THESIS FOR THE DEGREE OF LICENTIATE OF ENGINEERING

Shedding light on liquid chromophores using machine learning

Eric Lindgren

Department of Physics
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2024

Shedding light on liquid chromophores using machine learning
ERIC LINDGREN

Department of Physics
Chalmers University of Technology
SE-412 96 Göteborg, Sweden
Telephone +46 (0)31 772 10 00

Cover: The CALORINE salamander, a perylene molecule and a neural network resting on an island in a sea of liquid chromophores.

Chalmers digitaltryck
Göteborg, Sweden 2024

Shedding light on liquid chromophores using machine learning

Eric Lindgren
Department of Physics
Chalmers University of Technology

Abstract

Chromophores are a class of molecules with widespread use in nature. Chlorophyll in plants contain chromophores making photosynthesis possible and the retinal molecules in our eyes have chromophores making the world around us visible. Chromophores are also fundamental for developing a wide range of technologies crucial for a transition to a sustainable society, including organic electronics, solvent-free dyes and systems for storing solar energy in the form of heat. While chromophores have been widely studied experimentally, we still lack a sufficient understanding of their structure and dynamics on the atomic scale. This thesis outlines a simulation framework that links electronic structure calculations via molecular dynamics simulations to experiments, with a specific focus on neutron scattering. The key ingredient of this work are machine-learned force fields, allowing simulations with the accuracy of quantum mechanical calculations for large systems of chromophores, bridging the gap between theoretical simulations and experimental findings.

Keywords: chromophores, machine learning, machine learned force fields, molecular dynamics, neutron scattering

LIST OF APPENDED PAPERS

This thesis is based on work presented in the following papers:

I **Structural stability and dynamics of liquid chromophore aggregates**
Eric Lindgren, Jakub Fojt, Jan Swenson, Christian Müller, and Paul Erhart
In manuscript

II **GPUMD: A package for constructing accurate machine-learned potentials and performing highly efficient atomistic simulations**
Zheyong Fan, Yanzhou Wang, Penghua Ying, Keke Song, Junjie Wang, Yong Wang, Zezhu Zeng, Ke Xu, **Eric Lindgren**, J. Magnus Rahm, Alexander J. Gabourie, Jiahui Liu, Haikuan Dong, Jianyang Wu, Yue Chen, Zheng Zhong, Jian Sun, Paul Erhart, Yanjing Su and Tapio Ala-Nissila
The Journal of Chemical Physics, 157, 114801 (2022)

III **calorine: A Python package for constructing and sampling neuroevolution potential models**
Eric Lindgren, Magnus Rahm, Erik Fransson, Fredrik Eriksson, Nicklas Österbacka, Zheyong Fan, and Paul Erhart
The Journal of Open Source Software, 9(95), 6264 (2024)

PUBLICATIONS NOT INCLUDED IN THIS THESIS

The following publications are outside the scope of this thesis:

Machine Learning for Polaritonic Chemistry: Accessing Chemical Kinetics
Christian Schäfer, Jakub Fojt, **Eric Lindgren**, and Paul Erhart
J. Am. Chem. Soc, 146, 8, 5402–5413, (2024)

General-purpose machine-learned potential for 16 elemental metals and their alloys
Zheyong Fan, Keke Song, Rui Zhao, Jiahui Liu, Yanzhou Wang, **Eric Lindgren**, Yong Wang, Shunda Chen, Ke Xu, Ting Liang, Penghua Ying, Nan Xu, Zhiqiang Zhao, Jiuyang Shi, Junjie Wang, Shuang Lyu, Zezhu Zeng, Shirong Liang, Haikuan Dong, Ligang Sun, Yue Chen, Zhuhua Zhangm, Wanlin Guo, Ping Qian, Jian Sun, Paul Erhart, Tapio Ala-Nissilä and Yanjing Su
arXiv, preprint, submitted for review, (2023)

Tensorial properties via the neuroevolution potential framework: Fast simulation of infrared and Raman spectra
Nan Xu, Petter Rosander, Christian Schäfer, **Eric Lindgren**, Nicklas Österbacka, Mandi Fang, Wei Chen, Yi He, Zheyong Fan and Paul Erhart
arXiv, preprint, submitted for review, (2023)

The author's contribution to the papers included in the thesis:

I I designed and performed the molecular dynamics simulations, calculated the correlation functions, and analyzed and interpreted the results. I also visualized the results and wrote the paper.

II In this paper we present the GPUMD package, to which I am a developer. My main contribution to this paper was in being part of the development of the initial version of the CALORINE package, a companion software presented together with GPUMD. Additionally, I contributed usage examples for CALORINE to the paper.

III Here we present CALORINE in a standalone publication. I am the main developer and maintainer of CALORINE, and wrote the paper.

