. Since the correlation time of the currents (*i.e.* the time after that we can consider the correlation function to be zero) is usually a significantly smaller then the time-length of the simulation, one can write

$$\eta \approx \frac{V}{k_B T} \int_0^T \frac{1}{T} \int_0^{T-t} \Pi^{\text{off}}(\Gamma(t+\tau)) \Pi^{\text{off}}(\Gamma(\tau)) d\tau dt, \quad (2.4)$$

that is valid in the limit of *large-T*.

Within numerical simulations, the discretization of time leads to a discrete form of Eq. (2.4):

$$\eta \approx \frac{V}{k_B T} \sum_{n=1}^N \frac{\epsilon}{M} \sum_{m=1}^M \Pi^{\text{off}}(\Gamma(n+m)) \Pi^{\text{off}}(\Gamma(m)), \quad (2.5)$$

where $\Pi^{\text{off}}(\Gamma(m)) = \Pi^{\text{off}}(\Gamma(t_m)) = \Pi^{\text{off}}(\Gamma(m\epsilon))$ is the value of one of the off-diagonal elements of the stress tensor at the simulation step m , and ϵ is the time-step of the simulation. The evaluation of the viscosity through Eq. (2.5) is compromised by the fact that, while ideally the time-correlation function should go to zero for large time, in practise it is very noisy. As the function approaches to zero, Eq. (2.5) starts integrating noise and behaves like the distance traveled by a random walk, whose variance grows linearly with the upper integration limit. The evaluation of transport coefficients thus requires averaging over multiple trajectories (possibly multiple segments of a same long trajectory) and estimating the resulting uncertainty as a function of both the length of each trajectory and the upper limit of integration. This is a cumbersome task that often leads to a poor estimate of the statistical and systematic errors on the computed conductivity.

In the following sections we, first, present some useful notions and definitions of statistics and probability to provide several mathematical tools useful to understand the main analysis technique used in this work: the so-called *cepstral analysis*. This technique allows to avoid the direct calculation of the integral in (2.5) that needs very long trajectories in order to be accurate. In the last section of the chapter we present some applications and results that we obtained from classical simulations of water.

2.1 ELEMENTS OF STATISTICS

2.1.1 Stochastic processes

A *stochastic process* is, physically, any process running along in time and controlled by probabilistic laws. Mathematically speaking, a stochastic process is a collection of random variables $\{X(t), t \in T\}$, labelled by an index t defined in the range T and can be thought as the time. We will alternate between the notation X_t and $X(t)$ to denote the value of the stochastic process at time t .

The process may be continuous or discrete according to the nature of the index. For example, if $T = \{0, 1, 2, \dots\}$, we are considering a discrete stochastic process, while if $T = [0, \infty)$ we are considering a continuous one.

In numerical simulations, the stochastic processes correspond to the collections of the instantaneous values of the observables of the system. Since the time within the simulation is discrete, these stochastic processes, that are continuous in the physical world, become discrete and can be seen as the discretization of the real continuous ones, $X_m = X(\epsilon m)$, where ϵ is some discretization step (*i.e.* the time-step of the simulation). This discretization is only a numerical trick and we can always regard these processes as continuous in order to simplify the underlying math and manipulate them

more easily. A “numerically observed” sample, or realization of a (discrete) stochastic process is often referred to as a *time series* or *signal* and it will be indicated by $x(t)$.

Consider now a discrete stochastic process X_t , with the time index t in range $t \in [1, n]$, its joint probability density

$$\mathcal{P}(X_1, X_2, \dots, X_{n-1}, X_n), \quad (2.6)$$

totally describes the stochastic process itself. The covariance of the process at two different times is given by:

$$C_X(t_1, t_2) \stackrel{\text{def}}{=} \langle (X_{t_1} - \langle X_{t_1} \rangle) (X_{t_2} - \langle X_{t_2} \rangle) \rangle, \quad (2.7)$$

and it is also called its *time-correlation function*, where $\langle \cdot \rangle$ denote the expectation value with respect to the joint probability density (2.6).

Let us now introduce another important property of stochastic processes. A stochastic process X_t is said to be *weak stationary* if its expectation value does not vary with respect to time and if its time-correlation function is invariant under an arbitrary time translation, i.e.:

$$\begin{aligned} m_X(t) &\stackrel{\text{def}}{=} \langle X_t \rangle = m_X(t + \tau) \quad \forall \tau \in \mathbb{R}, \\ C_X(t_1, t_2) &\stackrel{\text{def}}{=} \langle (X_{t_1} - m_X(t_1)) (X_{t_2} - m_X(t_2)) \rangle = C_X(t_1 + \tau, t_2 + \tau) \quad \forall \tau \in \mathbb{R}. \end{aligned} \quad (2.8)$$

In particular, the last relation implies that the time-correlation function depends only on time differences

$$C_X(t_1, t_2) = C_X(t_1 - t_2, t_2 - t_2) = C_X(t_1 - t_2, 0). \quad (2.9)$$

The notion of weak stationary is enough for our purposes, even though if the marginal distributions of a process are invariant under time translation, the process is said to be *strong stationary* [Papoulis, 1991]. The marginal distribution of the variable X_i is defined as:

$$\mathcal{F}_i(X_1, X_2, \dots, X_{i-1}, X_{i+1}, \dots, X_n) = \int \mathcal{P}(X_1, X_2, \dots, X_i, \dots, X_n) dX_i. \quad (2.10)$$

In general, strong stationarity implies weak stationarity and not vice-versa, but there is a particular case: *Gaussian processes* [Ebden, 2015]. A stochastic process is said to be Gaussian if the marginal distributions of any n -uple of its values $\{X_{t_1}, X_{t_2}, \dots, X_{t_n}\}$ are multivariate normal distributions. For Gaussian processes, weak stationary implies strong stationary, because the covariance matrix uniquely determines all its the marginal distributions. This means that if the covariance matrix is invariant under time translation (i.e. the process is weak stationary) then also all the marginals function are invariant under time translation, involving the strong stationary of the process.

2.1.2 The Wiener-Khintchine Theorem

Even before enunciating the theorem, let us introduce some fundamental quantities that will become crucial for our future purposes.

Let us consider a process $\{X_t\}$, the time-correlation function is given in accord with Eq. (2.7) by:

$$C(t) = \langle X(t)X(0) \rangle, \quad (2.11)$$

where the brackets denote the averaging over an ensemble of realizations of the random variable and we assumed that $\langle X(t) \rangle = 0$. Let us assume that the process is weakly stationary, satisfying then conditions given by Eqs. (2.8). We will further assume that the Fourier transform of the time-correlation function $C(\tau)$ define by

$$\tilde{C}(\omega) = \int_{-\infty}^{\infty} C(t)e^{i\omega t} dt \quad (2.12)$$

exists.

The *truncated Fourier transform* of a realization $x(t)$ of the process $\{X_t\}$ over the interval $[-T/2, T/2]$ is defined as

$$\tilde{x}_T(\omega) \stackrel{\text{def}}{=} \int_{-T/2}^{T/2} x(t)e^{i\omega t} dt, \quad (2.13)$$

and since $x(t)$ is a random variable also $\tilde{x}_T(\omega)$ can be regarded as a stochastic process labelled by ω .

The *truncated spectral power density* of a process, $S_T(\omega)$, is defined as the expectation value of the modulus squared of the truncated Fourier transform of the process, divided by the width of the truncation window:

$$S_T(\omega) \stackrel{\text{def}}{=} \frac{1}{T} \langle |\tilde{x}_T(\omega)|^2 \rangle = \frac{1}{T} \left\langle \left| \int_{-T/2}^{T/2} x(t)e^{i\omega t} dt \right|^2 \right\rangle. \quad (2.14)$$

The *power spectral density* (PSD) or *power spectrum* of the random process $\{X_t\}$ is the limit for large- T of Eq. (2.14):

$$S(\omega) \stackrel{\text{def}}{=} \lim_{T \rightarrow \infty} S_T(\omega) = \lim_{T \rightarrow \infty} \frac{1}{T} \langle |\tilde{x}_T(\omega)|^2 \rangle. \quad (2.15)$$

The **Wiener-Khintchine theorem** [Khinchine, 1934; Wiener, 1930] states that the power spectral density is the Fourier transform of time-correlation function:

$$S(\omega) = \int_{-\infty}^{\infty} C(t)e^{i\omega t} dt. \quad (2.16)$$

Proof: Let us evaluate explicitly the power spectrum of the process

$$\langle |\tilde{x}_T(\omega)|^2 \rangle = \int_{-T/2}^{T/2} \int_{-T/2}^{T/2} \langle X(s)X(t) \rangle e^{i\omega(s-t)} ds dt \quad (2.17)$$

$$= \int_{-T/2}^{T/2} \int_{-T/2}^{T/2} C(s-t)e^{i\omega(s-t)} ds dt, \quad (2.18)$$

where the realization $x(t)$ was assumed to be real valued since, for our purpose, it assumes the meaning of the instantaneous values of physical observables.

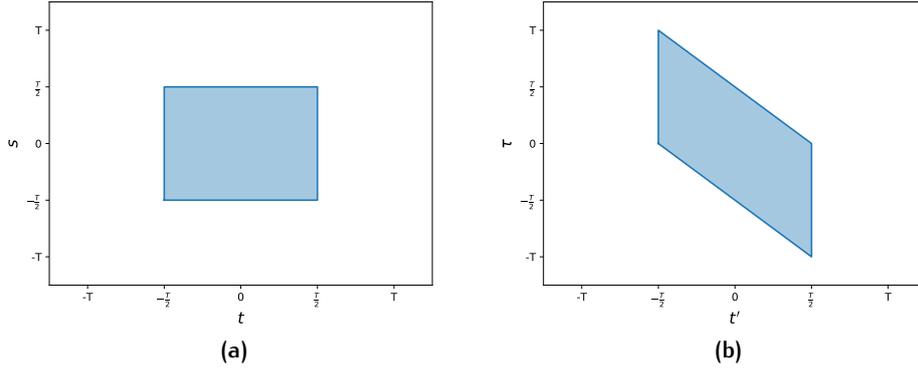


Figure 1: Domain of integration before (*left*) and after (*right*) the change of variables: $\tau = s - t$ and $t' = t$.

Since the integrand in Eq. (2.18) depends only on the variable $s - t$, defining a new variable $\tau = s - t$ and $t' = t$, with standard calculus manipulations, one can write

$$\int_{-T/2}^{T/2} dt \int_{-T/2}^{T/2} C(s-t)e^{i\omega(s-t)} ds = \int_{-T/2}^{T/2} dt' \int_{-T/2-t'}^{T/2-t'} C(\tau)e^{i\omega\tau} d\tau, \quad (2.19)$$

where the integration domain transforms as showed in Fig. 1 under the change of variables.

Going on with simple calculus manipulations, it is easy to show that the following relation holds

$$\langle |\tilde{x}_T(\omega)|^2 \rangle = \int_{-T}^T C(\tau)e^{i\omega\tau} (T - |\tau|) d\tau, \quad (2.20)$$

and dividing by T and letting T go to infinity becomes

$$\begin{aligned} S(\omega) &= \lim_{T \rightarrow \infty} \frac{1}{T} \langle |\tilde{x}_T(\omega)|^2 \rangle = \lim_{T \rightarrow \infty} \int_{-T}^T C(\tau)e^{i\omega\tau} \left(1 - \frac{|\tau|}{T}\right) d\tau \\ &= \lim_{T \rightarrow \infty} \int_{-T}^T C(\tau)e^{i\omega\tau} d\tau + \mathcal{O}\left(\frac{1}{T}\right), \end{aligned} \quad (2.21)$$

concluding the demonstration of the Wiener-Khintchine theorem.

Now one can show that weak stationarity implies that the time-correlation function is an even function of time:

$$C(t) = \langle X(t)X(0) \rangle = \langle X(t-t)X(0-t) \rangle = C(-t). \quad (2.22)$$

This let us write the PSD of the related process (i.e. the Fourier transform time correlation function) as:

$$\begin{aligned} S(\omega) &= \int_{-\infty}^{\infty} C(\tau)e^{i\omega\tau} d\tau \\ &= \int_{-\infty}^0 C(\tau)e^{i\omega\tau} d\tau + \int_0^{\infty} C(\tau)e^{i\omega\tau} d\tau \\ &= 2 \int_0^{\infty} C(\tau)\cos(\omega\tau) d\tau \end{aligned} \quad (2.23)$$

This result in conjunction with the Wiener-Khintchine theorem (2.16) allows one to write the shear viscosity transport coefficient given by (2.1) in a useful form:

$$\begin{aligned}\eta &= \frac{V}{k_B T} \int_0^\infty \langle \Pi^{\text{off}}(\Gamma_t) \Pi^{\text{off}}(\Gamma_0) \rangle dt \\ &= \frac{V}{k_B T} \int_0^\infty C_{\Pi^{\text{off}}}(t) dt \\ &= \frac{V}{2k_B T} S(\omega = 0).\end{aligned}\tag{2.24}$$

where $C_{\Pi^{\text{off}}}$ is the time correlation function of the off-diagonal element Π^{off} of the stress tensor.

This relation is very important because it allows to relate the zero frequency value of the power spectral density of the process, in the present case the stress process, to the corresponding transport coefficient, the shear viscosity.

2.2 CEPSTRAL ANALYSIS

In the present work we are interested in the shear viscosity coefficient and so let us indicate the time series of stress tensor as:

$$\Pi_n^{\text{off}} = \Pi^{\text{off}}(n\epsilon), \quad n = 0, 1, 2, \dots, N-1 \tag{2.25}$$

where ϵ is the sampling time (i.e. the time-step of the simulation) and N is the length of the time series that we assume to be even. For the sake of simplicity, let us drop the label off indicating the off-diagonal component of the stress tensor and let us consider Π_n as the time-series of one of them.

The discrete Fourier transform of the stress time series is defined as:

$$\tilde{\Pi}_k = \sum_{n=0}^{N-1} \Pi_n e^{2\pi i \frac{kn}{N}}, \tag{2.26}$$

for $0 \leq k \leq N-1$. The *sample spectrum* \hat{S}_k , called also *periodogram*, is defined as

$$\hat{S}_k = \frac{\epsilon}{N} |\tilde{\Pi}_k|^2, \tag{2.27}$$

and for large N it is an unbiased estimator of the power spectrum defined in Eq. (2.15), evaluated at $\omega_k = 2\pi k/N\epsilon$, namely: $\langle \hat{S}_k \rangle = S(\omega_k)$.

Since the Π 's are real, it is easy to show that the following relation holds

$$\tilde{\Pi}_k^* = \tilde{\Pi}_{N-k}, \tag{2.28}$$

since

$$\begin{aligned}\tilde{\Pi}_{N-k} &= \sum_{n=0}^{N-1} \Pi_n e^{2\pi i \frac{(N-k)n}{N}} \\ &= \sum_{n=0}^{N-1} \Pi_n e^{2\pi i n} e^{-2\pi i \frac{kn}{N}} \\ &= \sum_{n=0}^{N-1} \Pi_n e^{-2\pi i \frac{kn}{N}} = \tilde{\Pi}_k^*.\end{aligned}\tag{2.29}$$

Eqs. (2.28) and (2.27) naturally imply

$$\hat{S}_k = \hat{S}_{N-k}, \quad (2.30)$$

so that periodograms are usually reported for $0 \leq k \leq N/2$.

Now, we will show some important statistical properties the total stress, however these properties are valid for any general conserved-flux process. Any element of the stress tensor is defined as the integral over the volume of the corresponding stress density, $\tau(\mathbf{r}, t)$, whose space-correlation function, $\langle \tau(\mathbf{r}, t) \tau(\mathbf{r}', t) \rangle$, is usually short-ranged at equilibrium. Regarding the integral as a sum over different finite volumes and taking these volumes bigger than the average interaction length of the system, each different contribution to the total stress can be regarded as independent. Also, since the system is homogeneous in space, each contribution is identically distributed to the others. Thus the total stress tensor is a sum of *identically and independently distributed* variables (iid). The Central Limit Theorem [Papoulis, 1991] ensures the distribution of the total stress to be Gaussian with zero-mean (since at equilibrium the expectation value of the stress is zero). Thus, we can conclude that any conserved-flux process is Gaussian as well. The flux time series is in fact a multivariate stochastic variable that in the thermodynamic limit tends to a multivariate normal deviate. Since the Fourier transform of a Gaussian is itself a Gaussian, the Fourier transform of the process \tilde{P}_k is a Gaussian process with zero-mean too.

The variance of the $\tilde{\Pi}_k$ process is given by

$$\begin{aligned} \text{Var} [\tilde{\Pi}_k] &= \langle |\tilde{\Pi}_k|^2 \rangle \\ &= \langle [\Re(\tilde{\Pi}_k)]^2 + [\Im(\tilde{\Pi}_k)]^2 \rangle \\ &= \langle [\Re(\tilde{\Pi}_k)]^2 \rangle + \langle [\Im(\tilde{\Pi}_k)]^2 \rangle \\ &= \frac{N}{\epsilon} \langle \hat{S}_k \rangle = \frac{N}{\epsilon} S(\omega_k). \end{aligned} \quad (2.31)$$

Thus, indicating with $\mathcal{N}(\mu, \sigma^2)$ a normal distribution with mean value μ and variance σ^2 , one can say that the real and the imaginary part of the Fourier transform of the process are themselves stochastic process distributed as

$$\begin{aligned} k = 0, \frac{N}{2} : \quad \Re(\tilde{\Pi}_k) &\sim \mathcal{N}\left(0, \frac{N}{\epsilon} S(\omega_k)\right), \\ k \notin \{0, \frac{N}{2}\} : \quad \Re(\tilde{\Pi}_k), \Im(\tilde{\Pi}_k) &\sim \mathcal{N}\left(0, \frac{N}{2\epsilon} S(\omega_k)\right), \end{aligned} \quad (2.32)$$

since for $k = 0$ or $N/2$, $\tilde{\Pi}_k$ is real and since the real and the imaginary part of $\tilde{\Pi}_k$ are independent variables.

From Eqs. (2.31) and (2.32) we can derive the stochastic distribution of the periodogram \hat{S}_k for $k \notin \{0, N/2\}$:

$$\begin{aligned} \hat{S}_k &= \frac{\epsilon}{N} [\Re(\tilde{\Pi}_k)]^2 + [\Im(\tilde{\Pi}_k)]^2 \\ &= \frac{\epsilon}{N} \left\{ \left[\mathcal{N}\left(0, \frac{N}{2\epsilon} S(\omega_k)\right) \right]^2 + \left[\mathcal{N}\left(0, \frac{N}{2\epsilon} S(\omega_k)\right) \right]^2 \right\} \\ &= \frac{S(\omega_k)}{2} \{ \mathcal{N}_1^2 + \mathcal{N}_2^2 \}, \end{aligned} \quad (2.33)$$

where \mathcal{N}_1 and \mathcal{N}_2 are standard Gaussian variables, obtained from the relation $\mathcal{N} = \mathcal{N}(0, \sigma^2)/\sigma$. For $k \in \{0, N/2\}$ the same expression gives

$$\hat{S}_k = S(\omega_k)\mathcal{N}_1^2. \quad (2.34)$$

The distribution of the sum of n squared independent standard Gaussian variables is a chi-squared distribution χ_n^2 with n degrees of freedom. Chi-squared variables have the following statistic properties:

$$x : \chi_n^2 \longrightarrow \begin{cases} \langle x \rangle = n \\ \text{Var}[x] = 2n \end{cases} \quad (2.35)$$

Defining new variables $\hat{\xi}_k$ distributed as

$$\hat{\xi}_k \sim \frac{\chi_2^2}{2}, \quad k \notin \{0, N/2\}; \quad \hat{\xi}_k \sim \chi_1^2, \quad k \in \{0, N/2\}, \quad (2.36)$$

Eqs. (2.33) and (2.34) can now be written in a more compact and significant way:

$$\hat{S}_k = S(\omega_k)\hat{\xi}_k, \quad (2.37)$$

with $\hat{\xi}_k$ distributed as in Eq. (2.36). For the sake of simplicity, we consider all the $\hat{\xi}_k$ identically distributed, $\hat{\xi}_k \sim \chi_2^2/2$ for all values of k , thus making an error of order $O(1/N)$, which vanishes in the long-time limit $N \rightarrow \infty$.

Molecular dynamic simulations can provide us multiple realizations of the same process. In the present case, since the system under study is homogeneous and isotropic, the off-diagonal elements Π_{ij} of the stress matrix with $ij = xy, xz, yz$ are three independent and equivalent time-series of the same process, then a single MD trajectory provides three different independent realization of the same process

In the general case, let us suppose to have l different realizations of a process. We can then define a mean periodogram

$$\begin{aligned} {}^l\hat{S}_k &\stackrel{\text{def}}{=} \frac{\epsilon}{lN} \sum_{p=1}^l |{}^l\tilde{\Pi}_k|^2 \\ &= S(\omega_k){}^l\hat{\xi}_k, \end{aligned} \quad (2.38)$$

where ${}^l\hat{\xi}_k \sim \frac{1}{2l}\chi_{2l}^2$ since the periodogram is given by the sum of the square of $2l$ standard Gaussian variables. Eq. (2.38) shows once again that the mean periodogram is an unbiased estimator for the power spectral density of the process, $\langle {}^l\hat{S}_k \rangle = S(\omega_k)$, and in particular, through Eq. (2.24), its zero-frequency value is proportional to the transport coefficient.

An important property for a good estimator is the consistency: the variance of the estimator should decrease with the number of the sampled points. In this sense, ${}^l\hat{S}_k$ is not consistent. In fact, while it is unbiased and so its expectation value is the true value of the power spectral density, its variance is not affected by the number of sampled points N :

$$\begin{aligned} \langle {}^l\hat{S}_k \rangle &= S(\omega_k)\langle {}^l\hat{\xi}_k \rangle = S(\omega_k), \\ \text{Var}[{}^l\hat{S}_k] &= S(\omega_k)^2 \text{Var}[{}^l\hat{\xi}_k] = \frac{1}{l}S(\omega_k)^2, \end{aligned} \quad (2.39)$$

where we used the definition of ${}^l\hat{\xi}_k$ and the properties of the chi-square distribution in Eq. (2.35). This relations shows that the periodogram is unbiased but is not consistent. This is due to the fact that the noise given by ${}^l\hat{\xi}_k$ acts like a white-noise and is multiplicative. Thus, to deal with this problem one can take the logarithm of the periodogram in order to transform the multiplicative noise in an additive one, easier to treat with standard filtering techniques.