Contents

List of abbreviations

CPU central processing unit. 39

DFT density-functional theory. 4, 5, 25–29, 39, 40, 46

FF force field. 4, 25–29, 41, 45, 46

GP gaussian process. 29, 30

GPU graphics processing unit. 35, 39, 40, 42, 46

KS Kohn-Sham. 26, 27

MD molecular dynamics. 3–5, 7, 8, 11, 12, 15–29, 32, 38–43, 45–47

ML machine learning. 4, 31

ML-FF machine-learned force field. 4, 5, 25, 28–32, 40, 46

NEP neuroevolution potential. 5, 28–30, 32, 34–36, 38–40, 42, 43, 45–47

NN neural network. 30, 32, 34, 37

PBC periodic boundary conditions. 16, 17, 46

PCA principal component analysis. 33, 34

RMSE root mean squared error. 35, 38, 39

SNES separable natural evolution strategy. 36

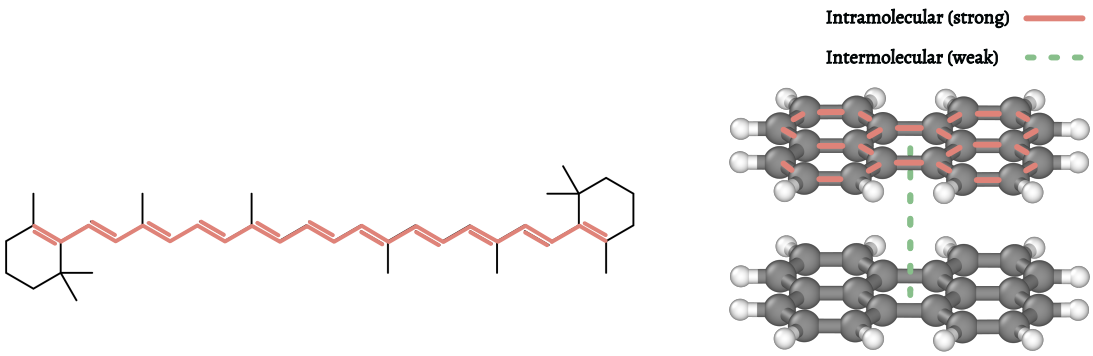
1

Introduction

Datorer är coolt

Nicklas, 2022

There is something mesmerizing about colors. Their alluring appeal has tantalized mankind throughout history, up to and including the modern materials scientist. This is not without reason. A colorful substance is not only pretty to look at, but potentially promises interesting optical properties. One class of organic molecules that live up to this promise are chromophores. Translated from ancient Greek, "chromophore" means "color-bearer", which reflects their role as the parts of molecules that are responsible for giving them color. Two famous examples of chromophores that can be found in nature include chlorophyll, which enables photosynthesis in plants, and β-carotene which makes autumn leaves, carrots and pumpkins appear orange. In recent years, chromophores have been studied as candidates for many applications that are important for a transition to a more sustainable society, including solar cells [1–3], dyes [4–6], organic light-emitting diodes [7–9], organic semiconductors [10–12] and solar thermal storage systems [13–18]. Common to all of these applications is the desire to control the optical properties of the chromophore systems. In this thesis, we will dive head-first into how computer simulations can shed light on the structure and dynamics of chromophores, which may hold the key for controlling their optical properties.



(a) The conjugated system of β-carotene

(b) Inter- and intramolecular bonds

Figure 1.1: Schematics of conjugated systems and bonds in and between molecules. **a)** The conjugated system of a β-carotene molecule formed by overlapping π-orbitals, marked in red. Electrons are free to delocalize along the length of the conjugated system. **b)** A sketch of strong intermolecular bonds, typically covalent bonds, and weak intramolecular bonds between two perylene monomers.

1.1 Controlling the optical properties of chromophores

Chromophores owe their colorfulness to them containing conjugated systems. A conjugated system is a long chain of atoms over which electrons are free to move (Fig. 1.1a). Specifically, the p-orbitals of the carbon atoms overlap forming π-bonds, which allows electrons to delocalize and move along the length of the conjugated carbon chain [19, Chapter 7]. The size (or extent) of the conjugated system affects which energy, and thus wavelength, of incoming light is required for an electron-hole pair to become excited, causing light of that wavelength to be absorbed. A helpful picture could be that of an antenna, where a longer chain would lead to longer wavelengths to be absorbed. The wavelengths that are not absorbed get reflected, making the chromophore appear colored.

In general, additional properties other than the length of the conjugated carbon chain affect the optical properties of systems of chromophores. The property most relevant for the types of chromophores studied in this thesis is their tendency to assemble into larger groupings of molecules [20–22]. These are known as supramolecular aggregates, as they are formed through interaction *between* molecules, so-called intermolecular bonds (Fig. 1.1b). Intermolecular bonds are typically relatively weak compared to the covalent bonds within the individual molecules (intramolecular bonds), being mediated by electrostatic Coulomb interaction or van-der-Waals forces [23]. Supramolecular aggregates impact the optical properties in two ways. First, the size of the resulting supramolecular

aggregates impact how light scatters in the sample [24]. Second, the conjugated systems of neighboring molecules can interact with each other, leading to distinct changes in the optical spectra of the aggregate system compared to the individual molecules [25]. Controlling the tendency of chromophores to self-assemble into supramolecular aggregates allow for control of their optical properties, which would be useful in a broad range of applications.

There are two primary approaches for controlling the self-assembly in chromophores. The **first** is side-group engineering; by appending side-chains of different length onto a core conjugated structure, the formation of supramolecular aggregates can be controlled [26–29]. The appended side-chains can also have the side-effect of modifying the conjugated structure of the chromophore, which changes the optical response of the individual monomers [30]. Examples of optical properties which can be impacted through sidegroup engineering include luminescence [5], and the efficiency of triplet-triplet annihilation up-conversion [31], a process that can increase the efficiency of solar cells. The **second** approach for controlling the structure of chromophore aggregates is mixing different types of chromophores [30, 32], which has the added benefit of reducing the need of solvating the chromophores in a potentially toxic solvent. Mixing have, for instance, been used to increase the stability of chromophore photovoltaics [33], and for increasing the efficiency of solar cells [34].

One specific application of mixtures of chromophores that is of particular interest in this thesis is that of glass forming systems. Recently, Hultmark and colleagues found mixtures of perylene derivatives, a type of chromophore, to be ultra-strong glass formers [35]. The authors attribute the strong glass-forming behavior of the perylene mixtures to a transition between two liquid phases. However, the detailed structure and dynamics of these liquid phases are not understood on the atomistic level. Atomic understanding of the structural and dynamic processes in mixtures of chromophores is vital to further develop mixtures as a handle for controlling aggregate structure, and to optimize their performance in applications.

1.2 Characterizing structure and dynamics with molecular dynamics and machine learning

One tool that can help shed light on the structure and dynamics in mixtures of liquid chromophores are computer simulations. In particular, as we shall see in Chapter 3, molecular dynamics (MD) simulations are well suited for this task, since each atom is explicitly considered in the simulation. Using MD to study chromophores is not new; MD simulations have been used to study everything from the self-assembly of chromophores on substrates [36, 37], via the structure and dynamics of supramolecular systems [38, 39], to optical detection of proteins linked to Alzheimer's disease [40]. How-

ever, these simulations are often performed in conjunction with experiments, and moreover typically consider relatively small or idealized systems. Such simulations are suited for elucidating experimental results but do not necessarily paint a broader theoretical picture.

We need to add a few more ingredients to the MD simulations if we are to reach the lofty goal of a simulation protocol that can capture the structure and dynamics of chromophore systems. MD simulations relies on accurately calculating the forces between pairs of atoms. Conventionally, these forces are obtained from a force field (FF), which are functions fitted to a broad range of materials. More accurate forces can be obtained from other means, for instance from electronic-structure calculations such as density-functional theory (DFT), but the computational cost of these typically scales strongly with the number of electrons in the system, which limits their use to small systems, typically involving hundreds or thousands of atoms. One potential solution to this trade-off between accuracy and computational feasibility that has emerged during the last couple of years are machine-learned force field (ML-FF). As we shall see in Chapter 4, by training a machine learning (ML) model to predict forces calculated with DFT, one can obtain a ML-FF that provides near DFT accuracy at the speed of a conventional FF.

Once one has performed an accurate MD simulation, the results can be connected to the observables from various experimental techniques. One such technique that is well suited studying the structure and dynamics of organic systems, including chromophores, is neutron scattering, which we will learn more about in Chapter 2 together with the connection to MD simulations. In the context of this thesis, one can thus see ML as a bridge for connecting simulations and experiments, by enabling highly accurate MD simulations that can be used to predict neutron scattering experiments, which in turn can improve our understanding of the structure and dynamics of liquid chromophores.

1.3 Research questions

In this thesis, I aim to develop a simulation framework combining molecular dynamics (MD) and machine-learned force fields (ML-FFs) for studying the structure and dynamics of chromophores, and connecting these results to experimental observables, bridging the gap between simulations and experiments. I will in particular study a class of chromophores, namely perylene ($C_{20}H_{12}$) and derivatives thereof, as a prototypical system, but the approach should be readily extendable to other systems. I will consider two research questions:

- To what extent can the developed simulation protocol capture the structure and dynamics of aggregates of perylene derivatives?

- How well can neutron scattering experiments be predicted using the simulation protocol?

1.4 Structure of the thesis

This thesis consists of three main chapters, a summary of the included papers, and a summary and outlook. The first chapter concerns neutron scattering experiments, with an emphasis on the theory and the quantities that are observable in computer simulations, such as the dynamic structure factor. This is followed by the second chapter, focusing on MD simulations and how they can be used to predict neutron scattering experiments. The third chapter details the theory of ML-FFs with a specific focus on the neuroevolution potential (NEP) framework, how such models are constructed and how they can be used to run MD simulations with the accuracy of an electronic structure method such as DFT. This is followed by a short summary of the three papers that are the foundation of this thesis, after which the thesis is concluded by a summary and outlook.

2

Neutron Scattering

I'm afraid neutrons will not be of any use to any one.

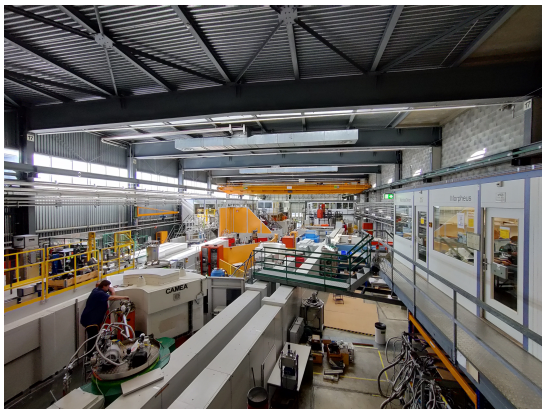
Sir James Chadwick,
discoverer of the neutron

Neutron scattering is one of the most important experimental techniques for studying the structure and dynamics of chromophores, as neutrons are excellent probes for organic matter. Producing neutrons requires large facilities, such as nuclear reactors or spallation sources, and, hence, these are only available in a handful of locations globally. See Fig. 2.1 for three of these locations; the TU Delft experimental reactor in the Netherlands, the PSI/SINQ site in Switzerland, and the European Spallation Source (ESS) currently under construction in Lund, Sweden.

In neutron scattering, the neutron scatters off of the atomic nucleus. The amount with which each atomic isotope scatters neutrons varies greatly across the periodic table, which increases the contrast between different atomic species. Hydrogen, in particular, scatters neutrons strongly, which makes neutrons well-suited for studying hydrogen-rich organic matter, such as chromophores. Neutrons additionally transfer relatively small amounts of kinetic energy to the sample under study, typically in the meV range, which is comparable with the energy scales associated with molecular motion such as rotations. However, like most experimental techniques, neutron scattering measurements needs to be paired with simulations in order to explain and elucidate the results. As we shall see at the end of this chapter, the connection between neutron scattering and MD simulations is straightforward, which further makes neutron scattering particularly interesting in the context of this thesis.

In this chapter we will follow Squires [41, Chapters 1, 2 and 4] in deriving some key results in neutron scattering theory, starting from the double differential cross section,

(a) TU Delft



(b) SINQ

(c) ESS (under construction)

Figure 2.1: Three of the handful of neutron scattering facilities globally. From left to right: the TU Delft experimental reactor in the Netherlands, the SINQ beamline in Switzerland, and the ESS beamline in Lund, Sweden.

namely the expressions for coherent and incoherent scattering, the dynamic structure factor, and how the dynamic structure factor can be related to observables in MD simulations.

2.1 Basic scattering theory of neutrons

The quantity measured in neutron scattering is the *scattered intensity* from each measured neutron, $I(\boldsymbol{q}, \omega)$, where $\boldsymbol{q}$ is the momentum transfer to the sample, and ω is the angular frequency of the excitation gained by the sample. Conceptually, this can be reformulated as $I(\theta, \partial E')$, where θ is the change in angle of the incoming neutron wave and ∂E is the energy transferred to the sample from the neutron, which leads to the interpretation of $I(\theta, \partial E')$ as the measured intensity of neutrons that change direction by

an angle θ and transfer energy $\partial E'$ to the sample. If $\partial E' = 0$, the process is called *elastic neutron scattering*, and if $\partial E' \neq 0$ it is called *inelastic neutron scattering*. Broadly speaking, neutron scattering techniques that utilize elastic scattering are used to study the structure of materials, whilst inelastic techniques are used to study their dynamics.

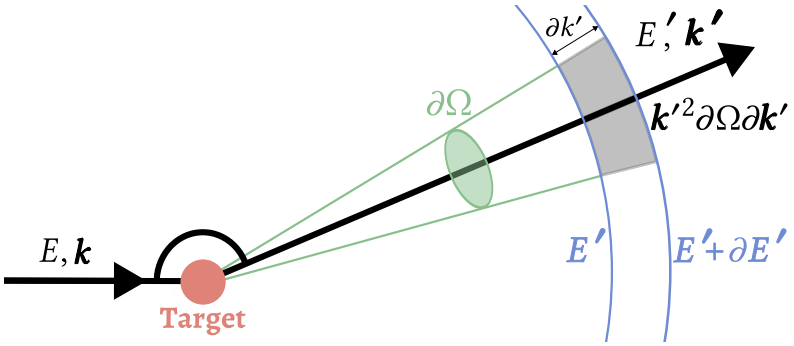


Figure 2.2: Scattering of a neutron with incoming wave vector $\boldsymbol{k}$ and energy E into solid angle $\partial\Omega$ when hitting a target, with the final neutron having wave vector $\boldsymbol{k}'$ and energy between E' and $E' + \partial E'$.