Let ${}^l\hat{\Lambda}_k = \log({}^l\hat{S}_k)$ be the *log-periodogram* of the time series. By taking the logarithm of Eq. (2.37) one obtains

$${}^l\hat{\Lambda}_k = \log(S(\omega_k)) + \log({}^l\hat{\xi}_k) = \log(S(\omega_k)) + {}^l\Lambda + {}^l\hat{\lambda}_k, \quad (2.40)$$

where ${}^l\hat{\lambda}_k = \log({}^l\hat{\xi}_k) - {}^l\Lambda$ are zero-mean identically and independent distributed stochastic variables, ${}^l\Lambda = \langle \log({}^l\hat{\xi}_k) \rangle = \varphi(1) - \log(1)$, and $\varphi(z)$ is the digamma function defined as the derivative of the logarithm of the Euler-Gamma function:

$$\varphi(z) = \frac{d}{dz} \log(\Gamma(z)) = \frac{\Gamma'(z)}{\Gamma(z)}. \quad (2.41)$$

The variance of the ${}^l\hat{\lambda}_k$ variables is $\sigma_1^2 = \varphi'(1)$, where $\varphi'(z)$ is the tri-gamma function, i.e. the second derivative of the logarithm of the Euler-Gamma function:

$$\varphi'(z) = \frac{d^2}{dz^2} \log(\Gamma(z)). \quad (2.42)$$

Then with this procedure we turned the initial multiplicative noise in Eq. (2.38) in an additive white noise with zero-mean. We wish now to clean up the logarithm of the periodogram from its high-frequency components that are mainly due to the white-noise. In practise, we apply a low-pass filter to Eq. (2.40) hoping in a reduction of the power of the noise, without affecting the true signal.

In order to do this, exploiting the so called "*cepstral analysis*" [Baroni et al., 2020; Childers et al., 1977; Ercole et al., 2017], we define the *cepstrum* of the time series P_n as the inverse Fourier transform of its log-periodogram [Childers et al., 1977]

$${}^l\hat{C}_n = \frac{1}{N} \sum_{k=0}^{N-1} {}^l\hat{\Lambda}_k e^{-2\pi i \frac{kn}{N}}. \quad (2.43)$$

Similarly to the sample power spectrum, the cepstral coefficients defined in Eq. (2.43) 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 ${}^l\hat{\lambda}_k$ appearing in Eq. (2.40) are a set of independent (almost) identically distributed zero-mean normal deviates [Peligrad & Wu, 2010]. It follows that:

$$\begin{aligned} {}^l\hat{C}_n &= C_n + \delta_{n0} {}^l\Lambda + {}^l\hat{\mu}_n, \\ C_n &= \frac{1}{N} \sum_{k=0}^{N-1} \log(S(\omega_k)) e^{-2\pi i \frac{kn}{N}}, \end{aligned} \quad (2.44)$$

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

$$\begin{aligned}\langle {}^l\hat{\mu}_n^2 \rangle &= \frac{\sigma_l^2}{N} \text{ for } n \notin \left\{ 0, \frac{N}{2} \right\} \\ \langle {}^l\hat{\mu}_n^2 \rangle &= 2\frac{\sigma_l^2}{N} \text{ otherwise}\end{aligned}\quad (2.45)$$

If the log-periodogram, $\log(S(\omega_k))$ is smooth enough, the number of non-null C_n in Eq. (2.44), is much smaller than N : only few of them really contributes to the smooth shape of $S(\omega_k)$ and so to the signal we want to compute. Analysing the different cepstral coefficients ${}^l\hat{C}_n$, one can then assume that for $n > P^*$, where P^* is the number of cepstral coefficient we want to keep, the coefficients do not carry information about the power spectral density, but only about the noise given by the variables ${}^l\hat{\mu}_n$ in Eq. (2.44). Then in order to clean the periodogram from the noise, one can perform a discrete Fourier transform of Eq. (2.43) restricted to $n < P^*$ and $n > N - P^*$:

$${}^l\hat{L}_k^* \equiv {}^l\hat{C}_0 + 2 \sum_{n=0}^{P^*-1} {}^l\hat{C}_n \cos\left(\frac{2\pi kn}{N}\right), \quad (2.46)$$

and in particular its zero-frequency component:

$$\begin{aligned}{}^l\hat{L}_0^* &= {}^l\hat{C}_0 + 2 \sum_{n=0}^{P^*-1} {}^l\hat{C}_n \\ &= \log(S(\omega = 0)) + {}^l\Lambda + {}^l\hat{\mu}_0 + 2 \sum_{n=0}^{P^*-1} {}^l\hat{\mu}_n.\end{aligned}\quad (2.47)$$

It follows from the properties satisfied by the ${}^l\hat{\mu}$'s in Eqs. (2.45) that ${}^l\hat{L}_0^*$ is a Gaussian variable with expectation value and variance given by:

$$\begin{aligned}\langle {}^l\hat{L}_0^* \rangle &= \log(S(\omega = 0)) + {}^l\Lambda, \\ \sigma_{{}^l\hat{L}_0^*}^2 &= \varphi'(l) \frac{4P^* - 2}{N},\end{aligned}\quad (2.48)$$

and finally the transport coefficient, in this case the viscosity, and its variance:

$$\begin{aligned}\eta &= \frac{V}{2k_B T} S(\omega = 0) = \frac{V}{2k_B T} e^{\langle {}^l\hat{L}_0^* \rangle - {}^l\Lambda}, \\ \sigma_\eta^2 &= \eta^2 \sigma_{{}^l\hat{L}_0^*}^2 = \eta^2 \varphi'(l) \frac{4P^* - 2}{N}.\end{aligned}\quad (2.49)$$

Then this technique not only allows to estimate the zero-frequency component of the log-periodogram, which is directly connected by the Wiener-Khintchine theorem to the transport coefficient, but gives also its standard deviation. In particular we see that this error depends over three different parameters: the length of the time-series N , the number of cepstral coefficients we decide to keep P^* and the number of realizations of the same process that we dispose l .

In order to estimate the optimal number of cepstral coefficients necessary to keep the features of the real power spectral density and to remove the

noise, in [Ercole et al., 2017] the authors suggested to use the Akaike's Information Criterion [Akaike, 1974]. This method consists in choosing P^* as the one that minimizes the function:

$$\text{AIC}(P) = -2 \max_{\theta} \log \mathcal{L}(\theta, P) + 2P, \quad (2.50)$$

where $\mathcal{L}(\theta, P)$ is the likelihood of the parameters $\theta = (\theta_1, \theta_2, \dots, \theta_P)$ of the model and P is the number of parameters that characterized the chosen model. Then the optimal number of parameters is the argument of the AIC minimum:

$$P_{\text{AIC}}^* \equiv \arg \min_P \text{AIC}(P). \quad (2.51)$$

In the present case, the parameters of the model are the coefficients C_n as defined in Eqs. (2.44). The log-likelihood of this parameters is

$$2 \log \mathcal{L}(C, P) = -\frac{N}{2\sigma_l^2} (\hat{C}_0 - (C_0 + \lambda)) ^2 - \frac{N}{\sigma_l^2} \sum_{n=1}^{P-1} (\hat{C}_n - C_n)^2 - \frac{N}{\sigma_l^2} \sum_{n=P}^{N/2} \hat{C}_n^2. \quad (2.52)$$

This result follow the fact that in our model we assume that for $n < P$ the stochastic variables \hat{C}_n contain information about the power spectral density and then are gaussian variables with mean value equal to the Fourier coefficients of the logarithm of the power spectral density of the process. While for $n \geq P$ the \hat{C}_n 's are due to the white noise giving zero-mean Gaussian variables.

The maximum of Eq. (2.52) is given by

$$2 \max_C \log \mathcal{L}(C, P) = -\frac{N}{\sigma_l^2} \sum_{n=P}^{N/2} \hat{C}_n^2, \quad (2.53)$$

and the AIC of this model turns out to be

$$\text{AIC}(P) = \frac{N}{\sigma_l^2} \sum_{n=P}^{N/2} \hat{C}_n^2 + 2P, \quad (2.54)$$

that is the quantity we have to minimize in order to get the right P^* in accord with Eq. (2.51).

2.3 APPLICATIONS TO CLASSICAL SIMULATIONS

In order to benchmark the methodology described above and to get used with this technique, we have applied it to the calculation of the shear viscosity of classical water. In particular the aim of this preliminary study, as well as validate the technique itself, is to determine the dependence of the viscosity over the size of the system and to find a reference length of the trajectory in order to obtain a error of about 10% on the viscosity value.

Classical MD simulations were ran using LAMMPS package [Plimpton, 1995]. More details over the specific simulation will be given later on.

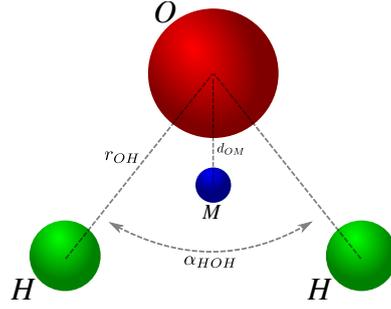


Figure 2: Depiction of a TIP4P/2005 water molecule.

	$\epsilon/k_B(\text{K})$	$\sigma(\text{\AA})$	$q_H(e)$	$d_{OM}(\text{\AA})$
TIP4P/2005	93.2	3.1589	0.5564	0.1546

Table 1: Optimized parameters for the TIP4P/2005 model.

2.3.1 The TIP4P/2005 model of water

The TIP4P/2005 [Abascal & Vega, 2005] is a rigid four sites model for the simulation of classical water. It consists of four interaction sites: three of them are placed at the positions of the oxygen atom and of the two hydrogen atoms of the molecule while the other site, often called the M site, is coplanar with the O and H sites and is located at the bisector of the H–O–H angle. In Fig. 2 is given the structure of the model. The model is rigid, then the value of the H–O distance and of the H–O–H angle are kept fixed to the experimental values respectively of $r_{OH} = 0.9572\text{\AA}$ and $\alpha_{HOH} = 104.52^\circ$.

The inter-molecular pair potential has two contributions: a Lennard-Jones U_{LJ} term and an electrostatic interaction U_{el} . The peculiar feature of the model is that the oxygen site O carries no charge, but contributes to the the LJ term:

$$U_{LJ} = 4\epsilon \left[\left(\frac{\sigma}{r_{OO}} \right)^{12} - \left(\frac{\sigma}{r_{OO}} \right)^6 \right]. \quad (2.55)$$

Conversely the H and M sites are charged and contribute only to the electrostatic potential between two molecules, namely i and j in the following way

$$U_{el} = \frac{e^2}{4\pi\epsilon_0} \sum_{a,b} \frac{q_a q_b}{r_{ab}}, \quad (2.56)$$

where e is the electron charge, ϵ_0 is the permittivity of vacuum, and a and b stands for the charged sites of molecules i and j , respectively.

The model is then fully described by four parameters: the strength ϵ and the length σ of the Lennard-Jones interaction, the charge of the hydrogen site (or of the M site equivalently), the distance d_{OM} between the oxygen site and the M site. The optimized values of these parameters are given in Table 1.

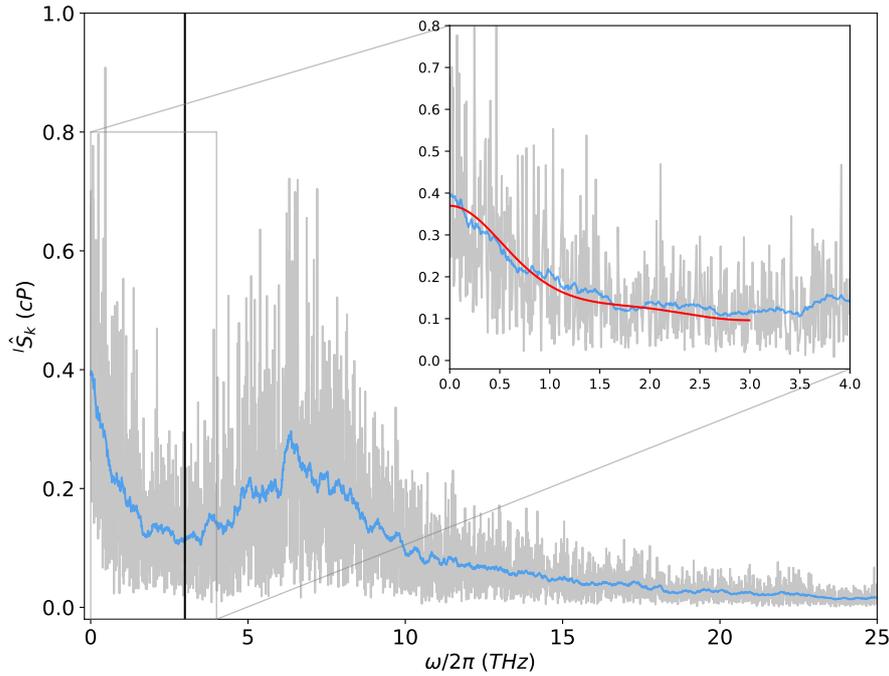


Figure 3: Mean periodogram of stress time series of water obtained from a 150 ps molecular dynamics trajectory. *Grey*: periodogram obtained directly for Eq. (2.38) with $l = 3$ (*i.e.* with the three equivalent components of the stress tensor). *Blue*: periodogram filtered with a moving average window of width 0.3 THz, useful to reveal the main features of the spectrum. The vertical line identifies f^* used in order to restrict the analysis over the low-frequencies region of the spectrum. The inset is a magnification of the low-frequency region with the result of the cepstral analysis in *red*.

2.3.2 Validation of the technique

In this section the results of the cepstral analysis on water TIP4P/2005 model are presented. The water system was chosen as a cubic super-cell containing 384 atoms (128 water molecules) with a density $\rho = 0.998 \text{ g/cm}^3$.

A first molecular dynamics simulation was run in order to equilibrate the system in the NVT ensemble at the target temperature $T = 350\text{K}$ for several hundred picoseconds, then we performed a very long (about 50 ns) NVE simulation in order to collect data for the analysis. Time-step of the simulations was chosen $\epsilon = 0.2 \text{ fs}$ in order to have a satisfying conservation of the total energy.

In order to perform the cepstral analysis technique, we used the software thermocepstrum [Ercole & Bertossa, 2018]. The software is a python module designed specifically to handle the computation of thermal conductivities via cepstral analysis. The original software, available as an open source project on GitHub, has been slightly modified in order to deal with stress time series and to predict the shear viscosity.

In Fig. 3 we report the sampled mean periodogram of the stress tensor averaged over the three equivalent off-diagonal elements (*i.e.* xy, xz, yz).

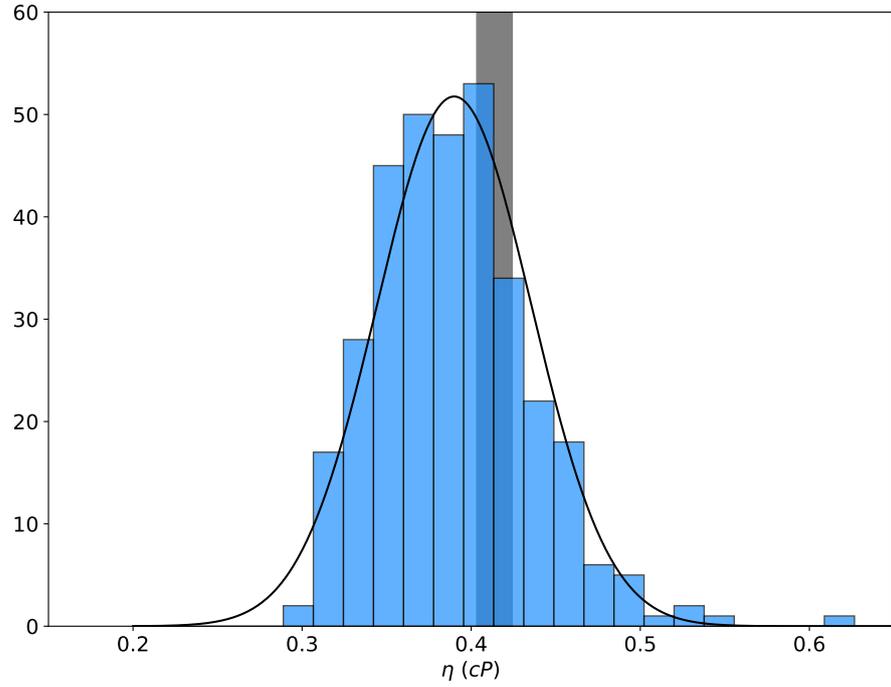


Figure 4: Distribution of the results given by the cepstral analysis protocol applied different trajectories of 150 ps. The *black line* is a Gaussian function with expectation value given by the sample mean and standard deviation obtained from the empirical statistics of the resulting data. The *vertical grey band* indicates the location of the value of the viscosity, η_B , evaluated from block analysis.

The solid blue line represents a moving average¹ computed over a narrow frequency window while the vertical black line indicate a cutoff frequency used for the cepstral analysis. It is in fact convenient, since the transport coefficient depends on the low-frequency behavior of the spectrum, to apply a low-pass filter to the time series in order to eliminate the high-frequency portion of the spectrum. Given a new cutoff frequency f^* , one can directly re-sample the stress time series with a new time-step $\epsilon' = (2f^*)^{-1}$ resulting in a low-pass filter at the chosen frequency. According with the previous application of this technique for the calculation of transport coefficients [Ercole et al., 2017], the value of the cutoff frequency, f^* , was chosen just at the end of the first narrow peak.

In order to validate the cepstral data-analysis protocol, we first computed the viscosity from a direct integration of the stress auto-correlation function, Eq. (2.4), combined with standard block analysis over the 50 ns long trajectory obtaining $\eta_B = 0.41 \pm 0.01$ cP, that will be taken as a reference. In Fig. 4 we show the distribution of the values obtained from the cepstral analysis technique, η , on each segment of 150 ps of the trajectory. The optimal P_A^* has been determined independently on each segment, while the f^* was chosen once for all from one of them, since the different periodograms of the different segments share the same main features. The graph shows a fair

¹ Given a sequence $\{x_i\}_{i=1}^N$ the moving average of width n is a new sequence $\{s_i\}_{i=1}^{N-n+1}$ defined from the x_i by taking the arithmetic mean of subsequences of n terms: $s_i = \sum_{j=1}^{i+n-1} x_j$

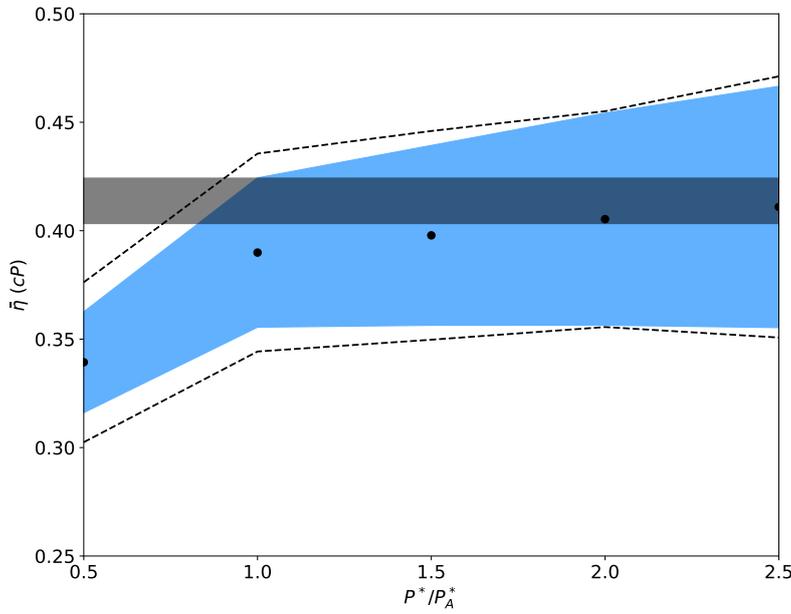


Figure 5: Dependence of the mean $\bar{\eta}$, as estimated with our data analysis technique, on the number of cepstral coefficients P^* . Here P_A^* is the optimal number of cepstral coefficients given by the AIC method. The *black dots* represent the mean value of the viscosity provided and the *blue area* is its average error, both computed over multiple molecular dynamics trajectories (about 150 ps). The *black dashed lines* indicate the standard deviation of data estimated from empirical statistics. The *horizontal grey band* indicates the location of the value of the viscosity, η_B , evaluated from block analysis.

compatibility between the values computed through the data-analysis protocol and the value obtained by blocks analysis approach, depicted as the black region, proving the reliability of the technique.

In order to analyse the dependence between the transport coefficient and the number of cepstral coefficients P^* , we display in Fig. 5 the mean viscosity transport coefficient $\bar{\eta}$ for different choice of P^* , averaged over all the different 150 ps long trajectory. In particular, taking as reference the number of optimal cepstral coefficients given by the Akaike Information Criterion, P_A^* , we report the dependence between $\bar{\eta}$ and P^*/P_A^* . We observe that when P^* is larger than P_A^* , the estimated value of η doesn't depend markedly on P^* . Still, a very slight bias seems to be present probably due to the difficulty that the AIC seems to present when it has to deal with a sharp low-frequency peak in the power visible in Fig. 3.

In Fig. 6 we display the dependence of the resulting estimate of $\bar{\eta}$, the average viscosity over the different segment of the whole trajectory, as a function of the cut-off frequency, along with the dependence of the optimal number of the cepstral coefficients, P_A^* . Fig. 6a shows a bias between the reference value, η_B , and the viscosity evaluated through the cepstral technique, $\bar{\eta}$. In particular this bias becomes relevant for large f^* , leading to a non-compatibility between the two approaches. As mentioned before, this is due to the fact that the AIC seems not to be a totally reliable indicator of the number of cepstral coefficients. This is, again, due to the sharp peak very close to the zero-frequency of the spectrum. By increasing the optimal

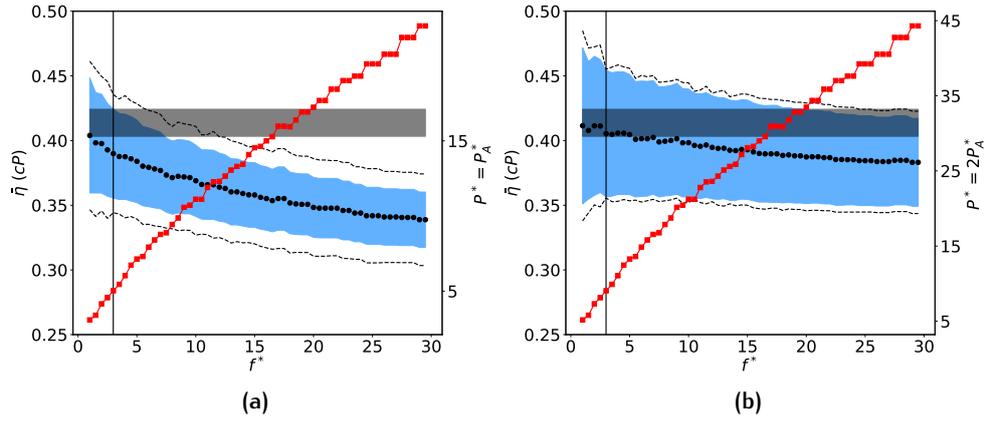


Figure 6: The two different graphs display the results obtained with cepstral analysis with respectively $P^* = P_{\Lambda}^*$ (left) and $P^* = 2P_{\Lambda}^*$ (right), with P_{Λ}^* the optimal number of cepstral coefficients provided by the AIC.