Formally, the intensity measured in a neutron scattering experiment is directly related to the *partial differential cross section*,

$$\frac{\partial^2\sigma}{\partial\Omega\partial E'} = \frac{1}{\Phi}\frac{\text{No. neutrons scattered per time into solid angle } \partial\Omega \text{ with final energy between } E' \text{ and } E' + \partial E'}{\partial\Omega\partial E'} \tag{2.1}$$

with Φ being the flux of the incoming neutron beam, i.e., the number of neutrons per unit area and second, and σ denoting the cross section. Now, let the incoming neutron have wave vector $\boldsymbol{k}$ and energy E, and wave vector $\boldsymbol{k}'$ and energy E' after the scattering event. Furthermore, denote the initial state of the scattering system λ, and the final state as λ'. This is the situation described in Fig. 2.2. With these definitions one can, after a relatively lengthy derivation, which we skip here in the interest of time and space, express Eq. 2.1 explicitly,

$$\frac{\partial^2\sigma}{\partial\Omega\partial E'} = \frac{|k'|}{|k|}\frac{1}{2\pi\hbar}\sum_{jj'} b_j b_{j'} \int_{-\infty}^{\infty} \left\langle e^{-i\boldsymbol{q}\cdot\boldsymbol{R}_{j'}(0)} e^{i\boldsymbol{q}\cdot\boldsymbol{R}_j(t)} \right\rangle e^{-i\omega t} dt \tag{2.2}$$

where the double sum runs over all pairs of scattering nuclei with position $\boldsymbol{R}_j$ and *scattering length* b_j. Note that $\boldsymbol{R}_j(t)$ is the time-dependent position operator in the Heisenberg picture, defined as

$$\boldsymbol{R}_j(t) = \exp(iHt/\hbar)\, \boldsymbol{R}_j\, \exp(-iHt/\hbar), \quad \boldsymbol{R}_j(0) = \boldsymbol{R}_j. \tag{2.3}$$

$\boldsymbol{q} = \boldsymbol{k} - \boldsymbol{k'}$ is the change in momentum for the scattered neutron, and $\hbar\omega = E - E'$ is the change in the kinetic energy of the neutron. $\langle ... \rangle$ denotes a thermal average over all initial system states λ, which are assumed to follow a Boltzmann distribution. That is, the thermal average of an operator A for a system with Hamiltonian H is defined as

$$\langle A \rangle = \sum_\lambda p_\lambda \langle \lambda | A | \lambda \rangle, \quad p_\lambda = \frac{1}{Z} e^{-\beta E_\lambda}, \quad Z = \sum_\lambda e^{-\beta E_\lambda}, \quad \beta = \frac{1}{k_B T}. \tag{2.4}$$

I would like to highlight two key steps in the derivation of Eq. 2.2. First, the starting point for the derivation is to express the number of neutrons scattered into solid angle $\partial\Omega$ per unit time in terms of the transition rate of the neutron and sample system transition from state $|\boldsymbol{k}, \lambda\rangle$ to $|\boldsymbol{k'}, \lambda'\rangle$, where $\boldsymbol{k'}$ is in $\partial\Omega$. This transition rate can be expressed using Fermi's golden rule,

$$\Gamma_{\lambda \to \lambda'} = \frac{2\pi}{\hbar} \rho_{\boldsymbol{k'}} \left| \langle \boldsymbol{k}, \lambda | V | \boldsymbol{k'}, \lambda' \rangle \right|^2. \tag{2.5}$$

Here, $\rho_{\boldsymbol{k'}}$ is the density of neutrons with momentum $\boldsymbol{k'}$ in $\partial\Omega$. V is the scattering potential, which leads us to the next step I would like to highlight. A common choice of V, which has been used in deriving Eq. 2.2, is the *Fermi pseudopotential*,

$$V(\boldsymbol{r}) = \frac{2\pi\hbar^2}{m} b \delta(\boldsymbol{r}), \tag{2.6}$$

here expressed for a single nucleus with mass m centered at the origin. b is the scattering length, and can be interpreted as the strength of the scattering potential; a larger value of b means a more repulsive potential. The scattering length is different for different isotopes and is in general a complex number, with the imaginary part representing neutron absorption. For most nuclei the imaginary part is small, and hence b has been assumed to be real in the derivation of Eq. 2.2.

2.2 Coherent and incoherent neutron scattering

We split the sum over j, j' in Eq. 2.2 into two terms, corresponding to $j \neq j'$ and $j = j'$,

$$\begin{aligned} \frac{\partial^2\sigma}{\partial\Omega \partial E'} = \frac{|k'|}{|k|} \frac{1}{2\pi\hbar} \Bigg(& \sum_{j \neq j'} b_j b_{j'} \int_{-\infty}^{\infty} \left\langle e^{-i\boldsymbol{q}\cdot\boldsymbol{R}_{j'}(0)} e^{i\boldsymbol{q}\cdot\boldsymbol{R}_j(t)} \right\rangle e^{-i\omega t} dt \\ & + \sum_j b_j^2 \int_{-\infty}^{\infty} \left\langle e^{-i\boldsymbol{q}\cdot\boldsymbol{R}_j(0)} e^{i\boldsymbol{q}\cdot\boldsymbol{R}_j(t)} \right\rangle e^{-i\omega t} dt \Bigg). \end{aligned} \tag{2.7}$$

Each nuclei in the sample can possibly have a different scattering length b_j that occurs with abundance f_j. However, a macroscopic sample consists of a large number of nuclei,

and thus we can replace the factors b_j^2 and $b_j b_{j'}$ in Eq. 2.7 with their averages, $\overline{b_j^2}$ and $\overline{b_j b_{j'}}$. The average scattering lengths are

$$\overline{b_j^2} = \sum_j f_j b_j^2 \equiv \overline{b^2} \quad \text{and} \quad \overline{b_j b_{j'}} = \sum_j f_j b_j \sum_{j'} f_{j'} b_{j'} \equiv \bar{b}^2. \tag{2.8}$$

By adding and subtracting the missing term for $j = j'$ in the first term in Eq. 2.7 we arrive at

$$\begin{aligned}\frac{\partial^2\sigma}{\partial\Omega\partial E'} =& \frac{|k'|}{|k|}\frac{1}{2\pi\hbar}\bar{b}^2 \sum_{jj'}\int_{-\infty}^{\infty}\left\langle e^{-i\boldsymbol{q}\cdot\boldsymbol{R}_{j'}(0)} e^{i\boldsymbol{q}\cdot\boldsymbol{R}_j(t)}\right\rangle e^{-i\omega t}dt \\ &+\frac{|k'|}{|k|}\frac{1}{2\pi\hbar}\left(\overline{b^2}-\bar{b}^2\right)\sum_j\int_{-\infty}^{\infty}\left\langle e^{-i\boldsymbol{q}\cdot\boldsymbol{R}_j(0)} e^{i\boldsymbol{q}\cdot\boldsymbol{R}_j(t)}\right\rangle e^{-i\omega t}dt.\end{aligned} \tag{2.9}$$

By introducing $\sigma_{\text{coh}} = 4\pi\bar{b}^2$ and $\sigma_{\text{inc}} = 4\pi\left(\overline{b^2}-\bar{b}^2\right)$ we identify the two terms

$$\begin{aligned}\left(\frac{\partial^2\sigma}{\partial\Omega\partial E'}\right)_{\text{coh}} =& \frac{\sigma_{\text{coh}}}{4\pi}\frac{|k'|}{|k|}\frac{1}{2\pi\hbar}\bar{b}^2 \sum_{jj'}\int_{-\infty}^{\infty}\left\langle e^{-i\boldsymbol{q}\cdot\boldsymbol{R}_{j'}(0)} e^{i\boldsymbol{q}\cdot\boldsymbol{R}_j(t)}\right\rangle e^{-i\omega t}dt \\ \left(\frac{\partial^2\sigma}{\partial\Omega\partial E'}\right)_{\text{inc}} =& \frac{\sigma_{\text{inc}}}{4\pi}\frac{|k'|}{|k|}\frac{1}{2\pi\hbar}\left(\overline{b^2}-\bar{b}^2\right)\sum_j\int_{-\infty}^{\infty}\left\langle e^{-i\boldsymbol{q}\cdot\boldsymbol{R}_j(0)} e^{i\boldsymbol{q}\cdot\boldsymbol{R}_j(t)}\right\rangle e^{-i\omega t}dt,\end{aligned} \tag{2.10}$$

as the *coherent* and the *incoherent* partial neutron cross sections, respectively. The sum in the incoherent cross section runs over each atom j, and, thus, the incoherent scattering contains the scattered intensity from the *individual nuclei*. The coherent cross section, on the other hand, describes the scattering contributions from all *pairs of nuclei*. Since the scattered neutrons behave like waves, the coherent scattering can exhibit interference effects, which manifest themselves as the peaks seen in a typical neutron diffraction experiment.

In Fig. 2.3 the coherent and incoherent scattering lengths are plotted for the first 46 elements of the periodic table. This plot demonstrates why neutrons are well suited for studying chromophores; elements that are abundant in chromophores, like hydrogen (H) and carbon (C) have relatively large scattering lengths. This can be contrasted to the case of other experimental techniques like X-rays, where the cross section scales with the number of electrons bound to the atom.

2.3 Simulating neutron scattering via correlation functions

We can further massage the coherent and incoherent cross partial cross sections in Eq. 2.10 into a form that makes them easily relatable to observables in MD simulations. Start-

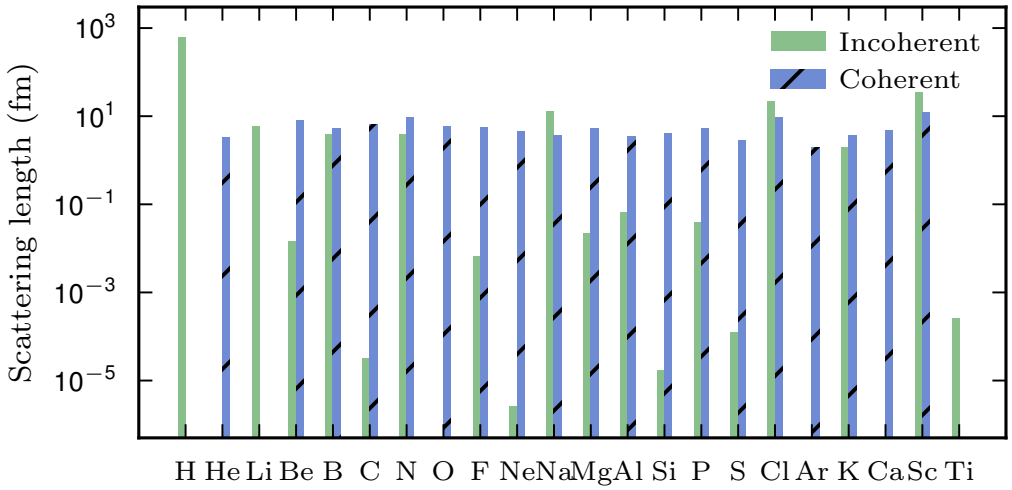


Figure 2.3: Incoherent and coherent scattering lengths for the first 46 elements in the periodic table. Note the logarithmic scale, and the large incoherent scattering length of hydrogen (H).

ing from Eq. 2.10, we can move the sums inside the integral sign and define them as the intermediate scattering function $F(\boldsymbol{q},t)$ for the coherent cross section, and the self intermediate scattering function $F_s(\boldsymbol{q},t)$ for the incoherent cross section,

$$F(\boldsymbol{q},t) = \frac{1}{N}\sum_{jj'}\left\langle e^{-i\boldsymbol{q}\cdot\boldsymbol{R}_{j'}(0)}e^{i\boldsymbol{q}\cdot\boldsymbol{R}_j(t)}\right\rangle \quad \text{and} \quad F_s(\boldsymbol{q},t) = \frac{1}{N}\sum_{j}\left\langle e^{-i\boldsymbol{q}\cdot\boldsymbol{R}_j(0)}e^{i\boldsymbol{q}\cdot\boldsymbol{R}_j(t)}\right\rangle, \tag{2.11}$$

with N being the number of nuclei in the system. From $F(\boldsymbol{q},t)$, we can further define the time-dependent pair-correlation function $G(\boldsymbol{r},t)$ and the dynamic structure factor $S(\boldsymbol{q},\omega)$,

$$G(\boldsymbol{r},t) = \frac{1}{(2\pi)^3}\int F(\boldsymbol{q},t)e^{-i\boldsymbol{q}\cdot\boldsymbol{r}}d\boldsymbol{r} \quad \text{and} \quad S(\boldsymbol{q},\omega) = \frac{1}{2\pi\hbar}\int F(\boldsymbol{q},t)e^{-i\omega t}dt. \tag{2.12}$$

Similarly, from $F_s(\boldsymbol{q},t)$ we get the self time-dependent pair-correlation function $G_s(\boldsymbol{r},t)$ and the incoherent dynamic structure factor $S_i(\boldsymbol{q},\omega)$ equivalently. We can now rewrite the coherent and incoherent cross sections in terms of $S(\boldsymbol{q},\omega)$ and $S_i(\boldsymbol{q},\omega)$,

$$\begin{aligned}\left(\frac{\partial^2\sigma}{\partial\Omega\partial E'}\right)_{\text{coh}} &= \frac{\sigma_{\text{coh}}}{4\pi}\frac{|k'|}{|k|}NS(\boldsymbol{q},\omega)\\ \left(\frac{\partial^2\sigma}{\partial\Omega\partial E'}\right)_{\text{inc}} &= \frac{\sigma_{\text{inc}}}{4\pi}\frac{|k'|}{|k|}NS_i(\boldsymbol{q},\omega).\end{aligned} \tag{2.13}$$