Red squares: dependence of the optimal number of cepstral coefficients given by the AIC, P_{Λ}^* , on the cut-off frequency f^* .

Black dots: dependence of the mean viscosity $\bar{\eta}$ on the cut-off frequency f^* . The *vertical black line* indicates the value of f^* chosen for the previous analysis. The meaning of the *blue region* and of the *black dashed lines* are the same as in Fig. 5.

number of cepstral coefficients provided by AIC by a factor two, $P^* = 2P_{\Lambda}^*$, the bias introduced decreases as indicated in Fig. 6b. In any case, with the cut-off frequency we've chosen, the bias introduced by the AIC seems not to be relevant and the result is totally compatible with the result given by block analysis.

To conclude, Fig. 7 displays the average relative error given by the cepstral analysis technique for different block sizes. In particular, a trajectory of 150 ps seems to be enough to guarantee an error smaller than 10%.

To sum up the results of the present discussion, we benchmarked the cepstral analysis technique by comparing it with standard procedure, block analysis, for the evaluation of viscosity transport coefficient from molecular dynamics trajectories. The results are satisfying and show a fully compatibility between the two procedures. Still, cepstral analysis seems to introduce a slight bias on the transport coefficients, probably relied on the sharpness of the low-frequency peak of the spectrum. In this case the AIC does not predict the correct number of cepstral coefficients and increasing this number by a factor two, $P^* = 2P_{\Lambda}^*$, appear to be a more satisfactory choice. Moreover, a trajectory of 150 ps seems to be enough in order to obtain a acceptable standard deviation over the transport coefficient.

2.3.3 Size scaling

In this last section, we want to study the dependence between the size of the simulated system and the viscosity transport coefficient obtained from the cepstral analysis. The aim of this section is to find a satisfying size of the system in order to not include significant surface effects and evaluate correctly the viscosity in the bulk.

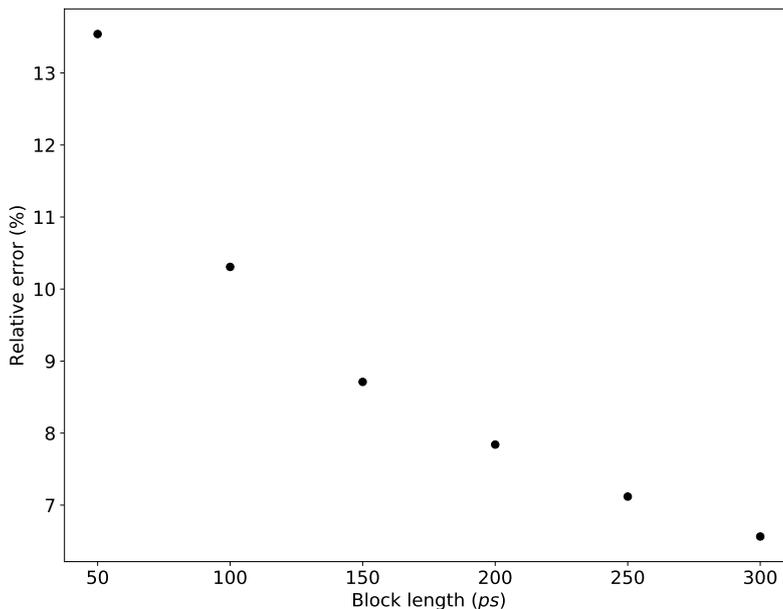


Figure 7: Dependence of the relative error as estimated from cepstral technique and averaged over different trajectories (of same length), on different trajectory lengths.

In order to do this, we ran a NVE simulation (about 10 ns) for five different water systems with different number of particles, respectively of 96, 192, 384, 768 and 1536 particles. Each system was brought to a target temperature of about 350K through a NVT simulation of several hundred of picoseconds. The density of the systems was chosen to be $\rho = 0.998 \text{ g/cm}^3$. The time-step of the simulations was set to $\epsilon = 0.2 \text{ fs}$. For every different system we repeated the previous data analysis protocol over segments of 100, 150, 200 and 250 ps of the whole trajectory in order to compute the viscosity coefficient of each system. Since each different periodograms share the same main features, the cutoff frequency $f^* = 3\text{THz}$ was chosen once all sizes. The optimal number of cepstral coefficients was, again, estimated using AIC method independently for each different trajectory of each different system.

Fig. 8 shows the average viscosity coefficient, $\bar{\eta}$, for different size systems obtained from the cepstral analysis as a function of trajectory lengths. The results look well constant over the block sizes. In Fig. 9 we show the average value of the shear viscosity obtained with 150 ps long trajectories as function of the size of the system, displaying a fair compatibility of the mean shear viscosity over the different systems.

We conclude that shear viscosity of water shows no significant system-size dependencies. These results match with those observed in earlier studies [Yeh & Hummer, 2004].

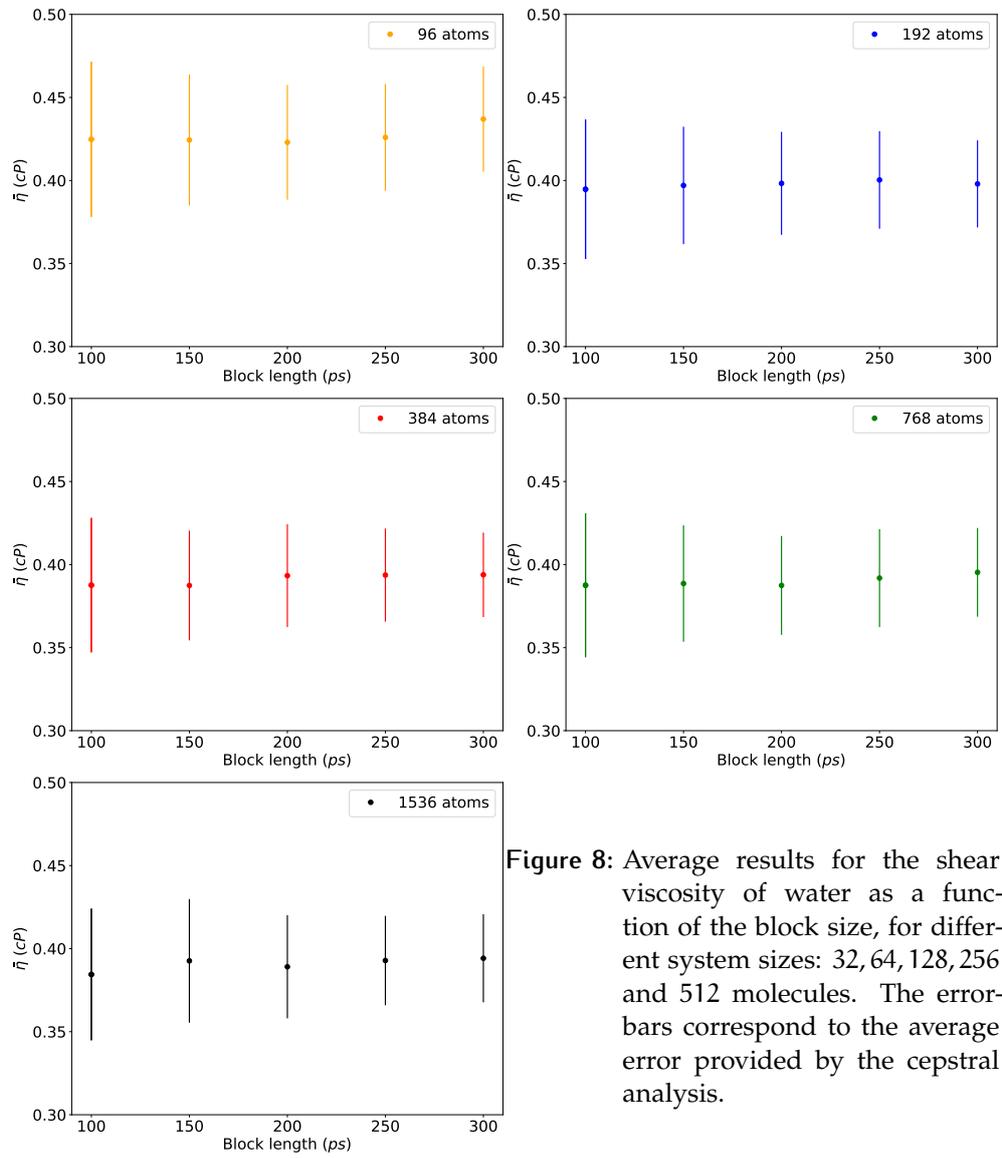


Figure 8: Average results for the shear viscosity of water as a function of the block size, for different system sizes: 32, 64, 128, 256 and 512 molecules. The error bars correspond to the average error provided by the cepstral analysis.

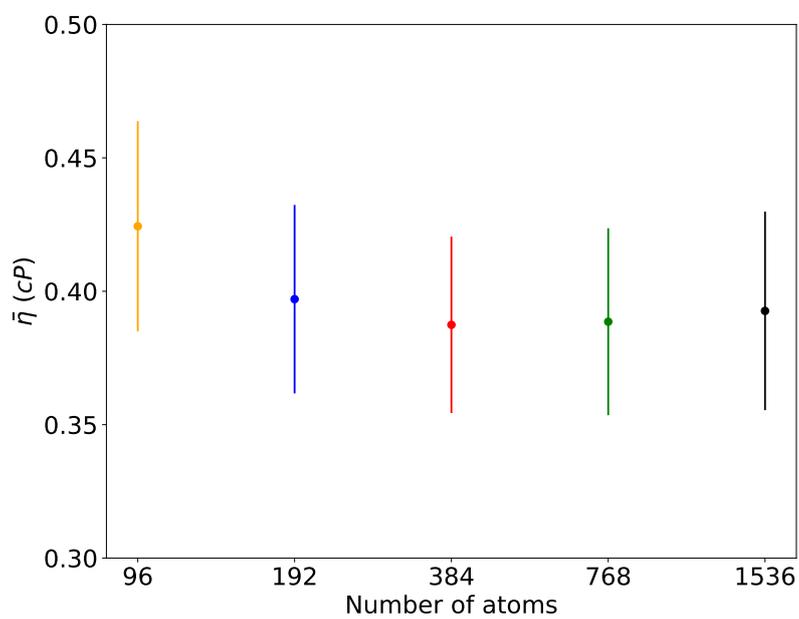


Figure 9: Average results for the shear viscosity of water obtained from trajectories of 150 ps for different system sizes. The errorbars correspond to the average error provided by the cepstral analysis.

3

AB INITIO MOLECULAR DYNAMICS

Ab initio electronic structure methods are computational methods based on quantum mechanics. *Ab initio* means "from first principle" since the only inputs required in such calculations are physical constants and no empirical data are required. *Ab initio* calculations aim to evaluate the many-electron wave function which is the solution of the electronic Schrödinger equation, given the positions of the nuclei and the number of electrons. From the many-electron wave function, one can evaluate several properties of the system such as electron densities, energy, and other observables. *Ab initio molecular dynamics* (AIMD) can also be performed, exploiting the Born-Oppenheimer approximation [Born & Oppenheimer, 1927]. The main difference between classical molecular dynamics and AIMD is the way in which the forces between atoms are evaluated. In classical MD a functional form for inter-atomic forces is assumed on the basis of physical considerations. An empirical potential is built fitting the parameters to a given experimental and/or theoretical data base. In contrast, in AIMD the inter-atomic forces are computed from electronic structure calculations in a consistent and accurate way as the simulation proceeds, no empirical data are needed and only the number of atoms and the species of each atom are used.

In the following sections we first briefly review the theoretical background behind *ab initio* electronic structure calculations like the Born-Oppenheimer approximation [1927], Density-Functional theory [Hohenberg & Kohn, 1964; Kohn & Sham, 1965] and the Car-Parrinello technique for AIMD simulations [Car & Parrinello, 1985]. In the last section we present some original results for the shear viscosity obtained from AIMD simulations.

3.1 BORN-OPPENHEIMER APPROXIMATION

In condensed matter physics we deal with nuclei and electrons. Thus, let us consider a system of N nuclei and n electrons. In principle one has to solve the Schrödinger equation:

$$H(\{\mathbf{r}\}, \{\mathbf{R}\})\Psi(\{\mathbf{r}\}, \{\mathbf{R}\}) = E_{\text{tot}}\Psi(\{\mathbf{r}\}, \{\mathbf{R}\}), \quad (3.1)$$

where $\{\mathbf{r}\}$ and $\{\mathbf{R}\}$ are respectively the entire set of coordinates of electrons and nuclei, while E_{tot} is the energy of the whole system. The hamiltonian H is given by

$$H(\{\mathbf{r}\}, \{\mathbf{R}\}) = - \sum_{\mu=1}^N \frac{\hbar^2}{2M_{\mu}} \nabla_{\mu}^2 - \sum_{j=1}^n \frac{\hbar^2}{2m_j} \nabla_j^2 + \frac{1}{2} \sum_{\mu \neq \mu'} \frac{Z_{\mu} Z_{\mu'} e^2}{|\mathbf{R}_{\mu} - \mathbf{R}_{\mu'}|} - \sum_{j, \mu} \frac{Z_{\mu} e^2}{|\mathbf{r}_j - \mathbf{R}_{\mu}|} + \frac{1}{2} \sum_{j \neq j'} \frac{e^2}{|\mathbf{r}_j - \mathbf{r}_{j'}|}, \quad (3.2)$$

where the first two terms are respectively the kinetic energy of the nuclei and of the electrons, the third term is the Coulomb repulsion between nuclei, the thirist term represents the Coulomb attraction between nuclei and electrons and finally the last contribution is the Coulomb repulsion between electrons.

A solution to the Schrodinger equation (3.1) would be a function dependent on the coordinates of all particles of the system. However, exploiting the fact that the nuclei are more massive than the electrons by at least three orders of magnitude, one can separate the global problem into an electronic problem for fixed nuclei and a nuclear problem under an effective potential generated by electrons. This approach is known as the *Born-Oppenheimer approximation* (BO approximation) [Born & Oppenheimer, 1927].

Formally, we assume the total wave function Ψ of the form

$$\Psi(\{\mathbf{r}\}, \{\mathbf{R}\}) = \phi(\{\mathbf{R}\})\psi_{\mathbf{R}}(\{\mathbf{r}\}), \quad (3.3)$$

where the electronic wave-function $\psi_{\mathbf{R}}$ depends now parametrically on the positions of the nuclei \mathbf{R}_{μ} and solves the *electronic Schrodinger equation*:

$$\left[-\sum_{j=1}^n \frac{\hbar^2}{2m_j} \nabla_j^2 - \sum_{j,\mu} \frac{Z_{\mu}e^2}{|\mathbf{r}_j - \mathbf{R}_{\mu}|} + \frac{1}{2} \sum_{j \neq j'} \frac{e^2}{|\mathbf{r}_j - \mathbf{r}_{j'}|} \right] \psi_{\mathbf{R}}(\{\mathbf{r}\}) = E(\{\mathbf{R}\})\psi_{\mathbf{R}}(\{\mathbf{r}\}), \quad (3.4)$$

where the nuclei positions $\{\mathbf{R}\}$ are fixed and the energy $E(\{\mathbf{R}\})$ of the electron gas depends upon them.

Inserting Eq. (3.3) in (3.1) and using Eq. (3.4) one can write

$$\left[-\sum_{\mu=1}^N \frac{\hbar^2}{2M_{\mu}} \nabla_{\mu}^2 + \frac{1}{2} \sum_{\mu \neq \mu'} \frac{Z_{\mu}Z_{\mu'}e^2}{|\mathbf{R}_{\mu} - \mathbf{R}_{\mu'}|} + E(\{\mathbf{R}\}) \right] \Psi(\{\mathbf{r}\}, \{\mathbf{R}\}) = E_{\text{tot}}\Psi(\{\mathbf{r}\}, \{\mathbf{R}\}). \quad (3.5)$$

Assuming now that the electronic wave-function depends only weakly upon the nuclei positions, i.e.:

$$\left[-\sum_{\mu=1}^N \frac{\hbar^2}{2M_{\mu}} \nabla_{\mu}^2 \right] \phi(\{\mathbf{R}\})\psi_{\mathbf{R}}(\{\mathbf{r}\}) \approx \psi_{\mathbf{R}}(\{\mathbf{r}\}) \left[-\sum_{\mu=1}^N \frac{\hbar^2}{2M_{\mu}} \nabla_{\mu}^2 \phi(\{\mathbf{R}\}) \right], \quad (3.6)$$

Eq. (3.5) leads to a Schrodinger equation for nuclei coordinates only

$$\left[-\sum_{\mu=1}^N \frac{\hbar^2}{2M_{\mu}} \nabla_{\mu}^2 + \frac{1}{2} \sum_{\mu \neq \mu'} \frac{Z_{\mu}Z_{\mu'}e^2}{|\mathbf{R}_{\mu} - \mathbf{R}_{\mu'}|} + E(\{\mathbf{R}\}) \right] \phi(\{\mathbf{R}\}) = E_{\text{tot}}\phi(\{\mathbf{R}\}), \quad (3.7)$$

where the electrons appear only in the form of a effective potential.

If the nuclei are regarded as classical particles, as it's usually done, the Schrodinger equation (3.7) can be replaced by the Newton's law. In particular one can write the force acting on the μ -th nucleus as

$$\begin{aligned} \ddot{\mathbf{R}}_{\mu} &= \frac{1}{M_{\mu}} \mathbf{F}_{\mu} = -\frac{1}{M_{\mu}} \nabla_{\mu} \left[\frac{1}{2} \sum_{\mu \neq \mu'} \frac{Z_{\mu}Z_{\mu'}e^2}{|\mathbf{R}_{\mu} - \mathbf{R}_{\mu'}|} + E(\{\mathbf{R}\}) \right] \\ &= -\frac{1}{M_{\mu}} \nabla_{\mu} V_{\text{pes}}(\{\mathbf{R}\}), \end{aligned} \quad (3.8)$$

where $V_{pes}(\{\mathbf{R}\})$ is known as *potential energy surface*, and it is the effective potential acting on the nuclei depending only on the atomic coordinates and species.

However, even if the Born-Oppenheimer approximation simplifies considerably the problem, all solutions of the electronic Schrodinger equation (3.4) are still very complex function of the coordinates of all electrons, parametrized by the atomic position.

3.2 DENSITY-FUNCTIONAL THEORY

The resolution of the Schrodinger equation for a many body system is made consistently hard by the need of working with the many-body wave-function, even in the BO approximation. This is a formidable task, even for systems of a few particles, and thus the search of schemes where one can work with objects simpler than the many-body wave-function has had a special appeal since the born of quantum mechanics. One such a scheme is provided by the so called Density-Functional theory (DFT), whereby one focuses on the one-body density $\rho(\mathbf{r})$ rather than on the many-body wave-function $\psi(\mathbf{r}_1, \dots, \mathbf{r}_n)$, as it is done in the Schrodinger equation. Its resolution is still far from being simple.

Density-Functional theory is based on the Hohenberg-Kohn theorem [Hohenberg & Kohn, 1964]. Hohenberg and Kohn were able to prove that in order to obtain information of the ground state of the system, only the electronic density of the ground state needs to be known. The latter is a quantity that depends only on one spacial coordinate, taking much less effort to be handled. In particular the theorem states the ground-state charge density of a system of n electrons

$$\begin{aligned} \rho(\mathbf{r}) &= \langle \psi | \sum_{j=1}^n \delta(\mathbf{r} - \mathbf{r}_j) | \psi \rangle \\ &= N \int |\psi(\mathbf{r}, \mathbf{r}_2, \dots, \mathbf{r}_n)|^2 d\mathbf{r}_2 \dots d\mathbf{r}_n, \end{aligned} \quad (3.9)$$

uniquely identifies the external potential V_{ext} acting on electrons and in turn the whole Hamiltonian $H = T + U + V_{ext}$, where T is the kinetic energy and U is the electron-electron repulsion. Thus, the energy of the ground-state is a functional of the charge density:

$$E[\rho] = \langle \psi | H | \psi \rangle = \langle \psi | T + U + V_{ext} | \psi \rangle = F[\rho] + \int v_{ext}(\mathbf{r}) \rho(\mathbf{r}) d\mathbf{r}, \quad (3.10)$$

where $F[\rho]$ is a universal functional describing the kinetic contribution and the electron-electron repulsion, moreover we've assumed that the external potential has the form $V_{ext} = \sum_j v_{ext}(\mathbf{r}_j)$ and could be, for example, the ion-electron interaction.

This sketch provides a relevant simplification for obtaining the properties of the system. While the electronic density still needs to be found, it can now be obtained from a variational ansatz by minimizing the total energy¹

$$\rho(\mathbf{r}) = \arg \min_{\rho} E[\rho], \quad (3.11)$$

This avoids the necessity to use many-body wave functions.