Eq. 2.13 is quite remarkable. If we can calculate $S(\boldsymbol{q},\omega)$ and $S_i(\boldsymbol{q},\omega)$ from an MD simulation, then we could estimate the partial differential cross section, and by extension the

intensity that one would measure in a neutron scattering experiment. The remaining question is, how do we calculate $S(\boldsymbol{q}, \omega)$? Let $\rho(\boldsymbol{r}, t)$ be the particle density,

$$\rho(\boldsymbol{r}, t) = \sum_j \delta(r - \boldsymbol{R}_j(t)), \tag{2.14}$$

which we can Fourier transform in space to obtain $F(\boldsymbol{q}, t)$ (Eq. 2.11),

$$\begin{aligned} \rho(\boldsymbol{q}, t) &= \sum_j e^{-i\boldsymbol{q}\cdot\boldsymbol{R}_j(t)} \\ \Rightarrow F(\boldsymbol{q}, t) &= \frac{1}{N} \langle \rho(\boldsymbol{q}, 0)\rho(-\boldsymbol{q}, t)\rangle \end{aligned} \tag{2.15}$$

We can get the self intermediate scattering function, $F_s(\boldsymbol{q}, t)$, by only considering the terms in which $j = j'$ in Eq. 2.15. From Eq. 2.12 we know that we can obtain $S(\boldsymbol{q}, \omega)$ and $S_i(\boldsymbol{q}, \omega)$ by Fourier transforming $F(\boldsymbol{q}, t)$ and $F_s(\boldsymbol{q}, t)$ respectively in time. In a simulation we know the positions of all atoms, and, thus, we can record $\rho(\boldsymbol{r}, t)$ throughout the simulation (the "trajectory"). We can then after the fact compute the double Fourier transform in time and space over this trajectory, and via Eq. 2.15 and Eq. 2.12 compute $S(\boldsymbol{q}, \omega)$ and $S_i(\boldsymbol{q}, \omega)$, and by extension the simulated intensity for a neutron scattering experiment.

2.4 Takeaways

In this chapter we have seen that the partial differential cross section, which is directly related to the intensity one measures in a neutron scattering experiment, can be computed from the time dependent particle density. That is, if we know the positions of all atoms in a system as a function of time, then we could compute what their corresponding neutron scattering spectra would look like. Note however that the simulation and experiment will probably not match exactly, as we have used a number of assumptions in the derivations in this chapter. First, the partial differential cross section is related to the intensity, but there are other factors such as the resolution function of the specific instrument that determines exactly how these two relate, which we have not discussed in this chapter. Second, in deriving the expressions for the partial differential cross section we have used Fermi's Golden rule and the Fermi pseudopotential, which are approximative expressions. However, this approach is exact enough for our purposes, and we will see in the next chapter how we can use molecular dynamics simulations to obtain the time dependent particle density for molecular systems.

3

Molecular Dynamics

Thermodynamics is something you can dwell on when you retire

An unnamed previous
PhD student at the division

Phonons go brrrr

Petter

Sluta prata om fononer Petter

Pernilla, 2024

In the previous chapter we equipped ourselves with knowledge of how the measured intensity in a neutron scattering experiment directly corresponds to the time-varying particle density, which fully describes the structure and dynamics of, e.g., a system of chromophores. We now turn to how we can obtain the time-varying particle density using molecular dynamics (MD) simulations, which in essence allows the prediction of neutron scattering experiments. We begin the chapter by discussing the basics of MD, where we will focus on how to extract measurable quantities from the simulations. MD simulations will, thereafter, be applied to studying the structure and dynamics for some prototypical chromophore systems.

3.1 Modeling dynamics in an atomic system

MD is a simulation technique in which the position and velocity of each atom in the system is evolved in time. The propagation in time is done incrementally, increasing by a small amount of time called the *time step*, Δt. From one time step to the next, the position and velocity of each atom is updated in accordance with the forces acting on the atom from all other atoms. Typically, an MD simulation is on the order of 1 ps to 1 μs, with a typical time step on the order of 1 fs. A simulation thus consists of many small steps in which the position of each atom is updated.

The positions of the atoms are monitored, and from the movements of the atoms one can extract estimates from physical quantities such as temperature, pressure, density, and so forth. Since each atom has to be considered individually, a typical MD simulation can at most have a system size of up to hundreds of thousands or millions of atoms, which is far off from the $\sim 10^{23}$ atoms in a macroscopic sample that one would encounter in an experiment. MD simulations are often performed in a simulation box with volume V that has so-called periodic boundary conditions (PBC) in order to compensate for this limitation. Under PBC, the atoms crossing the boundary of the simulation box will reappear at the other side of the box, which in effect approximates an infinite system if the simulation box is large enough. See Fig. 3.1a for a schematic visualization of PBC, and Fig. 3.1b for an example of a simulation box.

To be more precise, in a MD simulation we work directly with the positions of the individual atoms, denoted $\boldsymbol{r}_i$ for atom i. Note that this means that we operate within the Born-Oppenheimer approximation, in the sense that we consider the movement of the electrons in the system to be averaged out, leaving us free to consider the position of the nuclei as $\boldsymbol{r}_i$. The movements of these atoms are governed by *Newton's equations of motion* [42, Chapter 3],

$$m_i \ddot{\boldsymbol{r}}_i = \boldsymbol{f}_i \quad \text{and} \quad \boldsymbol{f}_i = -\nabla_{\boldsymbol{r}_i} \phi(\boldsymbol{r}_1, ..., \boldsymbol{r}_i, ..., \boldsymbol{r}_N), \tag{3.1}$$

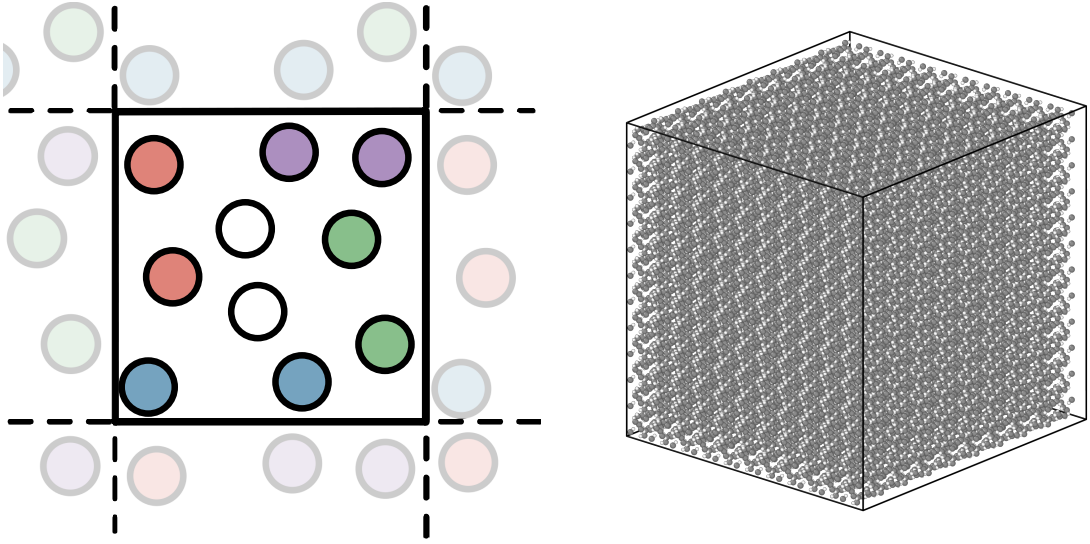
where m_i is the mass of atom i, $\phi(...)$ is the potential which is a function of all the N atoms in the system, and $\boldsymbol{f}_i$ is the total force acting on atom i from all other atoms.