3.2.1 Kohn-Sham equations

Kohn-Sham formalism [Kohn & Sham, 1965] provides a scheme in order to cast the density functional problem of interacting particles into a self-consistent problem of non-interacting particles. In practice, one starts assuming that any interacting one body density is also solution of an auxiliary non-interacting problem with an appropriate external potentials. Then, let us introduce a non-interacting system having the same electronic density as the true one

$$\rho(\mathbf{r}) = \sum_j |\psi_j(\mathbf{r})|^2, \quad (3.12)$$

where ψ_j are known as *Kohn-Sham orbitals* and are orthonormal single-electron wave-functions, $\langle \psi_k | \psi_j \rangle = \delta_{kj}$. We define the *exchange-correlation energy*, the functional given by

$$E_{xc}[\rho] = T[\rho] + U[\rho] - (T_{ni}[\rho] + V_H[\rho]), \quad (3.13)$$

where $T_{ni}[\rho]$ and $V_H[\rho]$ are respectively the kinetic energy of the non-interacting electrons, and the electrostatic *Hartree energy*:

$$T_{ni}[\rho] = -\frac{\hbar^2}{2m} \sum_j \langle \psi_j | \nabla_j^2 | \psi_j \rangle, \quad (3.14)$$

$$V_H[\rho] = \iint \frac{\rho(\mathbf{r})\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} d\mathbf{r}d\mathbf{r}'. \quad (3.15)$$

Finally the total energy of the system (3.10) can be written as

$$E[\rho] = T_{ni}[\rho] + U_H[\rho] + E_{xc}[\rho] + \int v_{ext}(\mathbf{r})\rho(\mathbf{r})d\mathbf{r}. \quad (3.16)$$

By imposing the condition that the Kohn-Sham orbitals minimize the energy functional (3.16) under the orthonormality constraints, one obtains the *Kohn-Sham equations*

$$H_{KS}\psi_j(\mathbf{r}) = \left[-\frac{\hbar^2}{2m} \nabla^2 + v_{eff}(\mathbf{r}) \right] \psi_j(\mathbf{r}) = \epsilon_j \psi(\mathbf{r}), \quad (3.17)$$

where the effective potential, $v_{eff}(\mathbf{r}) = v_H(\mathbf{r}) + v_{xc}(\mathbf{r}) + v_{ext}(\mathbf{r})$ is a functional of the density, in addition to being a function of \mathbf{r} , given by the sum of respectively the exchange-correlation potential, the Hartree potential and

¹ Here we assume the ground-state not to be degenerate.

the external potential for the interacting system. The first two contributions are defined as

$$v_H([\rho]; \mathbf{r}) = \frac{\delta V_H[\rho]}{\delta \rho(\mathbf{r})} = e^2 \int \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}', \quad (3.18)$$

$$v_{xc}([\rho]; \mathbf{r}) = \frac{\delta E_{xc}[\rho]}{\delta \rho(\mathbf{r})}, \quad (3.19)$$

where the derivative symbol $\delta/\delta\rho(\mathbf{r})$ denotes the functional derivative with respect to the density.

From the Kohn-Sham equations one self-consistently obtains the density given by Eq. (3.12), occupying the orbitals starting from the lowest one (in energy). In practice, one starts from a guess density $\rho(\mathbf{r})$ and iterates (3.17) and (3.12) up to self-consistency, once one has reached it, the total energy of the ground-state can be obtained as

$$E = \sum_j \epsilon_j - V_H[\rho] - \int v_{xc}(\mathbf{r}) \rho(\mathbf{r}) d\mathbf{r} + E_{xc}[\rho]. \quad (3.20)$$

There is still a substantial problem of this scheme: the exchange-correlation energy functional, E_{xc} , is still not known, and so the corresponding potential $v_{ex}(\mathbf{r})$. However a lot of approximations have been proposed and one of the most remarkable and surprisingly effective is the *Local Density Approximation* (LDA) [1965], which assumes that the exchange-correlation functional can be re-write as

$$E_{xc}[\rho] = \int \rho(\mathbf{r}) e_{xc}(\rho(\mathbf{r})) d\mathbf{r}, \quad (3.21)$$

using for the exchange-correlation energy density $e_{xc}(\rho(\mathbf{r}))$ the result for the homogeneous electron gas of density ρ , computed in each point \mathbf{r} . This calculation can be accurately carried out through Quantum Monte Carlo techniques.

3.2.2 Stress tensor in DFT

In the chapter 1 we presented the formal expressions for stress (1.44) applicable to general many-body systems. However, this expression is not in a form which can take advantage of the simplifications resulting from expressing the energy as a functional of the electronic density. Let us consider a macroscopic and finite solid of volume V , the *static* macroscopic stress is defined as the derivative of the total potential energy of the nuclei with respect to the macroscopic strain [Martin, 2004]

$$\Pi_{\alpha\beta}^s = \frac{1}{V} \frac{\partial V_{pes}}{\partial \epsilon_{\alpha\beta}} = \frac{1}{V} \frac{\partial E_{ii}}{\partial \epsilon_{\alpha\beta}} + \frac{1}{V} \int \rho(\mathbf{r}) \frac{\partial V_{ext}(\mathbf{r})}{\partial \epsilon_{\alpha\beta}} d\mathbf{r}, \quad (3.22)$$

where the last equality follows from the Helmann-Feymann theorem[DEB, 1973]. The apex s indicates the fact that this expression contains only the contribution due to the potential energy to the stress, and the kinetic contribution due to the classical motion of the ions. In Eq. (3.22) E_{ii} is the ionic repulsion, ρ is the electronic density, V_{ext} is the electron-ion potential, and

greek subscripts are cartesian components. Considering a strained system, the displaced ionic positions are given by

$$\mathbf{R}_{l,\alpha} \rightarrow \mathbf{R}'_{l,\alpha} = \mathbf{R}_{l,\alpha} + \sum_{\beta} \varepsilon_{\alpha\beta} \mathbf{R}_{l,\beta}, \quad (3.23)$$

the electron-ion potential is in the form

$$V_{\text{ext}}(\mathbf{r}) = \sum_{\mathbf{l}} v(|\mathbf{r} - \mathbf{R}_{\mathbf{l}}|). \quad (3.24)$$

We therefore recast (3.22) as

$$\mathbf{V} \cdot \Pi_{\alpha\beta}^s = \frac{\partial V_{\text{pes}}}{\partial \varepsilon_{\alpha\beta}} = \frac{\partial E_{\text{ii}}}{\partial \varepsilon_{\alpha\beta}} - \int \rho(\mathbf{r}) \sum_{\mathbf{l}} \mathbf{R}_{l\beta} v'_{\alpha}(\mathbf{r} - \mathbf{R}_{\mathbf{l}}) d\mathbf{r}, \quad (3.25)$$

where

$$v'_{\alpha}(\mathbf{r}) = \frac{r_{\alpha}}{r} \frac{\partial v'(|\mathbf{r}|)}{\partial |\mathbf{r}|}. \quad (3.26)$$

The expression of Eq. (3.25) is only meaningful for a finite system, since it is boundary sensitive and depends upon the position of the ions which is ill-defined in PBC. Therefore, let us transform it in a more useful expression. Within the Density-Functional theory, the density ρ , as we've seen, is given by

$$\rho(\mathbf{r}) = \sum_i f_i |\psi_i(\mathbf{r})|^2, \quad (3.27)$$

where f_i is an occupancy factor and the ψ 's are the usual Kohn-Sham orbitals, eigenfunctions of the hamiltonian H_{KS} define in Eq. (3.17). Exploiting the commutator

$$\left[H_{\text{KS}}, r_{\beta} \frac{\partial}{\partial r_{\alpha}} \right] = \frac{p_{\alpha} p_{\beta}}{m} - r_{\beta} \frac{\partial}{\partial r_{\alpha}} (V_{\text{ext}} + V_{\text{H}} + V_{\text{xc}}), \quad (3.28)$$

which has a vanishing mean value over the orbitals, and substituting $\mathbf{R}_{l,\beta} = r_{\beta} - (\mathbf{r} - \mathbf{R}_{\mathbf{l}})_{\beta}$ in Eq. (3.25), we obtain the following expression for the macroscopic stress tensor

$$\begin{aligned} \Pi_{\alpha\beta}^s &= \frac{1}{V} \frac{\partial E_{\text{ii}}}{\partial \varepsilon_{\alpha\beta}} + \frac{1}{V} \frac{\partial E_{\text{xc}}}{\partial \varepsilon_{\alpha\beta}} + \frac{1}{V} \int \rho(\mathbf{r}) \sum_{\mathbf{l}} (\mathbf{r} - \mathbf{R}_{\mathbf{l}})_{\beta} v'_{\alpha}(\mathbf{r} - \mathbf{R}_{\mathbf{l}}) d\mathbf{r} \\ &\quad - \frac{1}{V} \sum_i f_i \langle \psi_i | \frac{p_{\alpha} p_{\beta}}{m} | \psi_i \rangle - \frac{e^2}{2V} \int \rho(\mathbf{r}) \rho(\mathbf{r}') \frac{(r_{\alpha} - r'_{\alpha}) - (r_{\beta} - r'_{\beta})}{|\mathbf{r} - \mathbf{r}'|^3} d\mathbf{r} d\mathbf{r}', \end{aligned} \quad (3.29)$$

where all the terms, except the exchange-correlation (XC) contribution, are now manifestly boundary insensitive and extensive in form. Within LDA approximation, the XC contribution to the stress can be shown [Nielsen & Martin, 1985a] to be a pure isotropic pressure, while in the most general case (and for low-symmetry solids) it is anisotropic. In particular for gradient-corrected functionals (GGA), the explicit boundary insensitive and extensive form is given in the Ref. [Dal Corso & Resta, 1994].

Moreover, for some DFT calculation schemes it is advantageous to express all quantities, such as the energy, the density and the Kohn-Sham orbitals,

in reciprocal space where a manageable plane-wave basis set gives high calculation accuracy and efficiency. In this context, a general expression of the macroscopic stress tensor for either local and non-local pseudopotential can be found in the Ref. [Nielsen & Martin, 1985b].

3.3 CAR-PARRINELLO MOLECULAR DYNAMICS

Ab initio molecular dynamics, in the BO approximation, can be successfully carried out within the DFT scheme in order to obtain the PES at fixed atomic positions and then evaluating the force acting on the nuclei through Eq. (3.2) together with the Hellmann-Feymann theorem. This allows to write the force acting on the μ -th atom as

$$\mathbf{F}_\mu = -\nabla_\mu \left[-\int \rho(\mathbf{r}) \frac{Z_\mu e^2}{|\mathbf{r} - \mathbf{R}_\mu|} d\mathbf{r} + \sum_{\mu' (\neq \mu)} \frac{Z_\mu Z_{\mu'} e^2}{|\mathbf{R}_{\mu'} - \mathbf{R}_\mu|} \right], \quad (3.30)$$

where $\rho(\mathbf{r})$ is the electron density obtained from self-consistent calculation. The key problem, however, is that the calculations must be very efficient in order to reduce the expensive computational cost of DFT calculations.

The Car-Parrinello algorithm for AIMD [Car & Parrinello, 1985] consists in a unified approach that does not consider the motion of nuclei and the solution of the equations for electrons at fixed $\{\mathbf{R}_\mu\}$ as separate problems, but rather it tries to treat them at the same level. Again, in ab initio molecular dynamics, the motion of the nuclei is determined by the forces acting on them, evaluated starting from the solution of the electronic quantum problem. *The special feature of the Car-Parrinello algorithm is that it solve the quantum electronic problem using molecular dynamics* [Martin, 2004] rather than using DFT calculations or other ab initio approaches. This is achieved adding a *fictitious* kinetic energy carried by the electrons, which leads to a fictitious Lagrangian for both electrons and nuclei [Car & Parrinello, 1985]

$$\begin{aligned} \mathcal{L} = & \sum_j \frac{1}{2} \tilde{m} \int |\psi_j(\mathbf{r})|^2 d\mathbf{r} + \sum_\nu \frac{1}{2} M_\nu \dot{\mathbf{R}}_\nu^2 - V_{pes} [\{\psi_j\}, \{\mathbf{R}_\mu\}] \\ & + \sum_{jk} \Lambda_{jk} \left[\int \psi_j^*(\mathbf{r}) \psi_k(\mathbf{r}) d\mathbf{r} - \delta_{jk} \right], \end{aligned} \quad (3.31)$$

where M_μ are the physical ionic masses and \tilde{m} are arbitrary parameters of appropriate units. The ψ 's are single-electron orbitals, while $V_{pes} [\{\psi_j\}, \{\mathbf{R}_\mu\}]$ is the DFT potential energy surface written as a functional of the single-electron orbitals and the atomic positions. The last term in Eq. (3.31) ensures the orthogonality of the electronic states.

The Lagrangian (3.31) generates a dynamics for the parameters $\{\psi_j\}$ and $\{\mathbf{R}_\mu\}$ through the equation of motions:

$$\begin{aligned} \tilde{m} \ddot{\psi}_j(\mathbf{r}, t) &= -\frac{\delta E}{\delta \psi_j^*(\mathbf{r})} + \sum_k \Lambda_{jk} \psi_k(\mathbf{r}, t) \\ &= H_{KS} \psi_j(\mathbf{r}, t) + \sum_k \Lambda_{jk} \psi_k(\mathbf{r}, t) \end{aligned} \quad (3.32)$$

$$M_\mu \ddot{\mathbf{R}}_\mu = \mathbf{F}_\mu = -\nabla_\mu E \quad (3.33)$$

where we see that the parameter \tilde{m} plays the role of a *fictitious mass* for the electrons chosen in order to reach an optimal convergence of the solution to the true adiabatic solution. The equation of motions can be solved with standard techniques such as the Verlet algorithm [Frenkel & Smit, 2001].

We highlight the fact that the fictitious kinetic energy of the electrons has nothing to do with the real quantum kinetic energy and that the motion provided by the Car-Parrinello Lagrangian (3.31) is also fictitious. The purpose of this fictitious energy is to evolve the single-electron orbitals always staying close to the real ground-state (i.e. close to the Kohn-Sham orbitals). This can be achieved by an appropriate choice of the fictitious mass of the electrons and of the time-step of the simulation, the smaller the value of \tilde{m} the better the adiabaticity is preserved. Small values of \tilde{m} , however, result in small values for the integration time-step.

A typical Car-Parrinello simulation starts with an electronic minimization at fixed ionic positions, in order to bring the electronic system on the ground state relative to the starting atomic configuration. This can be accomplished, instead of solving the Kohn-Sham equations (3.17) in the standard self-consistent way, by cooling the system down reducing the fictitious kinetic energy through the fictitious mass parameter that plays the role of the temperature in the standard simulated annealing techniques. This approach is called *dynamical simulated annealing* and provides a way alternative to find the minimum of the self-consistent Kohn-Sham equations. Once the ground-state has been reached, integrating the equation of motions (3.32) and (3.33) allows one to carry out ab initio molecular dynamics.

The stress tensor in Car-Parrinello dynamics can be evaluated exploiting Eq. (3.29) and adding the kinetic contribution of the moving ions (present also in the classical equation of the stress tensor (1.43)). The macroscopic stress can be written as follows [Parrinello & Rahman, 1980]

$$\Pi_{\alpha\beta} = - \sum_{\mu} \frac{M_{\mu} v_{\alpha}^{\mu} v_{\beta}^{\mu}}{V} + \frac{1}{V} \frac{\partial V_{pes}}{\partial \varepsilon_{\alpha\beta}} \quad (3.34)$$

where v_{α}^{μ} is the component α of the velocity vector of the atom μ . The second term of the equation is the static contribution to the stress tensor due to the potential energy, and can be evaluated from (3.29) using the single electron orbitals solving the equation of motion of the CP Lagrangian (3.31).

3.4 SHEAR VISCOSITY

In this work, we use on the Perdew-Burke-Ernzerhof (PBE) approximation to the exact exchange and correlation (XC) functional, whose parameters have been determined from first principles [Perdew et al., 1996]. The PBE functional can provide a reasonable description of liquid water, although PBE simulations need to be run at an elevated temperature (e.g., 400 K) in order to approximate the properties of the liquid at 300 K [Sit & Marzari, 2005]. Hamann-Schluter-Chiang-Vanderbilt (HSCV) pseudopotentials [Hamann et al., 1979; Vanderbilt, 1985] were used to represent the core region of oxygen and hydrogen atoms. Among the possible ab initio molecular dynamics techniques, we have chosen the Car-Parrinello [Car & Parrinello, 1985]

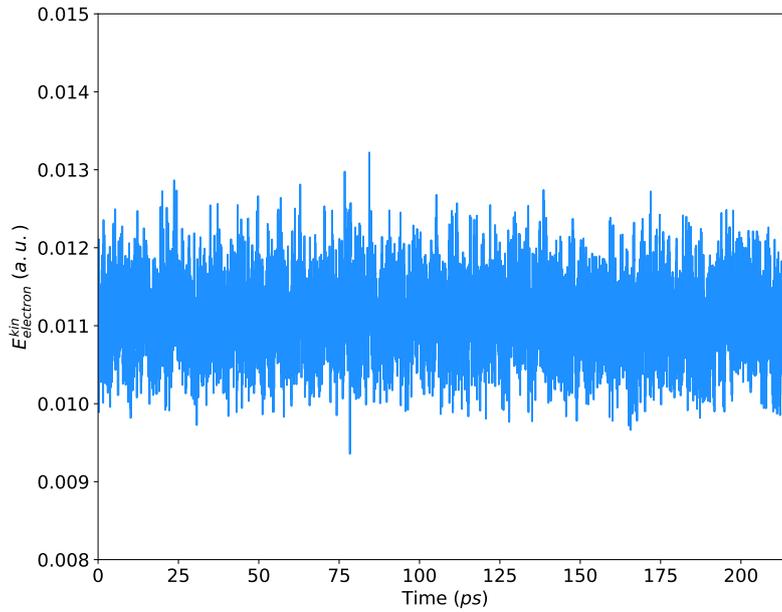


Figure 10: Fictitious energy of the electrons as a function of time.

method and all of our calculations were carried out with the plane-wave first-principles distributions of codes QUANTUM ESPRESSO [Giannozzi et al., 2009]. The plane-wave cutoff was set to 85Ry. The water model was chosen as a cubic super-cell with periodic boundary conditions containing 375 atoms (125 molecules of water), with a density $\rho = 1.0 \text{ g/cm}^3$.

As usual, a first NVT simulation of few picoseconds was carried out wherein the system was equilibrated at target temperature of 400K with a Nose-Hoover thermostat. Then we performed a NVE trajectory of about 200 ps in order to collect data. The mean temperature of the ions during the simulation was found to be $\langle T \rangle = 410 \pm 0.5 \text{ K}$. A time-step $\epsilon = 7.26 \cdot 10^{-2} \text{ fs}$ was used to integrate the electronic and ionic equations of motion; this value is about ten times smaller than the time step typically used in molecular dynamics. The fictitious energy of the electrons was chosen to be $\tilde{m} = 25m_e$.

In Fig. 10 we display the behaviour of the fictitious energy of the electron through the NVE simulation. The function oscillates with a maximum value of about $1.3 \cdot 10^{-2}$ Hartree and there is no indication of a systematic gain of kinetic energy of the electronic variables.

In Fig. 11 we show the different energies of the system as a function of the time. The green data represent the potential energy surface, V_{pes} , of the system, in red we plot the real energy of the whole system, formed by electron and ions, while in black we present the constant of motion of the Car-Parrinello Lagrangian (3.31). The almost constant behavior of the latter gives a check of the numerical accuracy in the integration of the equations of motions. Actually, very small fluctuations, due to the discretization of the time evolution are present, not visible on the scale of the picture since they correspond to relative variations of energy less than 10^{-4} Hartree. The difference between the conserved energy and the total energy of the system (respectively black and red data) corresponds to the fictitious energy of the electronic degrees, which is about six orders of magnitude smaller respect to the other.

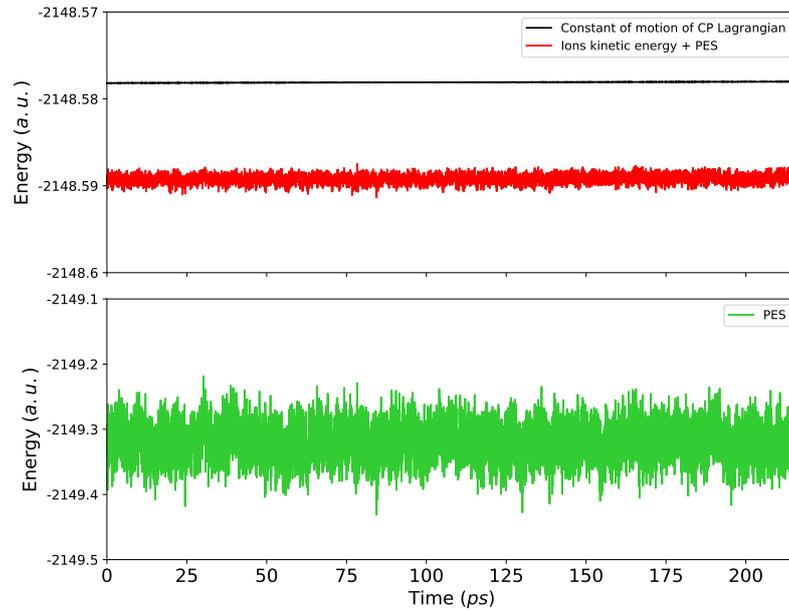


Figure 11: Behaviour of the energies during the CP dynamics. *Green:* the potential energy surface, i.e. the self consistent energy of the electrons within the density functional theory and the Born-Oppenheimer approximation. *Red:* the total energy of the system given by the sum of the PES and the kinetic energy of the ions. *Black:* the formal conserved energy of the Car-Parrinello Lagrangian.

3.4.1 Results and data analysis

After this short description of the simulation itself, we are now ready to discuss the shear viscosity of PBE water, following the scheme already presented in Chap. 2.

The collected data useful to the calculation of the shear viscosity consist of the time series of the three off-diagonal elements of the stress tensor, $\Pi_{\alpha\beta}$. We remind briefly the step of the data acquisition procedure and the subsequent analysis in order to evaluate the transport coefficient

- Compute the off-diagonal elements of the stress tensor: $\Pi_{xy}, \Pi_{xz}, \Pi_{yz}$;
- Compute the mean periodogram of the stress time series ${}^l\hat{S}_k$ from Eq. (2.37);
- Choose a cut-off frequency in order to get rid of high-frequency peaks, and resample the time series;
- Compute the cepstrum (2.43);
- Apply the Akaike Information Criterion (AIC), maximizing Eq. (2.54) in order to estimate the number of cepstral coefficients P^* ;
- Evaluate the shear viscosity and its statistical error from Eqs. (2.49);

The resulting mean periodogram is displayed in Fig. 12, and its moving average represented with the solid blue line. The features of the spectrum at low frequencies are similar to the one we already analysed for the

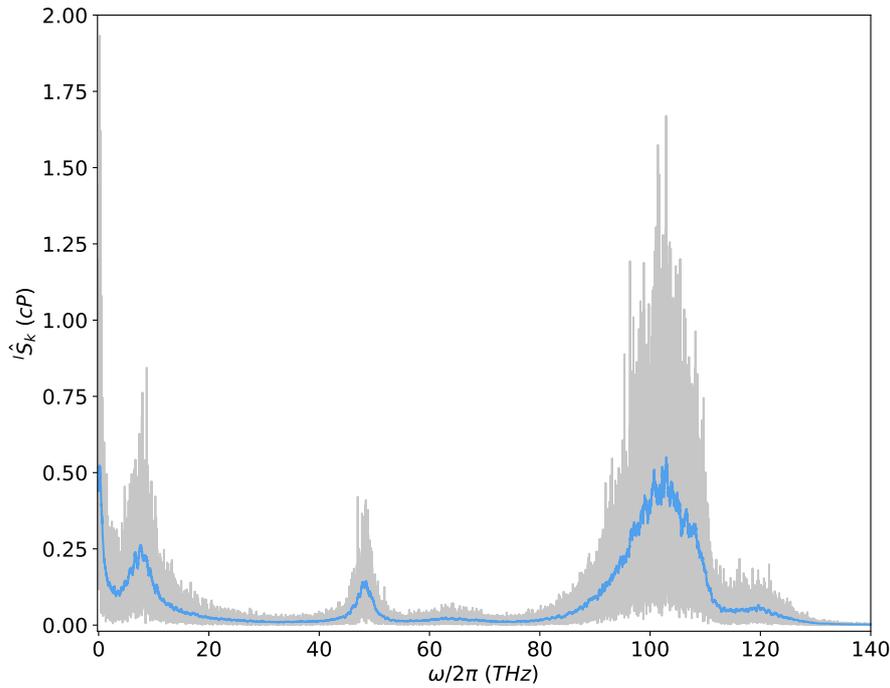


Figure 12: Mean periodogram of stress time series of water obtained from a 200 ps ab initio molecular dynamics trajectory. *Grey:* mean periodogram obtained directly for Eq. (2.38) with $l = 3$ (i.e. with the three equivalent components of the stress tensor). *Blue:* periodogram filtered with a moving average window of width 0.3 THz

TIP4P/2005 model in Fig. 3. However, the ab initio spectrum exhibits peaks also for higher frequencies, at approximate 48 THz and 102 THz.