Eq. 3.1 is a second-order differential equation, the solutions $\boldsymbol{r}_i(t)$ of which yield the *trajectory* for atom i. This equation can be solved using an iterative scheme known as the *Velocity-Verlet algorithm* [43], which comprises two steps. First, given a current time t, the velocity at half a time step in the future, $\boldsymbol{v}(t + \frac{1}{2}\Delta t)$ is estimated from the acceleration, and thus the force $\boldsymbol{f}_i(t)$, at time t,

$$\boldsymbol{v}\left(t + \frac{1}{2}\Delta t\right) = \boldsymbol{v}(t) + \frac{\Delta t}{2}\boldsymbol{a}(t). \tag{3.2}$$

From this estimate of the velocities, the positions at a time step in the future can be computed,

$$\boldsymbol{r}\,(t + \Delta t) = \boldsymbol{r}(t) + \Delta t \boldsymbol{v}\left(t + \frac{\Delta t}{2}\right) = \boldsymbol{r}(t) + \Delta t \boldsymbol{v}(t) + \frac{\Delta t^2}{2}\boldsymbol{a}(t). \tag{3.3}$$



(a) Schematic of PBC

(b) A realistic simulation box

Figure 3.1: a) Illustration of Periodic Boundary Conditions (PBC). Atoms and interactions between them are allowed to cross the boundary of the simulation box, represented by the black square, and interact with the images of the atoms of the other end of the cell. As long as the box is sufficiently large such that an atom will not interact with itself, this scheme effectively represents an infinite system. **b)** A realistic MD simulation box of crystalline benzene containing roughly 60 000 atoms.

Second, the positions $\boldsymbol{r}(t + \Delta t)$ can be used to compute the forces and thus the acceleration at time $t + \Delta t$, which in turn gives the velocities at time $t + \Delta t$,

$$\boldsymbol{v}\left(t + \Delta t\right) = \boldsymbol{v}\left(t + \frac{1}{2}\Delta t\right) + \frac{\Delta t}{2}\boldsymbol{a}\left(t + \Delta t\right) = \boldsymbol{v}(t) + \Delta t\frac{\boldsymbol{a}(t) + \boldsymbol{a}(t + \Delta t)}{2}. \tag{3.4}$$

The computationally expensive step in this scheme is the evaluation of the acceleration $\boldsymbol{a}\left(t + \Delta t\right)$, which involves computation of the forces. We will return to how to obtain these forces in the next chapter, using so-called force fields or interatomic potentials to calculate the forces between all the atoms in the system.

3.2 Extracting information from molecular dynamics simulations

We now turn to how we extract estimates of quantities of interest from the trajectory of positions $\boldsymbol{r}_i(t)$ and velocities $\boldsymbol{v}_i(t)$ that we obtain from the Velocity-Verlet scheme in Eq. 3.2, Eq. 3.3 and Eq. 3.4. For a system of N atoms, the instantaneous state can be described by $3N$ position and $3N$ velocity components, which can be collected into a

$6N$-dimensional vector $\mathbf{\Gamma}$. $\mathbf{\Gamma}$ is a point in the *phase space* of the system, with each point in phase space representing a different configuration of positions and velocities of the atoms. $\mathbf{\Gamma}$ is time-dependent, $\mathbf{\Gamma}(t)$, and thus the instantaneous value of any observable A at time t is $A(\mathbf{\Gamma}(t))$. $A(t)$ could for instance be the density $\rho(t)$, which is directly a function of the positions $\boldsymbol{r}_i(t)$ that are included in $\mathbf{\Gamma}(t)$. Within this formulation, an estimate for A can be extracted by averaging the instantaneous values $A(\mathbf{\Gamma}(t))$ in time over the N_t time steps of the simulation

$$A_{\text{estimate}} \approx \langle A(\mathbf{\Gamma}(t))\rangle_t = \frac{1}{N_t}\sum_{i=1}^{N_t} A(\mathbf{\Gamma}(t_i)). \tag{3.5}$$

N_t has to be sufficiently large such that the estimate A_{estimate} is converged to a sufficient degree. If the number of steps N_t becomes infinite, it is possible that the simulation will have visited every possible configuration $\mathbf{\Gamma}$ in phase space. If that is the case, the system is so-called *ergodic*, and we may replace the time-average in Eq. 3.5 by an ensemble average [42, Chapter 2],

$$\begin{aligned}&\langle A(\mathbf{\Gamma}(t))\rangle_t \iff \langle A\rangle_{\text{ensemble}} \Rightarrow \\ &A_{\text{estimate}} = \langle A\rangle_{\text{ensemble}} = \sum_{\mathbf{\Gamma}} A(\mathbf{\Gamma})\rho_{\text{ensemble}}(\mathbf{\Gamma}).\end{aligned} \tag{3.6}$$

The ensemble average is a weighted average over all configurations in phase space $\mathbf{\Gamma}$ by the probability of observing that configuration, denoted $\rho_{\text{ensemble}}(\mathbf{\Gamma})$. However, since the positions and velocities for each atom in the system can be changed continuously there are an infinite number of possible states $\mathbf{\Gamma}$, which means that the sum over $\mathbf{\Gamma}$ in Eq. 3.6 is actually an integral,

$$\langle A\rangle_{\text{ensemble}} = \int_{\mathbf{\Gamma}} A(\mathbf{\Gamma})\tilde{\rho}_{\text{ensemble}}(\mathbf{\Gamma})d\mathbf{\Gamma}. \tag{3.7}$$

Note that $\tilde{\rho}_{\text{ensemble}}(\mathbf{\Gamma})$ is now a probability distribution, and thus the probability of observing a specific configuration $\mathbf{\Gamma}$ is $\tilde{\rho}_{\text{ensemble}}(\mathbf{\Gamma})d\mathbf{\Gamma}$.