The results carried out by the cepstral analysis technique is showed in Fig. 13. The cepstrum is represented by the red line showed in the inset of the graph. The periodogram exhibits a high peak very close to the zero frequency, followed by a second smaller peak at about 8 THz.

In order to justify our chosen value for the cut-off frequency, we report in Fig. 14 the behaviour of the shear coefficient with its standard deviation as a function of f^* . The transport coefficient is quite constant and fully compatible in the interval $[0, 30]$, which, looking at the full periodogram of the stress time series in Fig. 12, corresponds to the region that goes from zero till the first peak. After this first peak, the viscosity still doesn't show dramatic discontinuities, but the compatibility with the low-frequency values is less evident. In the inset of the plot we display a magnification of the low frequency region and also the dependency of the optimal number of cepstral coefficients, given by the Akaike Information Criterion, as a function of the cut-off frequency. We found that the growth of the number of P^* is almost linear (slightly sub-linear) with the choice of f^* . With the aim of obtaining a good estimate of the shear viscosity, choosing an f^* as one of the value in the interval $[0, 30]$ would be equivalent, but since we also want a small standard deviation, we chose as cut-off frequency the value $f^* \approx 12$ THz which is a good compromise between the estimation of the shear viscosity and its error.

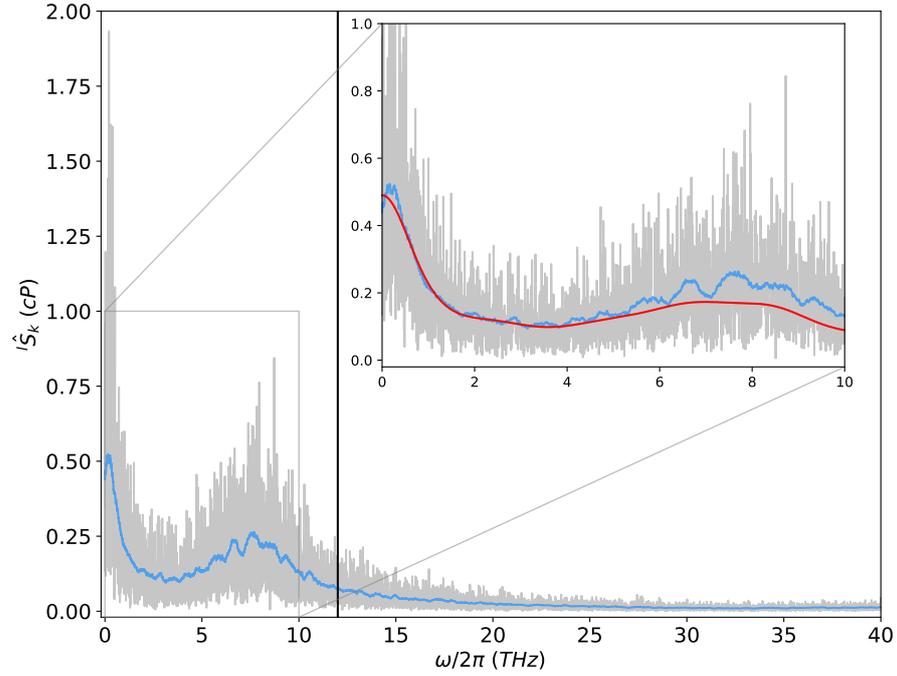


Figure 13: Cepstral analysis of stress time series of water obtained from a 200 ps molecular dynamics trajectory at 400K. *Grey:* mean periodogram. *Blue:* periodogram filtered with a moving average window of width 0.3 THz. The vertical line identifies the cut-off frequency f^* . The inset is a magnification of the low-frequency region with the result of the cepstral analysis in *red*.

Finally, we present the result (for the first time) of the shear viscosity of PBE water at about 410 K:

$$\eta = 0.488 \pm 0.029 \text{ cP} \quad (3.35)$$

with a relative error of the 6%. A previous ab initio shear viscosity for PBE water can be found in [Kuhne et al., 2009], however no previous results have ever been found with AIMD exploiting the Green-Kubo formalism. We compare our ab initio result with the experimental ones showed in Tab. 2, obtained from Ref. [Haynes et al., 2005]. Our result corresponds to the experimental value of liquid water at about 323 K, this discrepancy of about 100 K is quite in accord with previous studies [Grossman et al., 2004; Sit & Marzari, 2005].

As we can see from Tab. 2, experimental shear viscosity shows a strong dependency upon the temperature of the system. This implies that, in order to limit the bias introduced by temperature fluctuations on the value of

T(K)	273	283	293	303	313	323
η (cP)	1.793	1.307	1.002	0.798	0.653	0.547
T(K)	333	343	353	363	373	
η (cP)	0.467	0.404	0.354	0.315	0.282	

Table 2: Experimental values of the shear viscosity of water [2005].

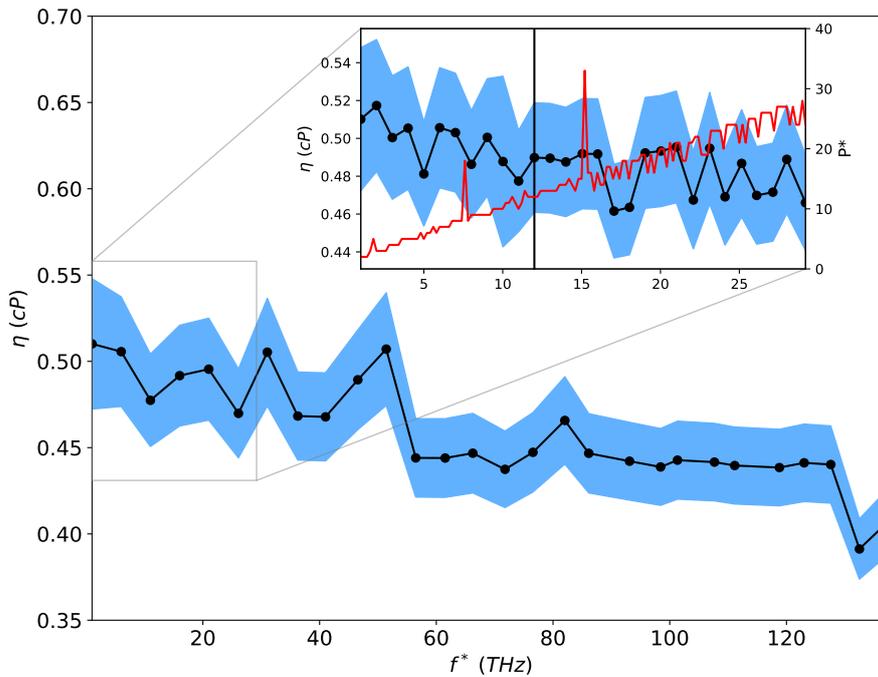


Figure 14: The graph displays the results obtained with cepstral analysis at different cut-off frequencies. The *blue* region represent the statistical error of each value of the shear viscosity while the *black dots* the actual values. The inset is a magnification of the first region the graph, with the addition, in *red*, of dependency of the optimal number of cepstral coefficients obtain from the AIC upon the f^* . The *vertical black line* indicates the cut-off frequency we've chosen in order to obtain the final estimate of the shear viscosity of water (3.35)

the viscosity, one has to carry out molecular dynamics simulations with a very stable temperature. In particular, in our the Car-Parrinello simulation of 125 water molecules, the standard deviation of the instant values of the temperature was founded to be about $\sigma_T = 15$ K. In order to reduce this fluctuations one can proceed in two different ways: a thermostat can be coupled to the dynamics of the ions, making sure that it does not affect the dynamics of the system, or equivalently one can simulate a larger system at constant energy, reducing in this way the amplitude of the temperature fluctuations. The latter choice is of course the safer, but the computational cost of ab initio molecular dynamics makes it unaffordable.

In the next chapter, we will try to overcome the impossibility of carrying out ab initio molecular dynamics simulations with large systems, exploiting neural network potentials. The latter will provide data that won't be distinguishable between the ab initio ones providing an accurate model to AIMD.

4

NEURAL NETWORK APPROACH TO MOLECULAR DYNAMICS

A lot of progress has been made in recent years in the development of atomistic potentials using machine learning (ML) techniques. Representing the interatomic potential energy surface (PES), both accurately and efficiently, is one of the most challenging problem in molecular modelling. Two are the mostly common traditional approaches: the direct application of quantum mechanics models such as DFT models, or empirically constructed potential models fitted from experimental data. The former approach is severely limited by the size of the system that one can handle, while as the latter class of methods are limited by the accuracy and the transferability of the model. Considering the importance of the PES in molecular modeling, a lot of interest has been recently put in order to provide a general framework for an ML-based PES that can equally describe different systems with high fidelity .

In this last chapter we focus our attention on a particular machine learning model called *Deep Potential - Smooth Edition* (DeepPot-SE) [Zhang, Han, Wang, Saidi, et al., 2018], which is able to efficiently represent the PES for a wide variety of systems with the accuracy of ab initio quantum mechanics models. In particular, a scheme for molecular simulation, the deep potential molecular dynamics (DPMD) [Zhang, Han, Wang, Car, et al., 2018] based on machine learning, will be exploited in order to obtain molecular dynamic simulations with inter-atomic forces predicted by a carefully crafted deep neural network trained with ab initio data.

In the first sections we will give a brief description of machine learning basic concepts [Goodfellow et al., 2016]. We will proceed describing the DeepPot-SE model [Zhang, Han, Wang, Saidi, et al., 2018] and how to use it in order to craft a neural network potential. Finally, we will present (for the first time) the result of the shear viscosity of water obtained from DPMD simulations trained on ab initio data.

4.1 MACHINE LEARNING BASICS

Let us begin with a general definition of a machine learning algorithm is: an algorithm that is able *to learn* from data. By learning we mean that these computer algorithms improve automatically through experience, in order to make predictions or decisions without being explicitly programmed to do so. This allows ML to tackle tasks that are too difficult to be solved with fixed programs written and designed by human beings. Many kind of tasks can be solved with machine learning (classification, transcription, machine translation, anomaly detection ...), here we focus our attention on one particular task: the *regression*. In a regression algorithm, the computer program is asked to predict a numerical value given some input. In order

to solve this task, the learning algorithm is asked to output a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$. From this point of view, this approach is then very similar to a fit: we have some data, we fit them with a specific function and then, given an input, we are able to provide some predictions.

Briefly, a regression algorithms usually work in the following way. A set of data, provided from the experience, is split in two subsets: the learning one and the test one. The first contains the data fed to the ML algorithm, the latter is used to check if the predictions that the program provide are correct. This kind of learning is called a *supervisor learning*.

4.1.1 Example: Linear Regression

To make more concrete the previous basic concepts we've just introduced, we present an example of a simple machine learning algorithm: the *linear regression*. Our goal is to build a model that can take a vector $\mathbf{x} \in \mathbb{R}^n$ as input, and can predict the value of a scalar $y \in \mathbb{R}$ as its output, which is a linear function of the input. Let us define the output predicted by our model as

$$\hat{y} = \mathbf{w}^T \mathbf{x}, \quad (4.1)$$

where $\mathbf{w} \in \mathbb{R}^n$ is a vector of parameters and the label T indicates the transpose. We can think of \mathbf{w} as a set of *weights* that determine how each single data x^i affects the prediction.

Suppose we have a dataset consisting in a matrix of m examples inputs and a relative vector that provide the correct value of y for each of these examples. Let us use this dataset as a testing set. We refer to the matrix of inputs as $\mathbf{X}^{(\text{test})}$ and to the vector of the corresponding targets as $\mathbf{y}^{(\text{test})}$:

$$\mathbf{X}^{(\text{test})} = (\mathbf{x}_1^T, \mathbf{x}_2^T, \dots, \mathbf{x}_m^T) \text{ where } \mathbf{x}_i = (x^1, x^2, \dots) \quad (4.2)$$

$$\mathbf{y}^{(\text{test})} = (y_1, y_2, \dots, y_m). \quad (4.3)$$

One way in order to control the performance of the model is to compute the *loss function* or *cost function*. The loss function is a fundamental tool in ML and must be chosen with care being the function that gives information of the model. A simple choice, in this case, is the mean squared error

$$\text{MSE}_{\text{test}} = \frac{1}{m} \sum_i \left(\hat{y}_i^{(\text{test})} - y_i^{(\text{test})} \right)^2, \quad (4.4)$$

where \hat{y}_i is the prediction that the model provide, given \mathbf{x}_i as input. This in some way measure the error of the model's predictions and it is equal to 0 when $\hat{\mathbf{y}}^{(\text{test})} = \mathbf{y}^{(\text{test})}$, i.e. the model provided the right predictions.

We need now an algorithm that improves the weights \mathbf{w} in order to reduce the value of the loss function MSE_{test} , while it is allowed to gain experience by observing a training set $(\mathbf{X}^{(\text{train})}, \mathbf{y}^{(\text{train})})$. An easy and intuitive way is just to minimize the $\text{MSE}_{\text{train}}$, the error on the training set, with respect to the weights. This can easily carried out solving the following problem

$$\nabla_{\mathbf{w}} \text{MSE}_{\text{train}} = 0, \quad (4.5)$$

that has solution

$$\mathbf{w} = \left(\mathbf{X}^{(\text{train})\text{T}} \mathbf{X}^{(\text{train})} \right)^{-1} \mathbf{X}^{(\text{train})} \mathbf{y}^{(\text{train})}. \quad (4.6)$$

Then evaluating the weights through this last equation constitutes a simple learning algorithm.

4.2 DEEP FEEDFORWARD NETWORKS

Deep feedforward networks [Goodfellow et al., 2016], also often called feedforward neural networks, are the fundamental pillars of deep learning models. These models are called *feedforward* because the information moves in only one direction, forward, from the input nodes, through the hidden nodes (if any) and to the output nodes. There are no cycles or loops in the network. The goal of a feedforward network is to approximate some function f^* . For example, for a regression, $y = f^*(\mathbf{x})$ maps an input vector \mathbf{x} to an output y . Then the feedforward network will define a map $y = f(\mathbf{x}; \mathbf{w})$ optimizing the values of the parameters \mathbf{w} in order to result in the best approximation function.

Deep feedforward neural networks are referred as *network* because they are typically constitutes of composition of many different functions. For example, we might have three functions $f^{(1)}, f^{(2)}, f^{(3)}$ connected in a chain, to form $f(\mathbf{x}) = f^{(3)}(f^{(2)}(f^{(1)}(\mathbf{x})))$. In this case, $f^{(1)}$ is called *first layer*, $f^{(2)}$ is called the *second layer* and so on. The number of function chained gives the *depth* of the network (this is where the term deep learning comes from). The final layer is called the *output layer*. When we feed a feedforward neural network with an input, we don't know how the layers individually behave and interact in order to provide the output. We do not decide the output of every different layer, but it is the learning algorithm itself that automatically decided how to use these layers to best implement an approximation of the f^* . Because the training data do not show the desired output for each of these layers, these layers are called *hidden layers*. The meaning of these intermediate outputs is often not totally understood.

Lastly, these networks are called *neural* because, at the very beginning, they were inspired by neuroscience. Each hidden layer of the network is typically vector-valued, and its dimensionality determines the *width* of the model. An hidden layer receives an input vector \mathbf{x}^{in} with n entries and gives back an output vector \mathbf{x}^{out} with m entries, acting then like a matrix $\mathbf{W} : M \times N$. We can think of the layer as consisting of many units that act in parallel, each representing a vector-to-scalar function. These units (often referred as nodes) resembles the different neurons in the sense that they receive an input from many other units (the entries of the vector \mathbf{x}^{in}) and the compute a scalar output (one of the components of \mathbf{x}^{out}). Each neuron is then the representation of one row of the matrix \mathbf{W} , and the number of rows of the latter corresponds to the number of neurons for that particular layer (i.e. the width of the layer).

Let us give a simple mathematical interpretation for the neuron. In Fig. 15 we display the way in which a usual neuron is represented. A vector

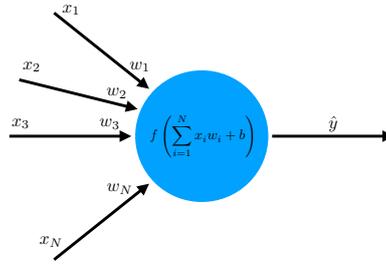


Figure 15: The neuron mathematical model of McCulloch and Pitts. In this particular model the activation function is a Heaviside function.

$\mathbf{x} = (x_1, x_2, \dots, x_N)$ is the input signal and enters the neuron with weights $\mathbf{w} = (w_1, w_2, \dots, w_N)$. Inside the neuron, the sum of the weighted entries is calculated, given $a = \sum_i x_i w_i + b$ where b is a bias. This result is used as argument of the *activation function* $f(z)$, a non linear function that finally gives the value of the output y . In order to describe the biological behaviour of a neuron, McCulloch and Pitts in the 1943 [Fitch, 1944] chose as activation function the Heaviside function, given as output

$$y = \Theta \left(\sum_{i=1}^N x_i w_i + b \right) = \Theta(a), \quad (4.7)$$

in order to describe neuron with binary response. The neuron is say to be *activated* if $a > 0$. Nowadays, other activation functions, like *hyperbolic functions*, *rectified linear functions* (ReLU) and *emphsigmoids*, have become more popular.

The example we presented is simple and trivial, but it's the fundamental unit of a feedforward neural network. In these models every layer can present a large number of neurons. Each of these neurons takes a vector as input and give back a scalar output, the latter will be fed again to another neurons, together with the output of the other neurons of that layer. This process will be repeated for every layer of the network, resulting in a almost chaotic sequence of operations.

For a better understanding, let us consider a single output feedforward neural network with L layers of neurons. We call \mathbf{x} the input vector and we refer as $\mathbf{o}^{(l)}$ the output of the l -th neuron layer, clearly $\mathbf{o}^{(0)} = \mathbf{x}$ is the input vector and $\mathbf{o}^{(L)} = \hat{y}$ is th output scalar value. For simplicity, let us restrict to the case $L = 3$ and we suppose each neuron in the same l -th layer to have the same activation function $f^{(l)}$. A schematic representation of the deep feedforward network is given in Fig. 16. Let us define

$$\mathbf{a}_k^{(l)} = \sum_j w_{kj}^{(l)} o_j^{(l-1)} + b_k^{(l)}, \quad o_k^{(l)} = f^{(l)} \left(\mathbf{a}_k^{(l)} \right), \quad (4.8)$$

where $w_{kj}^{(l)}$ and $b_k^{(l)}$ are respectively the weight and the bias for the output of the j -th neuron (of layer $l-1$) entering the k -th neuron of the layer l , while $o_k^{(l)}$ is the scalar output of the k -th neuron in the l -th layer. To simplify the mathematics further, the bias $b_k^{(l)}$ for node k in layer l will be incorporated into the weights as $w_{k0}^{(l)}$ with a fixed output of $o_0^{(l-1)} = 1$ for node 0 in layer $l-1$.

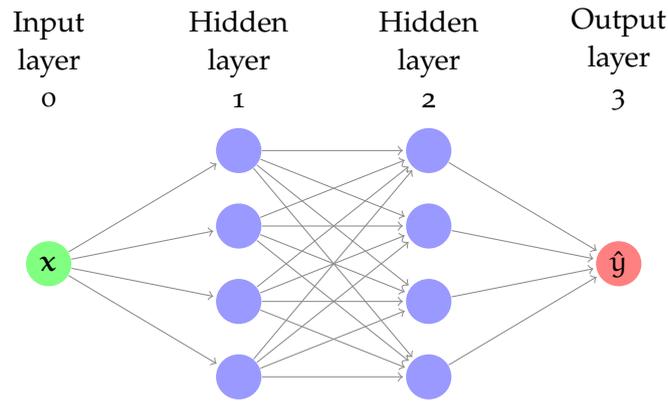


Figure 16: A simple feedforward neural network with two hidden layer

The final output value \hat{y} can be then written as

$$\begin{aligned} \hat{y} &= o_1^{(3)} = f^{(3)}(z_1^{(3)}) \\ &= f^{(3)}\left(\sum_i w_{1i}^{(2)} f^{(2)}\left(\sum_j w_{ij}^{(1)} f^{(1)}\left(\sum_k w_{jk}^{(0)} x_k^{(0)}\right)\right)\right), \end{aligned} \quad (4.9)$$

and we can again, given a training dataset of M vectors \mathbf{x}_i and their relative true associated output y_i , define the loss function as

$$\mathcal{L} = \frac{1}{M} \sum_{i=1}^M (y_i - \hat{y}_i)^2 = \frac{1}{M} \sum_{i=1}^M L_i, \quad (4.10)$$

and can be easily generalized for an arbitrary number of layers.

The learning process, which consists in minimizing the loss function (4.10), is powered by two fundamental algorithms: the stochastic gradient descent that provides a useful tool in order to manage the dataset to train the network, and the back-propagation that allows to compute the gradient of the loss function in an easy and computationally cheap way. We will end this brief introduction describing these two algorithms.

4.2.1 Stochastic gradient descent

Almost all deep learning algorithms are powered by one very important algorithm: *stochastic gradient descent* (STG) [Goodfellow et al., 2016]. This algorithm is an extension of the well-known gradient descent algorithm, since it replaces the actual gradient (calculated from the entire data set) by an estimate thereof (calculated from a randomly selected subset of the data). In the present case the function we want to minimize is the loss function (4.10), and in order to obtain a solid neural network, capable of providing good predictions from previously unobserved inputs (this ability is called *generalization*), large training sets are necessary. But large training sets are also more computationally expensive.

As we've seen before the loss function can be decomposed as a sum over training example of some per-example loss function

$$\mathcal{L}(\mathbf{w}) = \frac{1}{M} \sum_{i=1}^M L(\mathbf{x}_i, \mathbf{y}_i, \mathbf{w}) = \frac{1}{M} \sum_{i=1}^M L_i, \quad (4.11)$$

where L is the per-example loss function.

In order to minimize the loss function, gradient descent requires computing

$$\nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w}) = \frac{1}{M} \sum_{i=1}^M \nabla_{\mathbf{w}} L_i, \quad (4.12)$$

and the cost of this operation is $O(m)$, turning out being unaffordable for typical size of training dataset, made of millions to billions of examples.

The stochastic gradient descent provides a interesting alternative. The gradient of the loss function is approximately estimated using small set of samples called *minibatch*. In practice, on each step of the algorithm we can sample a minibatch of examples $\mathbb{B} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{M'}\}$ drawn uniformly from the training set. The minibatch size M' is chose drastically smaller than the size of the entire dataset, ranging from 1 to few hundreds.

The estimate of the gradient is formed as

$$\mathbf{g} = \frac{1}{M'} \sum_{i=1}^{M'} \nabla_{\mathbf{w}} L_i, \quad (4.13)$$

using data from the minibatch \mathbb{B} . The new optimized parameters $\mathbf{w}^{(new)}$ are then given following the estimated gradient downhill:

$$\mathbf{w}^{(new)} = \mathbf{w}^{(new)} - \epsilon \mathbf{g}, \quad (4.14)$$

where ϵ is the step size, called the *learning rate*.

4.2.2 Back-propagation algorithm

The stochastic gradient descent allows us to save a significant amount of computational time, but still we need to explicitly compute the gradient of the loss function with respect to a relevant number of parameters. This made the computation of the gradient for a neural network slightly more complicated. Nevertheless, the *back-propagation algorithm* allows to do it efficiently and exactly [2016].

In fitting a neural network, the back-propagation algorithm computes the gradient of the loss function with respect to the weights of the network for a single example. The algorithm works by computing the gradient with respect to each weight exploiting the chain rule for derivatives, iterating backward from the last layer to the first. It is called back-propagation because it works in the opposite way of the natural forward propagation of the information in a feedforward neural network.

Let us try to describe the algorithm. We want to evaluate the gradient of the loss function (4.10) with respect to a particular weights $w_{kj}^{(l)}$. Referring

to (4.8), $w_{kj}^{(l)}$ is the weight that multiplies the output of the j -th neuron of the layer $l-1$, $o_j^{(l-1)}$, and contribute to $a_k^{(l)}$, the argument of the activation function of the k -th neuron of the l -th layer. We then write explicitly this derivative and, using the chain rule, we obtain

$$\frac{\partial \mathcal{L}}{\partial w_{kj}^{(l)}} = \frac{\partial \mathcal{L}}{\partial a_k^{(l)}} \frac{\partial a_k^{(l)}}{\partial w_{kj}^{(l)}} = \frac{1}{M} \sum_{i=1}^M \frac{\partial L_i}{\partial a_k^{(l)}} \frac{\partial a_k^{(l)}}{\partial w_{kj}^{(l)}}, \quad (4.15)$$

where $a_k^{(l)}$ is defined in (4.8) and L_i is what we previously called the per-example loss function. For simplicity, let us drop the index i . The first term in the sum is denoted as

$$\delta_k^{(l)} = \frac{\partial L}{\partial a_k^{(l)}}, \quad (4.16)$$

and can be interpreted as the error at layer l . The second term can be explicitly evaluated and it gives

$$\frac{\partial a_k^{(l)}}{\partial w_{kj}^{(l)}} = \frac{\partial}{\partial w_{kj}^{(l)}} \left(\sum_{n=1}^{r_{l-1}} w_{kn}^{(l)} o_n^{(l-1)} \right) = o_j^{(l-1)}, \quad (4.17)$$

where r_{l-1} is the number of neurons in the layer $l-1$.

Thus, the partial derivative of the loss function with respect to the weight $w_{kj}^{(l)}$ returns

$$\frac{\partial L}{\partial w_{kj}^{(l)}} = \delta_k^{(l)} o_j^{(l-1)}. \quad (4.18)$$

Starting from the final layer L , let's give an expression for $\delta_1^{(L)}$ (the subscript is 1 and not k because this derivation concerns a one-output neural network). The error at layer L can be evaluated as follow

$$\delta_1^{(L)} = \frac{\partial L}{\partial a_1^{(L)}} = \frac{\partial}{\partial a_1^{(L)}} \left[f^{(L)}(a_1^{(L)}) - y \right]^2 \quad (4.19)$$

$$= 2 \left[f^{(L)}(a_1^{(L)}) - y \right] f'^{(L)}(a_1^{(L)}), \quad (4.20)$$

where the prime indicates the derivative of the function with respect to its argument. Putting all together, the derivative of the loss function with respect to a weight in the final layer $w_{1k}^{(L)}$ is

$$\frac{\partial L}{\partial w_{1j}^{(L)}} = \delta_1^{(L)} o_j^{(L-1)} = 2(\hat{y} - y) f'^{(L)}(a_1^{(L)}) o_j^{(L-1)}. \quad (4.21)$$

The partial derivatives with respect to any other weight of hidden layers is a bit more complicated. Let us write the error of the hidden layer l in the following way

$$\delta_k^{(l)} = \frac{\partial L}{\partial a_k^{(l)}} = \sum_{n=1}^{r_{(l+1)}} \frac{\partial L}{\partial a_n^{(l+1)}} \frac{\partial a_n^{(l+1)}}{\partial a_k^{(l)}} = \sum_{n=1}^{r_{(l+1)}} \delta_k^{(l+1)} \frac{\partial a_n^{(l+1)}}{\partial a_k^{(l)}}. \quad (4.22)$$

Note that the bias input $o_0^{(l)}$ corresponding to $w_{k0}^{(l+1)}$ is fixed and do not enter in the sum since its value is not dependent on the outputs of previous

layers. The second term of the sum can be evaluated exploiting the definition of $a_i^{(l+1)}$:

$$a_n^{(l+1)} = \sum_{j=1}^{r_l} w_{nj}^{(l+1)} f^{(l)}(a_j^{(l)}) \longrightarrow \frac{\partial a_n^{(l+1)}}{\partial a_k^{(l)}} = w_{nk}^{(l+1)} f'^{(l)}(a_k^{(l)}). \quad (4.23)$$

Plugging this expression in (4.22) gives

$$\delta_k^{(l)} = f'^{(l)}(a_k^{(l)}) \sum_{n=1}^{r_{(l+1)}} \delta_k^{(l+1)} w_{nk}^{(l+1)}, \quad (4.24)$$

which is called *backpropagation formula*. Putting all together, the derivative of the loss function (4.15) finally gives

$$\frac{\partial \mathcal{L}}{\partial w_{kj}^{(l)}} = \delta_k^{(l)} o_j^{(l-1)} = f'^{(l)}(a_k^{(l)}) o_j^{(l-1)} \sum_{i=1}^{r_{(l+1)}} \delta_k^{(l+1)} w_{ik}^{(l+1)}. \quad (4.25)$$

This last equation is where backpropagation gets its name. In particular we see, Eq. (4.24), that the error $\delta_k^{(l)}$ at layer l is dependent on the error $\delta_k^{(l+1)}$ at the next layer $l+1$. One then proceed in the following way first. First, the forward phase is carried out, evaluating all the outputs of the different layer of the network, the different input of each neurons and the resultant loss function; then one can proceed backward, starting from evaluating the error at the output layer and going back. Once we have collected all the errors, through (4.25), the derivative of the loss function with respect to any weights is easily carried out.

This algorithm, flanked by the stochastic gradient descent, allows to train efficiently a deep feedforward neural network.

4.3 DEEP POTENTIAL - SMOOTH EDITION

After this brief introduction in machine learning basic topics, let us go back to physics. The *Deep Potential - Smooth Edition* [Zhang, Han, Wang, Saidi, et al., 2018] exploits machine learning tools in order to model the PES of an interacting system made of electrons and nuclei. The model, which naively consists in a deep feedforward neural network, can be trained with ab initio data allowing to predict the total energy of the system and the forces acting on each atom with quantum accuracy. By construction, DeepPot-SE is extensive and continuously differentiable, scales linearly with system size, and preserves all the natural symmetries of the system. Imposing the symmetries of the system, such as rotational and translational symmetry, to the model is not indispensable since one could let the neural network learn all this properties by itself. Anyway, forcing the model to present these symmetry properties reduces the time one has to spend to train it in order to have satisfactory predictions.

Let us give a brief description of this model. We consider a system of N atoms and we define the *coordinate matrix* $\mathcal{R} \in \mathbb{R}^{N \times 3}$ as

$$\mathcal{R} = \begin{pmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ \vdots & \vdots & \vdots \\ x_N & y_N & z_N \end{pmatrix}, \quad (4.26)$$

where $\mathbf{r}_i = (x_i, y_i, z_i)$ is the position of the i -th atom. As we've previously seen, the PES, $V_{\text{pes}}(\mathcal{R})$, is a function that maps the atomic coordinates and their chemical character to a real number.

The DeepPot-SE model, exploiting the extensivity of the total energy, works decomposing the PES in atomic contributions ε_i that depend on particular local atomic descriptors. The latter contain information of the local environment of each atom and its species. In order to construct these descriptors, let r_c be a pre-defined cut-off radius. For each atom i , we consider its neighbors list given by the ensemble of the atoms whose distance from the atom i is smaller than the cut-off distance: $r_{ij} < r_c$. Let N_i be the number of atom of this list, we can describe the local environment of atom i with the *relative coordinate matrix* $\mathcal{R}^i \in \mathbb{R}^{N_i \times 3}$ in terms of Cartesian coordinates

$$\mathcal{R}^i = \begin{pmatrix} x_{1i} & y_{1i} & z_{1i} \\ x_{2i} & y_{2i} & z_{2i} \\ \vdots & \vdots & \vdots \\ x_{N_i i} & y_{N_i i} & z_{N_i i} \end{pmatrix}, \quad (4.27)$$

where $\mathbf{r}_{ji} = \mathbf{r}_j - \mathbf{r}_i = (x_{ji}, y_{ji}, z_{ji})$ are defined as relative coordinates.

Now, in order to make the model smooth, the relative coordinates matrices $\mathcal{R}^i \in \mathbb{R}^{N_i \times 3}$ are mapped into generalized coordinates $\tilde{\mathcal{R}}^i \in \mathbb{R}^{N_i \times 4}$. The latter are referred as *renormalized coordinate matrix*:

$$\tilde{\mathcal{R}}^i = \begin{pmatrix} s(r_{1i}) & \frac{s(r_{1i})x_{1i}}{r_{1i}} & \frac{s(r_{1i})y_{1i}}{r_{1i}} & \frac{s(r_{1i})z_{1i}}{r_{1i}} \\ s(r_{2i}) & \frac{s(r_{2i})x_{2i}}{r_{2i}} & \frac{s(r_{2i})y_{2i}}{r_{2i}} & \frac{s(r_{2i})z_{2i}}{r_{2i}} \\ \vdots & \vdots & \vdots & \vdots \\ s(r_{N_i i}) & \frac{s(r_{N_i i})x_{N_i i}}{r_{N_i i}} & \frac{s(r_{N_i i})y_{N_i i}}{r_{N_i i}} & \frac{s(r_{N_i i})z_{N_i i}}{r_{N_i i}} \end{pmatrix}, \quad (4.28)$$

where $s(r_{ji})$ is a continuous and differentiable scalar weighting function defined as

$$s(r_{ji}) = \begin{cases} \frac{1}{r_{ji}} & r_{ji} < r_{cs} \\ \frac{1}{r_{ji}} \left[\frac{1}{2} \cos \left(\pi \frac{(r_{ji} - r_{cs})}{(r_c - r_{cs})} \right) + \frac{1}{2} \right] & r_{cs} < r_{ji} < r_c \\ 0 & r_{ji} > r_c \end{cases} \quad (4.29)$$

that plays the role of a smoothing function applied to each component. Here r_{cs} is a smooth cut-off parameter that allows the components of $\tilde{\mathcal{R}}^i$ in to smoothly go to zero at the boundary of the local region defined by r_c . In particular $s(r_{ji})$ plays the role of a smoothing function, reducing the weight of the particles that are more distant from atom i . It also removes the discontinuity introduced by the cut-off radius r_c .

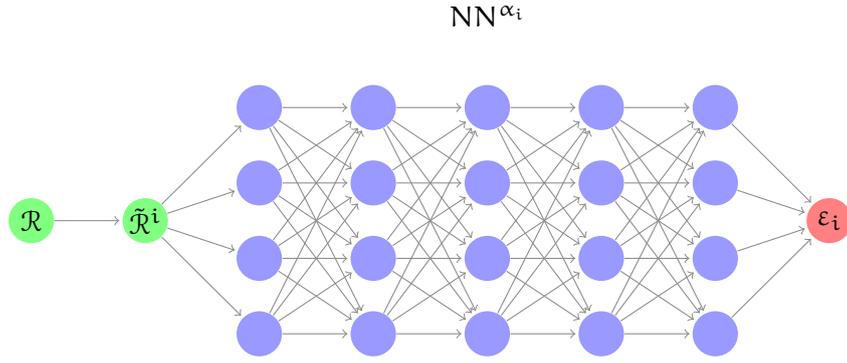


Figure 17: A pictorial representation of the Deep Potential-SE scheme. The coordinate matrix \mathcal{R} is firstly transformed in the renormalized coordinate matrix $\tilde{\mathcal{R}}^i$ which describes the local environment of the atom i . The letter is then fed to a feedforward neural network NN^{α_i} , specific for the atomic species α_i of the atom i . The network gives as result a scalar number trained to be the local atomic energy.

Then, each matrix $\tilde{\mathcal{R}}^i$ is fed into a specific deep feedforward network NN^{α_i} , where α_i specifies the species of the atom i . The output ε_i of the neural network is the local atomic energy of the atom i . Then the PES is given by the sum of the local atomic energies

$$V_{\text{pes}} = \sum_{i=1}^N \varepsilon_i, \quad (4.30)$$

while the atomic forces are given by

$$\mathbf{F}_n = -\nabla_n V_{\text{pes}} = -\sum_{i=1}^N \nabla_n \varepsilon_i, \quad (4.31)$$

The training is performed with the stochastic descent method, minimizing the loss function with respect to the weights of the deep feedforward network. The evaluation of the gradient is carried out using backpropagation algorithm, while the loss function is chosen to be

$$\mathcal{L} = \frac{1}{M} \sum_{l=1}^M p_E |V_{\text{pes}} - E^l|^2 + p_F |\mathbf{F} - \mathbf{F}^l|^2, \quad (4.32)$$

where M is the number of examples in the minibatch used for the STG method, E^l and \mathbf{F}^l are the predicted energies and total forces, and p_E, p_F are tunable prefactors useful to improve the efficiency of the training.

Let us notice that the evaluation of the atomic forces is straightforward and it is given 'for free' by the backpropagation algorithm. In fact we can write each term in the sum (4.31)

$$\nabla_n \varepsilon_i = \frac{\partial \varepsilon_i}{\partial \mathbf{r}_n} = \frac{\partial \varepsilon_i}{\partial \tilde{\mathcal{R}}_i} \frac{\partial \tilde{\mathcal{R}}_i}{\partial \mathbf{r}_n}, \quad (4.33)$$

where the first term is given by the backpropagation, while the second can be computed easily resulting in an analytic expression for the atomic forces.

4.4 SIMULATIONS

The DeepPot-SE model that computes the atomic interactions of the system can be built, trained and used for molecular dynamics simulations with the *DeepMD-kit*. This kit provides a package written in Python/C++, designed to minimize the effort required to build deep learning based model of interatomic potential energy and force field and to perform molecular dynamics (MD).

The typical procedure of using DeepPMD-kit is

- *Prepare data.* In order to train the model, one needs to provide the following information: the atoms type, the simulation box, the atoms coordinates, the atomic forces, and the system energy. All this information are taken from Car-Parrinello molecular dynamic simulations which provide a snapshot of the system dynamics at each step of the trajectory. Thus, the training energy and forces are evaluated from ab initio simulations, aiming to build a neural network able to give results indistinguishable from the original data and then able to reproduce ab initio molecular dynamics.
- *Build the model.* The deep feedforward neural network and the input matrix descriptors has to be build. In order to do build the descriptors, one needs to specify the two cut-off r_c and r_{cs} , the dimension of the renormalized coordinate matrix (i.e. the maximum possible number of neighbors). Then one need to specifies the features of the fitting deep feedforward network, such as the number of layers and the number of neuron for each layer. A seed is needed in order to initialize the starting weights of the net.
- *Train the model.* Given the model and the descriptors, the training of the net can be carried out. The parameter of the loss function (4.32), p_E and p_F , has to be fixed in order to give the desired weight of the energy and forces. These parameters can also be chosen to change during the training with some function of the training step. Finally, given the size of the minibatch for the STG and the size of the testing data set, useful to keep under control the training and avoid overfitting, the training can start.
- *Freeze the model.* Once the training is finished, usually at least after one million of steps, i.e. STG iteration, the model can be "freezed" and the final optimized parameters of the net are written in an output file.
- *Test the model.* The neural network potential can be tested comparing the results of DFT calculation with the ones predicted by the net on new snapshots.
- *Run MD.* If the testing provides successfull results, the model is ready and one can perform molecular dynamic simulations using the neural network potential in order to predict the interatomic forces.

Let us now give some information for what concern the neural network potential we used in order to carried out the shear viscosity of water. The

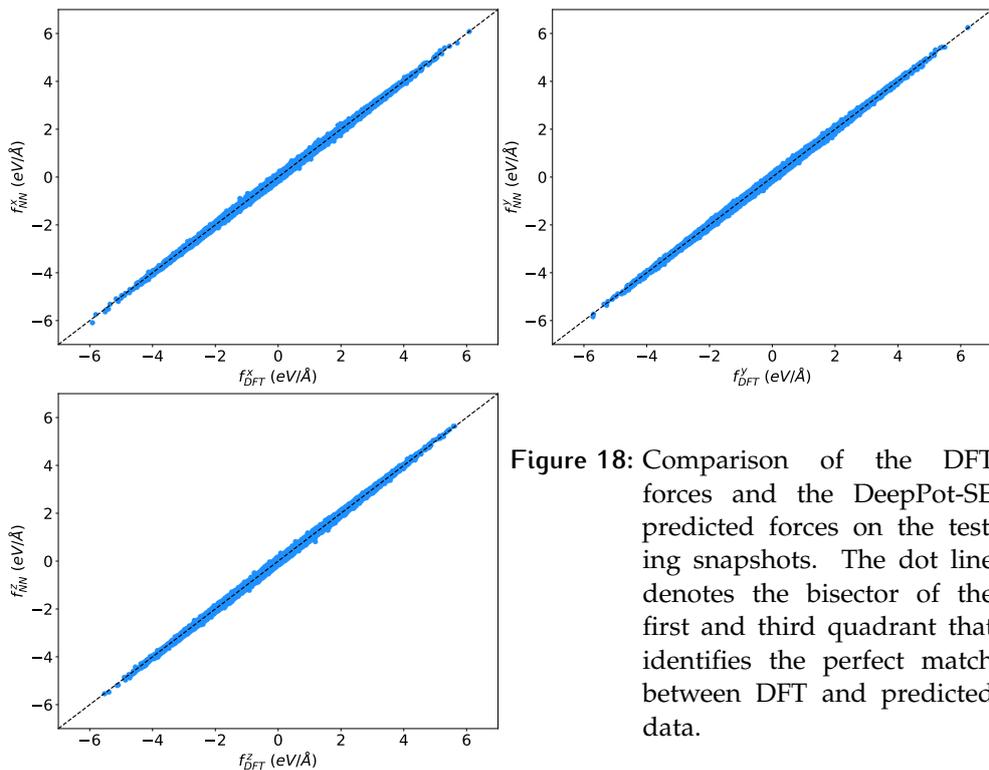


Figure 18: Comparison of the DFT forces and the DeepPot-SE predicted forces on the testing snapshots. The dot line denotes the bisector of the first and third quadrant that identifies the perfect match between DFT and predicted data.

training dataset consists of several millions of snapshots collected from Car-Parrinello molecular dynamics simulations of PBE water, run at different temperatures: 400, 430, 500 K. Using snapshots at different temperatures improves the stability and the reliability of the model and avoids annoying instabilities during the simulations. Instabilities may be due to thermal fluctuations that in the simulations can lead to unusual atomic dispositions, that differ remarkably from the training dataset. When the network is fed with these unusual snapshots, the predicted atomic forces can be wrong and tend lead to a dramatically wrong dynamic.

The parameters that controls the weights of the energy and forces in the loss function (4.32) were chosen to asymmetrical at the beginning of the training, giving more weight to the forces, and ending up in the opposite way, giving more weight to the energy. Also the learning rate defined in (4.14) was chosen to decrease exponentially during the training in order to reach the minimum more efficiently. The size of the minibatch used in the stochastic gradient descent was less than ten snapshots. The training was carried out for millions of iteration, feeding the network with several hundreds of thousands of snapshots

In order to preliminary test the network, let us check the quality of the predicted forces on 1000 different snapshots taken from a NVE ab initio trajectory of a PBE water system made of 125 molecules at about 400 K. To compare these two results, in Fig. 18 we plot the points $(f_{\text{DFT}}, f_{\text{NN}})$ whose coordinates are respectively the true DFT forces and the predicted ones given by the neural network DeePot-SE model. The plot exhibits a good agreement between the two results.

4.4.1 Validation of the neural network potential

Before focusing the attention on the shear viscosity of the neural network water model, one needs to test not only the accuracy of the predicted forces, but also the static and dynamical properties of the simulated system. From this perspective, we simulated a system of 125 water molecules at density $\rho = 1.0 \text{ g/cm}^3$. The temperature of the simulation was chosen to be about 400 K in order to match with the ab initio simulation we've already carried out for the shear viscosity calculation. The molecular dynamics simulation were, again, performed using the LAMMPS package for molecular dynamics [Plimpton, 1995], where the neural network potential provided by the DeepMD-kit is already interfaced with the MD code. From the simulation, we evaluate some structural properties such as the radial distribution function and simple dynamical properties such as the mean square displacement of the atoms and the vibrational spectrum, i.e. the power spectrum of the velocity. Numerical results for these properties can be compared with the ab initio ones and thus serve as a criterion to test the neural network model.

Let us start discussing the radial distribution function. The radial distribution gives fundamental information about the structure of a system. For a simple mono-atomic fluid the radial distribution function, $g(r)$, is defined as [Allen & Tildesley, 2017]

$$g(r) = \rho^{-2} \left\langle \sum_i \sum_{j(\neq i)} \delta(\mathbf{r}_i) \delta(\mathbf{r}_j - \mathbf{r}) \right\rangle = \frac{V}{N^2} \left\langle \sum_{j \neq i} \delta(r - |\mathbf{r}_{ij}|) \right\rangle, \quad (4.34)$$

where $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ is the distance between atom i and atom j , and the dependence of the function just over the modulus of the distance between two atoms is due to the isotropy of the fluid.

The $g(r)$ of a fluid system provides insight into the liquid structure, characterizing its local structure. It also provides preliminary information about the phase of the simulated system.

Since water is a diatomic fluid, we can evaluate different radial distribution functions:

$$g_{OO}(r) \propto \left\langle \sum_{i=1}^{N_{\text{oxy}}} \sum_{j=1(\neq i)}^{N_{\text{oxy}}} \delta(r - |\mathbf{r}_i^o - \mathbf{r}_j^o|) \right\rangle, \quad (4.35)$$

$$g_{HH}(r) \propto \left\langle \sum_{i=1}^{N_{\text{hyd}}} \sum_{j=1(\neq i)}^{N_{\text{hyd}}} \delta(r - |\mathbf{r}_i^h - \mathbf{r}_j^h|) \right\rangle, \quad (4.36)$$

$$g_{OH}(r) \propto \left\langle \sum_{i=1}^{N_{\text{oxy}}} \sum_{j=1}^{N_{\text{hyd}}} \delta(r - |\mathbf{r}_i^o - \mathbf{r}_j^h|) \right\rangle, \quad (4.37)$$

where $N_{\text{oxy}}, N_{\text{hyd}}$ are respectively the number of oxygen atoms and hydrogen atoms of the system, and $\mathbf{r}_i^o, \mathbf{r}_i^h$ are the positions of the oxygen and hydrogen atom i .

The oxygen-oxygen radial distribution function, $g_{OO}(r)$, and the hydrogen hydrogen radial distribution function, $g_{HH}(r)$, provide the surrounding environment of the oxygen/hydrogen atom with respect to the other

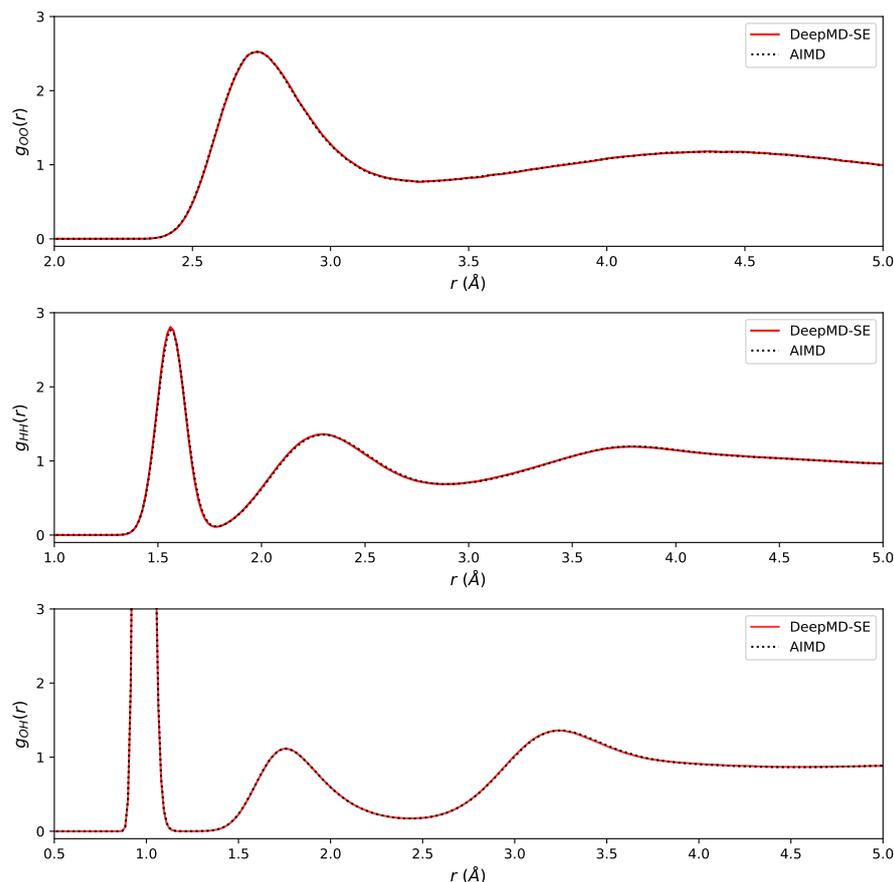


Figure 19: O–H (bottom), H–H (middle) and O–O (top) radial distribution functions for water as estimated from ab initio simulation (*black dotted line*) are compared with the results obtained using the DeepMD-SE potential (*red solid line*) to model the inter-atomic interactions in classical simulations.

oxygen/hydrogen atoms. The oxygen-hydrogen radial distribution function, $g_{\text{OH}}(r)$, gives information about the length of the oxygen-hydrogen bond.

In Fig. 19 we present the three different radial distribution functions for water evaluated from ab initio simulation (black dotted line) and using the DeepMD-SE potential (red solid line) trained on ab initio data in order to model the interactions. The two approaches give the same results with good precision and no differences are visible on the scale of the plot. In particular, the position of the first peak in the oxygen-oxygen radial distribution function, corresponding to the position the first oxygen atoms neighbours, matches the experimental one for ambient water at 2.8 \AA , measured through x-ray diffraction [Skinner et al., 2013].

The OH bond length for the isolated water molecule is known accurately from microwave and infrared spectroscopies [Cook et al., 1974] and turns out to be equal to 0.97 \AA , pretty close to the first peak of $g_{\text{OH}}(r)$ in Fig. 19. We conclude that the water DeepMD-SE potential that we trained starting from ab initio data, reproduces the same structural properties of the original ab initio simulation. Moreover, the PBE water at about 400 K and with a density of 1.0 g/cm^3 has after all the same main features of the experimental ambient water in the local structure.

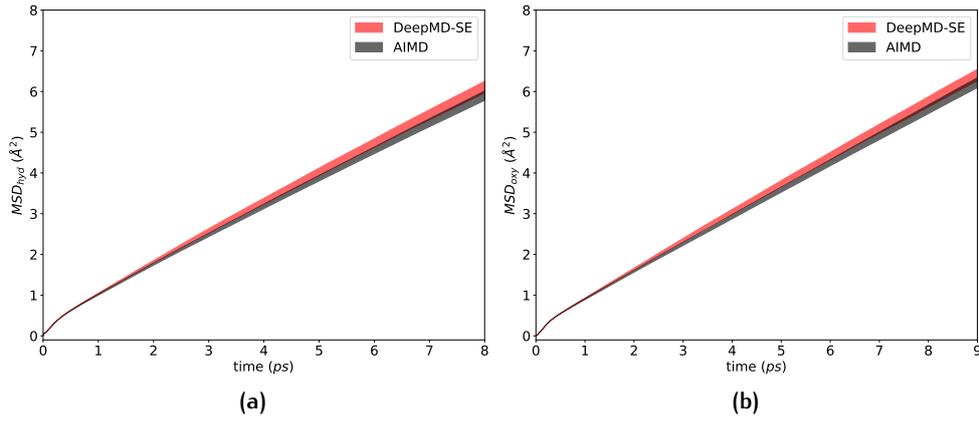


Figure 20: Mean square displacement (MSD) of hydrogen atoms (*left*) and oxygen atoms (*right*) of water molecules versus time. *Black:* results obtained from the ab initio NVE simulation. *Red:* results obtained from classical simulation with the neural network potential.

Let us now compare some basic dynamical properties of the two models. We start from the mean square displacement (MSD) of the atoms. The MSD is a measure of the deviation of the position of a particle with respect to a reference position over time. It is given by [Frenkel & Smit, 2001]

$$\text{MSD}(t) = \frac{1}{N} \sum_{i=1}^N \langle |\mathbf{r}_i(t) - \mathbf{r}_i(0)|^2 \rangle, \quad (4.38)$$

where \mathbf{r}_i is the position vector of the atom i , N is the number of atom of the system, and the brackets $\langle \cdot \rangle$ denotes the usual ensemble average.

In Fig. 20 we show the results of the evaluation of the mean square displacement respectively for the hydrogen atoms (left plot) and oxygen atoms (right plot), as a function of time. The two molecular dynamics approaches, AIMD in black and DeepMD-SE, provides slightly different results: both the neural network's curve present a steeper slope than the ab initio ones. Nevertheless, the results appear to be still compatible.

We end up the validation of the the neural network potential, discussing the VDOS of the system. The vibrational spectrum is given by

$$D(\nu) = \frac{1}{3} \int_0^{\infty} \text{VAC}(t) e^{i2\pi\nu t} dt, \quad (4.39)$$

where

$$\text{VAC}(t) = \frac{1}{N} \sum_{i=1}^N \langle \mathbf{v}_i(t) \mathbf{v}_i(0) \rangle, \quad (4.40)$$

is the velocity time-correlation function, and \mathbf{v}_i is the velocity of the atom i at time t .

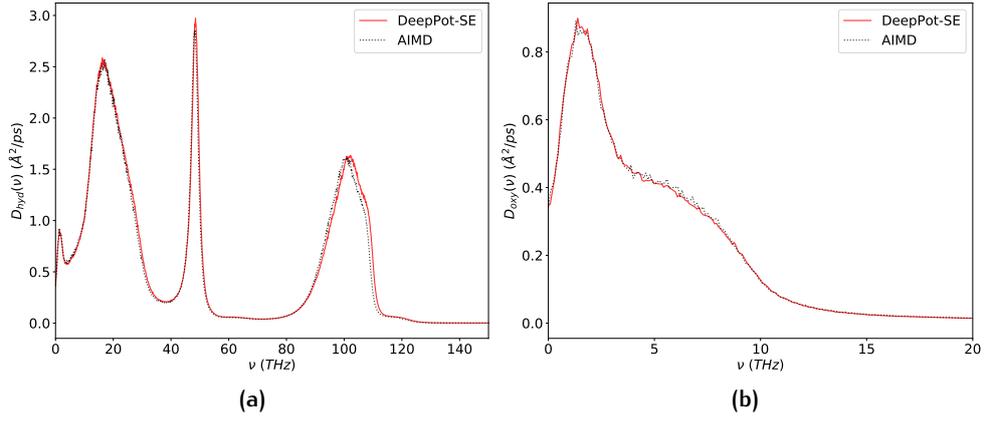


Figure 21: Vibrational spectrum, $D(\nu)$, as defined in Eq. (4.41), of hydrogen atoms (*left*) and oxygen atoms (*right*) as a function of the frequency ν . We compare the results obtained from the ab initio NVE simulation (*black dotted line*), and results obtained from classical simulation with the neural network potential (*red solid line*).

The zero-frequency value of the vibrational spectrum is related to the diffusion coefficient [2001]

$$\begin{aligned} D = D(\nu = 0) &= \frac{1}{3} \int_0^\infty \text{VAC}(t) dt \\ &= \frac{1}{3} \int_0^\infty \frac{1}{N} \sum_{i=1}^N \langle \mathbf{v}_i(t) \mathbf{v}_i(0) \rangle dt, \end{aligned} \quad (4.41)$$

This is nothing but the Green-Kubo integral for the mass transport coefficient, the diffusivity.

In Fig. 21 we display the vibrational spectrum, $D(\nu)$, of hydrogen atoms (Fig. 21a) and oxygen atoms (Fig. 21b) as a function of the frequency ν . The low-frequency region of the graph looks to be well reproduced by the neural network potential, both for the hydrogen and the oxygen atoms. While for the oxygen atoms the plot shows fully compatibility between ab initio and neural network data, for the hydrogen ones, at high frequency, a small mismatch between can be seen in Fig. 21a, in particular on the peak at about 100 THz. The high-frequency region of the graph describes the short-wavelength behaviour of the particle's motion, which seems to be worse approximated by the neural network than the long-wavelength one. Nevertheless, we are still satisfied by the quality of the molecular dynamics reproduced by the DeepPot-SE potential.

In Fig. 22 we compare the low-frequency region of the (Fig. 21) for both the ab initio and neural network simulation. Keeping in mind Eq. (4.41), we see that for both the hydrogen atoms and oxygen atoms the diffusion coefficient (i.e. the zero-frequency value of the Fourier transform of the velocity time-correlation function) of the two approaches is fairly compatible. In particular we obtain:

$$\begin{cases} D_{\text{hyd}}^{\text{NN}} = 0.358 \pm 0.010 \text{ \AA}^2/\text{ps} \\ D_{\text{hyd}}^{\text{AI}} = 0.361 \pm 0.008 \text{ \AA}^2/\text{ps} \end{cases} \quad \begin{cases} D_{\text{oxy}}^{\text{NN}} = 0.342 \pm 0.010 \text{ \AA}^2/\text{ps} \\ D_{\text{oxy}}^{\text{AI}} = 0.346 \pm 0.022 \text{ \AA}^2/\text{ps} \end{cases} \quad (4.42)$$

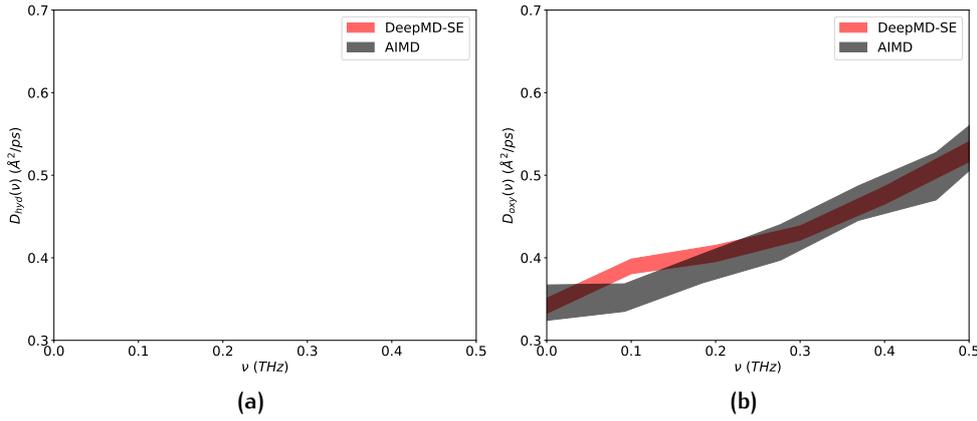


Figure 22: Low-frequency region Fourier transform of the vibrational spectrum, $D(\nu)$, of hydrogen atoms (*left*) and oxygen atoms (*right*). The zero-frequency value correspond to the diffusion coefficient D . In *black* are represented the results obtained from the ab initio NVE simulation, in *red* the results obtained from classical simulation with the neural network potential.

where the apex AI and NN denote respectively the value obtained from AIMD and DeepPot-SE. Moreover also the diffusivity of the two different species of atoms (i.e. hydrogen and oxygen) are compatible as it should be for not dissociated water. The value of the diffusion coefficient of PBE water at about 410 K is somewhere between the two the experimental values: $D^{\text{exp}} = 0.324 \text{ \AA}^2/\text{ps}$ at 313 K and $D^{\text{exp}} = 0.356 \text{ \AA}^2/\text{ps}$ at 318 K [Holz et al., 2000]. This result shows, again, the shift in temperature of about 100 K of PBE water with respect to real water.

After this brief discussion where we analysed standard properties of liquid water comparing the ab initio simulation to the neural network one, we can turn to the shear viscosity. We want to compare the results of the viscosity calculation obtained respectively from AIMD and DPMD. As usual, the simulations provide us the stress time series that were analysed with the cepstral analysis technique in order to obtain the value of the shear viscosity transport coefficient. The cut-off frequency was chosen to be $f^* = 10 \text{ THz}$ for both simulation and the optimal number of cepstral coefficients P^* was found independently using the AIC. The results obtained is presented in Fig. 23; the graph displays the filtered mean periodogram and the cepstrum of both the AIMD and DPMD simulations. The shear viscosity transport coefficient obtained from the neural network trajectory, with a mean temperature of $\langle T \rangle = 415 \pm 0.7 \text{ K}$ is

$$\eta^{\text{NN}} = 0.491 \pm 0.035 \text{ cP} \quad (4.43)$$

fully compatible with the result obtained previously from the AIMD simulation at $410 \pm 0.5 \text{ K}$ given in Eq. (3.35).

In Fig. 24 we display the behaviour of the transport coefficient as a function of the cut-off frequency for both the DeePot-SE simulation and the AIMD one. The results are fairly compatible and the neural network shear viscosity shows a good stability upon f^* .

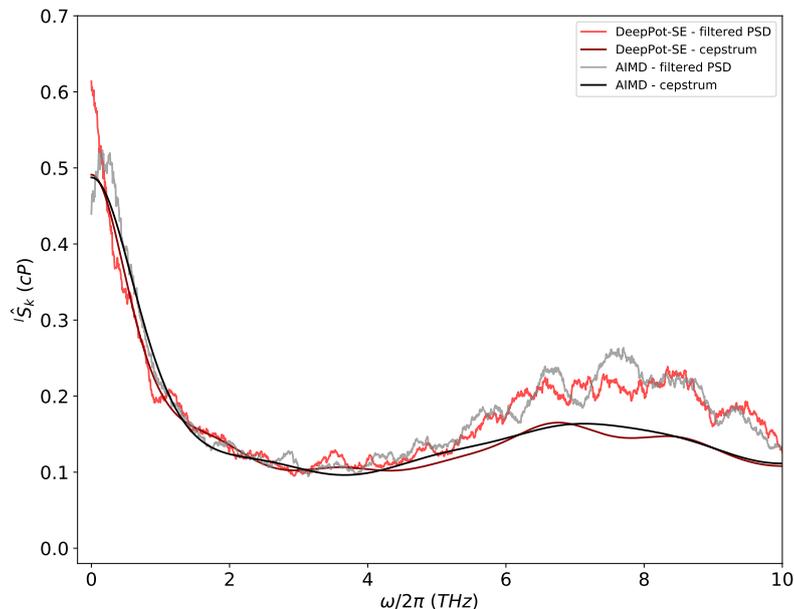


Figure 23: Mean periodogram and cepstrum of the stress time series. The PSD and the cepstrum obtained from ab initio trajectory simulation are denoted respectively in *red* and *dark red*, while the PSD and the cepstrum obtained from the neural network trajectory are denoted in *grey* and *black*.

This ends our discussion in order to validate the DeepMD-SE potential. The inter-atomic forces given by this potential fairly reproduce the static and dynamic properties of the ab initio simulation. The shear viscosity is also well-reproduced and results compatible with the ab initio calculation.

4.5 SHEAR VISCOSITY RESULTS

Finally, we present the results for the calculation of the shear viscosity of water with neural network potential. As we previously mentioned in Chapter 3, one of the problematic features of the shear viscosity of water is its strong dependency upon temperature. Simulations with a more stable temperature can be carried out using a thermostat or increasing the number of particles. The latter choice is the safer one since it does not introduce additional degrees of freedom that can get worse the dynamics of the atoms. The computational cost of AIMD makes the simulation of big system unaffordable. Despite this, machine learning and the DeepPot-SE model provide us alternatives in order to run classical simulations with a neural network potential that reproduce ab initio data, but with a computational cost that is more than ten times lower. This allows us to simulate bigger systems with a classical potential that reproduces quantum forces.

From this prospective, we choose to simulate a water system made of 3000 atoms, 1000 molecules of water, in a cubic super-cell. The size of the super-cell was set in order to recover the same density of the previous simulations: $\rho = 1.0 \text{ g/cm}^3$. We carried out simulations at different temperatures, starting with an NVT of several picoseconds and ending with a 200 ps long NVE that provided the data of the stress time series. The target temperatures of

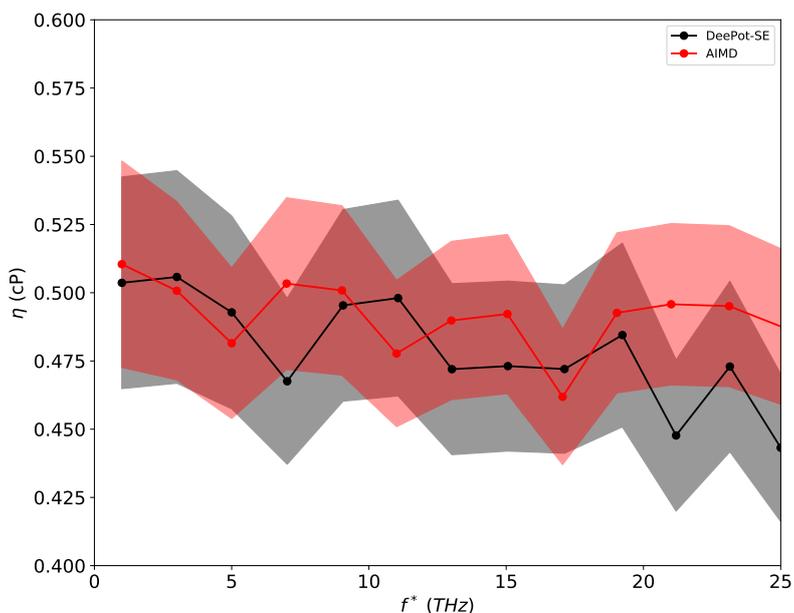


Figure 24: Shear viscosity transport coefficient obtained from the cepstral analysing technique as a function of the cut-off frequency f^* . *Black*: results obtained from the DPMD simulation with their standard deviation. *Red*: results obtained from the AIMD simulation with their standard deviation.

the different NVT of the different simulations were chosen to be in the range 370 – 550 K.

Each different trajectory provides the time stress series that were analysed using the already mentioned protocol. The cut-off for each trajectory’s analysis was chosen ad hoc, but approximately between 5 – 15 THz. Higher temperatures trajectories shows a better behaviour and stability of the shear viscosity as a function of the cut-off frequency, probably due to the less sharpness of the low-frequency peak. The optimal number of cepstral coefficients, P^* , was again chosen independently through the AIC. To give an illustration, in Fig. 25 we report the full mean periodogram of the time series of the stress at 415 ± 0.7 K. The periodogram shows the same principal features as the ab initio one previously analyzed showed in Fig. 12, confirming once again that the neural network potential well reproduce also the dynamical properties of the PBE water.

Finally, in Fig. 26, we present the results for the shear viscosity as a function of the mean temperature of the simulation. The graph confirms the experimental a decreasing trend of the shear viscosity as a function of the temperature. We also report the the ab initio results (in *red*) obtained from the system of 125 molecules which is compatible, but not so evidently, with the other results obtained from the simulations of 1000 molecules.

It is well established that PBE water simulates more structured water, at a temperature that around 100 K lower then the one of the simulation. Since the viscosity is very sensitive upon the temperature of the system, it can be chose as a good indicator in order to estimate of this discrepancy. To give an illustration of the comparison between our results for PBE water and the experimental shear viscosity of water, given in Tab. 2, we report in Fig. 27 all the results together. In particular, in order to match our results with the

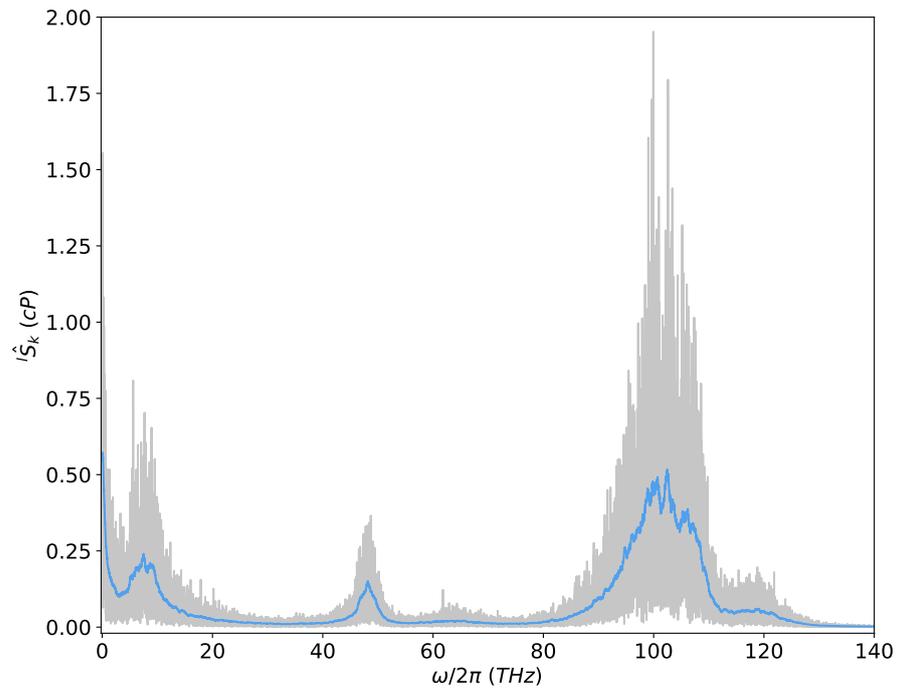


Figure 25: Mean periodogram of stress time series of water obtained from a 200 ps trajectory generated with the neural network potential at 415 ± 0.7 K. *Grey:* mean periodogram obtained directly for Eq. (2.38) with $l = 3$ (*i.e.* with the three equivalent components of the stress tensor). *Blue:* periodogram filtered with a moving average window of width 0.3 THz

experimental ones, we shifted the mean temperature of our simulations of 85 K lower. Our results suggest that in order to achieve a good compatibility between the shear viscosity of PBE water and the shear viscosity of experimental water, the first has to be simulated at a temperature that is about 85 K higher than the temperature of real water.

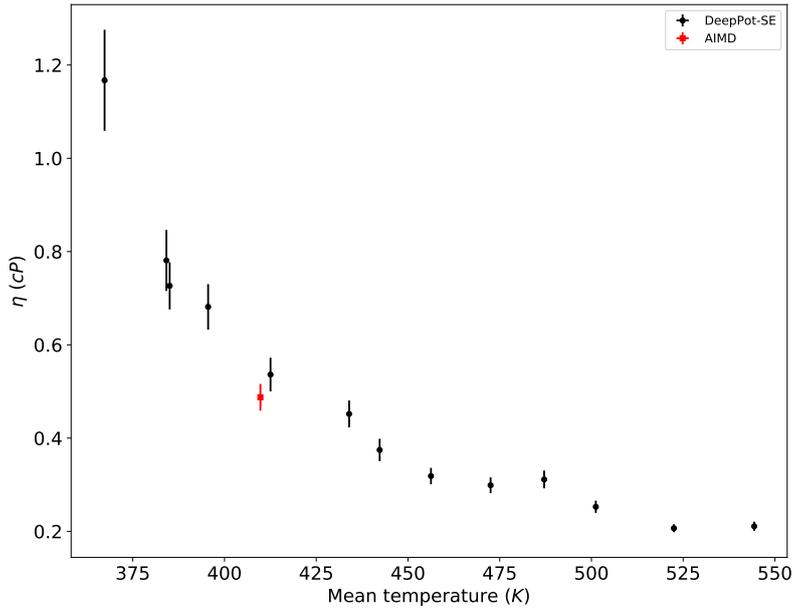


Figure 26: Shear viscosity results computed at different temperatures using the neural network potential trained with ab initio data. The *red* data reports the ab initio value, Eq. (3.35), previously computed.

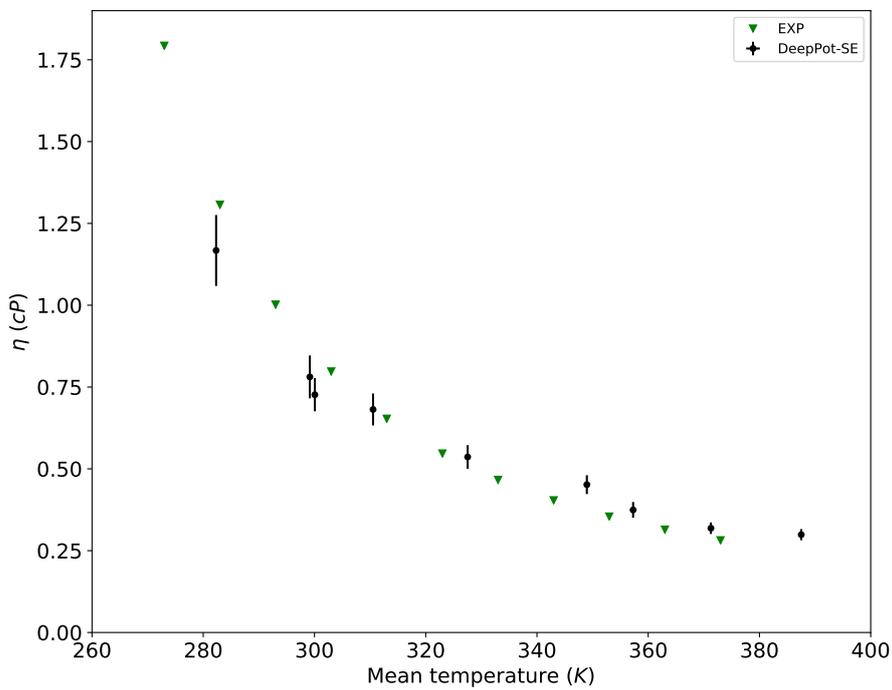


Figure 27: Shear viscosity of water as a function of the temperature. *Black*: results obtained from neural network simulations of a PBE water system made of 1000 molecules; *green*: experimental values of the shear viscosity of water [2005].

5 | CONCLUSIONS

In this study we have applied the Green-Kubo linear response theory to evaluate the shear viscosity transport coefficient of water both from classical simulations, ab initio molecular dynamics simulations and from classical simulation with neural network potentials.

The study proceeded in different steps. The first aim was to validate the cepstral analysis: a data analysis technique used to clean up the integral of the time-correlation function of the conserved currents, in the present case, the off diagonal elements of the stress tensor. The cepstral analysis has been already successfully tested for the thermal transport coefficient [Baroni et al., 2020; Ercole et al., 2017] but no tries have ever been made for the shear viscosity case. With the aim of testing the data analysis technique, we evaluate the shear viscosity of the TIP4P/2005 water model, a classical rigid model, that gives good results in simulating water systems. The potential is an empirical model, fitted over experimental data, and provides a good description of the interactions in liquid water. The validation consisted in comparing the shear viscosity resulting from the cepstral analysis technique and the one resulting from the straight integration of the Green-Kubo formula for the shear viscosity (2.1). The latter approach needs a very long trajectory in order to give a reliable results, unaffordable with AIMD simulations. Size effects were also considered in order to choose the right size of the system in the subsequent ab initio simulation. Also a hypothetical length of the simulation of 150 s were found in order to obtain a relative error on the shear viscosity transport coefficient of about 10 %.

After this first part, were we got used with the data analysis technique, we switched to AIMD simulations. We simulated the PBE water model at a temperature around 400 K which corresponds to about ambient liquid water. We ran a 200 ps long Car-Parrinello trajectory and we evaluate the time-series of the off diagonal elements of the stress tensor. The latter, provides us the fundamental ingredients in order to evaluate the shear viscosity using as always the cepstral analysis. We obtained the following results

$$\eta = 0.488 \pm 0.029 \text{ cP} \quad (5.1)$$

for PBE water at about 410 K. This is the first result of shear viscosity of water obtained from the integration of the Green-Kubo formula.¹

We finally end this study considering the state-of-art technique of modeling atomic interactions using machine learning. The DeePot-SE model was used in order to obtain a neural network potential able to predict interatomic forces with ab initio accuracy. In this sense we trained a deep feed-forward neural network with ab initio data of PBE water and, after a brief

¹ Another calculation of the shear viscosity of water can be found in the literature [Kuhne et al., 2009]. The calculation exploits a relation between the diffusion coefficients value at finite size and its extrapolated value at infinite size, which includes the shear viscosity of the liquid.

validations of the predicted forces and property of the system, we carried out molecular dynamics simulations using this model. We simulate a bigger water system of 1000 molecules at different temperatures in the range 370 – 550 K. From the resulting trajectories, we evaluated, for the first time, the shear viscosity from a neural network simulation. The strong dependency of the shear viscosity upon the temperature and the comparison between neural network results for the shear viscosity and experimental measures, provided us a way in order to estimate the discrepancy between PBE water and real water. We found that PBE water shear viscosity well match the experimental value if the simulation are run at a temperature that is 85 K higher than the temperature of real water.

BIBLIOGRAPHY

- Abascal, J. L. F., & Vega, C. (2005). A general purpose model for the condensed phases of water: Tip4p/2005. *The Journal of Chemical Physics*, 123(23), 234505. <https://doi.org/10.1063/1.2121687>
- Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6), 716–723. <https://doi.org/10.1109/TAC.1974.1100705>
- Allen, M. P., & Tildesley, D. J. (2017). *Computer simulation of liquids*. Oxford university press.
- Baroni, S., Bertossa, R., Ercole, L., Grasselli, F., & Marcolongo, A. (2020). Heat transport in insulators from ab initio green-kubo theory. In *Handbook of materials modeling: Applications: Current and emerging materials* (pp. 809–844). Cham, Springer International Publishing. https://doi.org/10.1007/978-3-319-44680-6_12
- Behler, J., & Parrinello, M. (2007). Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.*, 98, 146401. <https://doi.org/10.1103/PhysRevLett.98.146401>
- Bernasconi, M., Chiarotti, G., Focher, P., Scandolo, S., Tosatti, E., & Parrinello, M. (1995). First-principle-constant pressure molecular dynamics [Proceedings of the 6th International Conference on High Pressure Semiconductor Physics]. *Journal of Physics and Chemistry of Solids*, 56(3), 501–505. [https://doi.org/https://doi.org/10.1016/0022-3697\(94\)00228-2](https://doi.org/https://doi.org/10.1016/0022-3697(94)00228-2)
- Born, M., & Oppenheimer, R. (1927). Zur Quantentheorie der Molekeln. *Annalen der Physik*, 389(20), 457–484. <https://doi.org/10.1002/andp.19273892002>
- Car, R., & Parrinello, M. (1985). Unified approach for molecular dynamics and density-functional theory. *Phys. Rev. Lett.*, 55, 2471–2474. <https://doi.org/10.1103/PhysRevLett.55.2471>
- Childers, D. G., Skinner, D. P., & Kemerait, R. C. (1977). The cepstrum: A guide to processing. *Proceedings of the IEEE*, 65(10), 1428–1443.
- Cook, R. L., De Lucia, F. C., & Helminger, P. (1974). Molecular force field and structure of water: Recent microwave results. *Journal of Molecular Spectroscopy*, 53(1), 62–76. [https://doi.org/10.1016/0022-2852\(74\)90261-6](https://doi.org/10.1016/0022-2852(74)90261-6)
- Dal Corso, A., & Resta, R. (1994). Density-functional theory of macroscopic stress: Gradient-corrected calculations for crystalline se. *Phys. Rev. B*, 50, 4327–4331. <https://doi.org/10.1103/PhysRevB.50.4327>
- de Groot, S. R., & Mazur, P. (1954). Extension of onsager's theory of reciprocal relations. i. *Phys. Rev.*, 94, 218–224. <https://doi.org/10.1103/PhysRev.94.218>
- de Groot, S. R., & Mazur, P. (1962). *Non-equilibrium thermodynamics*. North-Holland Publishing Company.

- DEB, B. M. (1973). The force concept in chemistry. *Rev. Mod. Phys.*, 45, 22–43. <https://doi.org/10.1103/RevModPhys.45.22>
- Deringer, V. L., & Csányi, G. (2017). Machine learning based interatomic potential for amorphous carbon. *Phys. Rev. B*, 95, 094203. <https://doi.org/10.1103/PhysRevB.95.094203>
- Dickey, J. M., & Paskin, A. (1969). Computer simulation of the lattice dynamics of solids. *Phys. Rev.*, 188, 1407–1418. <https://doi.org/10.1103/PhysRev.188.1407>
- Ebden, M. (2015). Gaussian processes: A quick introduction.
- Ercole, L., & Bertossa, R. (2018). Thermocepstrum. a code to estimate transport coefficients from the cepstral analysis of a multi-variate current stationary time series. <https://github.com/lorisercole/thermocepstrum>.
- Ercole, L., Marcolongo, A., & Baroni, S. (2017). Accurate thermal conductivities from optimally short molecular dynamics simulations. *Scientific Reports*, 7(15835). <https://doi.org/10.1038/s41598-017-15843-2>
- Fitch, F. B. (1944). Warren s. mcculloch and walter pitts. a logical calculus of the ideas immanent in nervous activity. bulletin of mathematical biophysics, vol. 5 (1943), pp. 115–133. *Journal of Symbolic Logic*, 9(2), 49–50. <https://doi.org/10.2307/2268029>
- Forster, D. (2018). *Hydrodynamic fluctuations, broken symmetry, and correlation functions*. CRC Press. <https://doi.org/10.1201/9780429493683>
- Frenkel, D., & Smit, B. (2001). *Understanding molecular simulation* (2nd). USA, Academic Press, Inc.
- Giannozzi, P., Baroni, S., Bonini, N., Calandra, M., Car, R., Cavazzoni, C., Ceresoli, D., Chiarotti, G. L., Cococcioni, M., Dabo, I., Corso, A. D., de Gironcoli, S., Fabris, S., Fratesi, G., Gebauer, R., Gerstmann, U., Gougoussis, C., Kokalj, A., Lazzeri, M., . . . Wentzcovitch, R. M. (2009). QUANTUM ESPRESSO: A modular and open-source software project for quantum simulations of materials. *Journal of Physics: Condensed Matter*, 21(39), 395502. <https://doi.org/10.1088/0953-8984/21/39/395502>
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. The MIT Press.
- Green, M. S. (1952). Markoff random processes and the statistical mechanics of time-dependent phenomena. *The Journal of Chemical Physics*, 20(8), <https://doi.org/10.1063/1.1700722>, 1281–1295. <https://doi.org/10.1063/1.1700722>
- Grossman, J. C., Schwegler, E., Draeger, E. W., Gygi, F., & Galli, G. (2004). Towards an assessment of the accuracy of density functional theory for first principles simulations of water. *The Journal of Chemical Physics*, 120(1), <https://doi.org/10.1063/1.1630560>, 300–311. <https://doi.org/10.1063/1.1630560>
- Hamann, D. R., Schlüter, M., & Chiang, C. (1979). Norm-conserving pseudopotentials. *Phys. Rev. Lett.*, 43, 1494–1497. <https://doi.org/10.1103/PhysRevLett.43.1494>
- Haynes, W. M., Lide, D. R., & Bruno, T. J. (2005). *Crc handbook of chemistry and physics: A ready-reference book of chemical and physical data*. Florida: CRC Press.

- Hohenberg, P., & Kohn, W. (1964). Inhomogeneous electron gas. *Phys. Rev.*, *136*, B864–B871. <https://doi.org/10.1103/PhysRev.136.B864>
- Holz, M., Heil, S., & Sacco, A. (2000). Temperature-dependent self-diffusion coefficients of water and six selected molecular liquids for calibration in accurate 1h nmr pfg measurements. *Physical Chemistry Chemical Physics*, *2*. <https://doi.org/10.1039/B005319H>
- Khintchine, A. (1934). Korrelationstheorie der stationären stochastischen prozesse. *Mathematische Annalen*, *109*, 604–615. <http://eudml.org/doc/159698>
- Kohn, W., & Sham, L. J. (1965). Self-consistent equations including exchange and correlation effects. *Phys. Rev.*, *140*, A1133–A1138. <https://doi.org/10.1103/PhysRev.140.A1133>
- Kubo, R. (1957). Statistical-mechanical theory of irreversible processes. i. general theory and simple applications to magnetic and conduction problems. *Journal of the Physical Society of Japan*, *12*(6), 570–586. <https://doi.org/10.1143/JPSJ.12.570>
- Kubo, R., Toda, M., & Hashitsume, N. (1991). *Statistical physics II*. Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-58244-8>
- Kubo, R., Yokota, M., & Nakajima, S. (1957). Statistical-mechanical theory of irreversible processes. ii. response to thermal disturbance. *Journal of the Physical Society of Japan*, *12*(11), 1203–1211. <https://doi.org/10.1143/JPSJ.12.1203>
- Kuhne, T. D., Krack, M., & Parrinello, M. (2009). Static and dynamical properties of liquid water from first principles by a novel car-parrinello-like approach. *Journal of Chemical Theory and Computation*, *5*, 235–241. <https://doi.org/10.1021/ct800417q>
- Martin, R. M. (2004). Quantum molecular dynamics (qmd). In *Electronic structure: Basic theory and practical methods* (pp. 371–386). Cambridge University Press. <https://doi.org/10.1017/CBO9780511805769.022>
- Nielsen, O. H., & Martin, R. M. (1985a). Quantum-mechanical theory of stress and force. *Phys. Rev. B*, *32*, 3780–3791. <https://doi.org/10.1103/PhysRevB.32.3780>
- Nielsen, O. H., & Martin, R. M. (1985b). Stresses in semiconductors: Ab initio calculations on si, ge, and gaas. *Phys. Rev. B*, *32*, 3792–3805. <https://doi.org/10.1103/PhysRevB.32.3792>
- Onsager, L. (1931a). Reciprocal relations in irreversible processes. i. *Phys. Rev.*, *37*, 405–426. <https://doi.org/10.1103/PhysRev.37.405>
- Onsager, L. (1931b). Reciprocal relations in irreversible processes. ii. *Phys. Rev.*, *38*, 2265–2279. <https://doi.org/10.1103/PhysRev.38.2265>
- Ortiz de Zárate, J. M., & Sengers, J. V. (2006). Chapter 2 - nonequilibrium thermodynamics (J. M. Ortiz de Zárate & J. V. Sengers, Eds.). In J. M. Ortiz de Zárate & J. V. Sengers (Eds.), *Hydrodynamic fluctuations in fluids and fluid mixtures*. Amsterdam, Elsevier. <https://doi.org/https://doi.org/10.1016/B978-044451515-5/50002-8>
- Papoulis, A. (1991). *Probability, random variables and stochastic processes* (3rd). McGraw-Hill Companies.
- Parrinello, M., & Rahman, A. (1980). Crystal structure and pair potentials: A molecular-dynamics study. *Phys. Rev. Lett.*, *45*, 1196–1199. <https://doi.org/10.1103/PhysRevLett.45.1196>

- Peligrad, M., & Wu, W. B. (2010). Central limit theorem for fourier transforms of stationary processes. *Ann. Probab.*, 38(5), 2009–2022. <https://doi.org/10.1214/10-AOP530>
- Perdew, J. P., Burke, K., & Ernzerhof, M. (1996). Generalized gradient approximation made simple. *Phys. Rev. Lett.*, 77, 3865–3868. <https://doi.org/10.1103/PhysRevLett.77.3865>
- Plimpton, S. (1995). Fast parallel algorithms for short-range molecular dynamics. *Journal of Computational Physics*, 117(1), 1–19. <https://doi.org/https://doi.org/10.1006/jcph.1995.1039>
- Sit, P. H.-L., & Marzari, N. (2005). Static and dynamical properties of heavy water at ambient conditions from first-principles molecular dynamics. *The Journal of Chemical Physics*, 122(20), 204510. <https://doi.org/10.1063/1.1908913>
- Skinner, L. B., Huang, C., Schlesinger, D., Pettersson, L. G. M., Nilsson, A., & Benmore, C. J. (2013). Benchmark oxygen-oxygen pair-distribution function of ambient water from x-ray diffraction measurements with a wide q-range. *The Journal of Chemical Physics*, 138(7), 074506. <https://doi.org/10.1063/1.4790861>
- Smith, J. S., Isayev, O., & Roitberg, A. E. (2017). Ani-1: An extensible neural network potential with dft accuracy at force field computational cost. *Chem. Sci.*, 8, 3192–3203. <https://doi.org/10.1039/C6SC05720A>
- Thompson, A. P., Plimpton, S. J., & Mattson, W. (2009). General formulation of pressure and stress tensor for arbitrary many-body interaction potentials under periodic boundary conditions. *The Journal of Chemical Physics*, 131(15), <https://doi.org/10.1063/1.3245303>, 154107. <https://doi.org/10.1063/1.3245303>
- Vanderbilt, D. (1985). Optimally smooth norm-conserving pseudopotentials. *Phys. Rev. B*, 32, 8412–8415. <https://doi.org/10.1103/PhysRevB.32.8412>
- Wiener, N. (1930). Generalized harmonic analysis. *Acta Math.*, 55, 117–258. <https://doi.org/10.1007/BF02546511>
- Yeh, I.-C., & Hummer, G. (2004). System-size dependence of diffusion coefficients and viscosities from molecular dynamics simulations with periodic boundary conditions. *The Journal of Physical Chemistry B*, 108(40), 15873–15879. <https://doi.org/10.1021/jp0477147>
- Zhang, L., Han, J., Wang, H., Car, R., & E, W. (2018). Deep potential molecular dynamics: A scalable model with the accuracy of quantum mechanics. *Phys. Rev. Lett.*, 120, 143001. <https://doi.org/10.1103/PhysRevLett.120.143001>
- Zhang, L., Han, J., Wang, H., Saidi, W. A., Car, R., & Weinan, E. (2018). End-to-end symmetry preserving inter-atomic potential energy model for finite and extended systems, Montréal, Canada, Curran Associates Inc.