To formulate $\tilde{\rho}_{\text{ensemble}}(\mathbf{\Gamma})$, we need to take a quick detour into statistical mechanics. The probability distribution for the possible accessible states $\mathbf{\Gamma}$ depends on which *thermodynamic ensemble* the simulation is conducted in. The thermodynamic ensemble is determined by which macroscopic variables are kept constant. So far, we have not allowed the number of particles or the simulation box in the MD simulation to change, which means that the states ($\mathbf{\Gamma}$) we are considering all keep N and V constant. Furthermore, Newton's equations of motion that we introduced in Eq. 3.1 in the previous section fulfill conservation of energy, meaning that the total energy E as a sum of potential energy and kinetic energy in the system is constant. This is known as the *micro-canonical* or NVE ensemble [42, Chapter 2], and the states in phase space ($\mathbf{\Gamma}$) thus lies on an isosurface where

N, V and E are all constant. One central assumption in statistical mechanics is that all accessible states are equally probable [44]. The probability distribution $\tilde{\rho}_{NVE}(\mathbf{\Gamma})$ is thus a uniform distribution,

$$\tilde{\rho}_{NVE}(\mathbf{\Gamma}) = \frac{1}{Z_{NVE}}, \tag{3.8}$$

where Z_{NVE} is called the *partition function* and is in this case equal to the number of states in $\mathbf{\Gamma}$ that keep N and V constant, and have energy E.

If one now allows the total energy of the system E to change by allowing the system to exchange energy with a heat bath at a constant temperature T, we arrive at the *canonical* or NVT ensemble. The accessible states ($\mathbf{\Gamma}$) now have the same number of particles and the same volume as for the micro-canonical ensemble, but the energy of each state $E_{\mathbf{\Gamma}}$ may be different. The probability distribution in this case $\tilde{\rho}_{NVT}(\mathbf{\Gamma})$ is known as the *Boltzmann distribution* [44],

$$\tilde{\rho}_{NVT}(\mathbf{\Gamma}) = \frac{e^{-E_{\mathbf{\Gamma}}/k_BT}}{\sum_{\mathbf{\Gamma}'} e^{-E_{\mathbf{\Gamma}'}/k_BT}} = \frac{e^{-E_{\mathbf{\Gamma}}/k_BT}}{Z_{NVT}}. \tag{3.9}$$

Similarly, by additionally allowing the simulation box volume V to change we arrive at the *isothermal-isobaric* or NPT ensemble, which has the following probability distribution [42],

$$\tilde{\rho}_{NPT}(\mathbf{\Gamma}) = \frac{e^{(-E_{\mathbf{\Gamma}}+PV)/k_BT}}{\sum_{\mathbf{\Gamma}'} e^{(-E_{\mathbf{\Gamma}'}+PV)/k_BT}}, = \frac{e^{(-E_{\mathbf{\Gamma}}+PV)/k_BT}}{Z_{NPT}}, \tag{3.10}$$

where P is the pressure, and is kept constant. For completeness, if one further allows the number of particles N to vary, one arrives at the *grand canonical ensemble*.

There exists a multitude of different thermodynamic ensembles that one can sample, but these three, the NVE, NVT and NPT ensembles are the most common in the context of MD simulations. Of these, the NPT ensemble is the one that most closely match experimental conditions, since experiments are typically conducted at constant temperature and pressure. Recall, however, that the equations of motion that we outlined in the previous chapter only enable sampling in the NVE ensemble. In order to sample the NVT or NPT ensemble, we thus need to modify the equations of motion in order to keep the temperature and pressure constant, respectively. This can be done by introducing a *thermostat* for controlling the temperature, and a *barostat* for controlling the pressure in the simulation. One example of a commonly used thermostat is the canonical velocity rescaling thermostat proposed by Bussi *et al.* [45], and a popular barostat is the Parinello-Rahman barostat in which the simulation box is allowed to change size and shape to match the target pressure P [46]. There exists a large body of literature on different thermostats and barostats in addition to these, with different benefits and drawbacks, but common to all of them is that they directly influence the dynamics of the atoms in the simulation in order to keep the temperature and the pressure fixed, respectively. This can lead to artifacts, and hence in situations when one is interested in

the unaltered dynamics of a system the MD simulation is often conducted in the NVE ensemble which obeys Newton's equations of motion directly.

Equipped with the knowledge of what an ensemble average is, we can return to the scattering function defined in the previous chapter (Eq. 2.11). There, we saw that the intermediate scattering function $F(\boldsymbol{q}, t)$ was proportional to the thermal average of the Fourier transform of the time-dependent particle density $\rho(\boldsymbol{q}, t)$, which in turn was related to the partial differential cross section via a Fourier transform (see Eq. 2.12 and Eq. 2.13)

$$\begin{aligned} \rho(\boldsymbol{r}, t) &= \sum_j \delta(r - \boldsymbol{R}_j(t)), \\ F(\boldsymbol{q}, t) &= \frac{1}{N} \langle \rho(\boldsymbol{q}, 0) \rho(-\boldsymbol{q}, t) \rangle , \\ \Rightarrow \left(\frac{\partial^2 \sigma}{\partial \Omega \partial E'} \right)_{\text{coh}} &= \frac{\sigma_{\text{coh}}}{8\pi\hbar} \frac{|k'|}{|k|} \int \langle \rho(\boldsymbol{q}, 0) \rho(-\boldsymbol{q}, t) \rangle_{\text{thermal}} \, e^{-i\omega t} dt. \end{aligned} \tag{3.11}$$

The thermal average defined in Eq. 2.4 is the same as an ensemble average conducted in the NVT ensemble, as can be seen from Eq. 3.9. Through ergodicity, we can rewrite the thermal average as a time average, to arrive at

$$\Rightarrow \left(\frac{\partial^2 \sigma}{\partial \Omega \partial E'} \right)_{\text{coh}} = \frac{\sigma_{\text{coh}}}{8\pi\hbar} \frac{|k'|}{|k|} \int \langle \rho(\boldsymbol{q}, 0) \rho(-\boldsymbol{q}, t') \rangle_t \, e^{-i\omega t'} dt'. \tag{3.12}$$

Eq. 3.12 states that if we conduct an MD simulation in the NVT ensemble, and the assumption of ergodicity holds, then we can compute the coherent partial differential cross section from the Fourier transform of the time average of the correlation of the particle density $\rho(\boldsymbol{q}, t)$. We can similarly obtain the incoherent partial differential cross section.

The restriction to the NVT ensemble for the MD simulation is only illusory. In the thermodynamic limit, $N \to \infty$, a thermal average of a function A in the NVE ensemble is identical to one in the NVT ensemble if the temperature T is chosen such that $E = \langle E \rangle_{NVT}$, since E and $\beta = 1/k_B T$ are *conjugate variables* [42]. For this to hold, the function A should be a sum of single particle functions, which is true for the particle density $\rho(\boldsymbol{r}, t)$. Note, however, that the fluctuations of the property A is different between NVE and NVT, even if the average is the same, which may lead to certain properties that are derived from the fluctuations being different. Taken together, since the scattered intensity in a neutron scattering experiment is proportional to the partial differential cross section, this means that we can use MD in either the NVT or NVE ensembles to simulate neutron scattering experiments.

3.3 Studying liquid chromophores with molecular dynamics

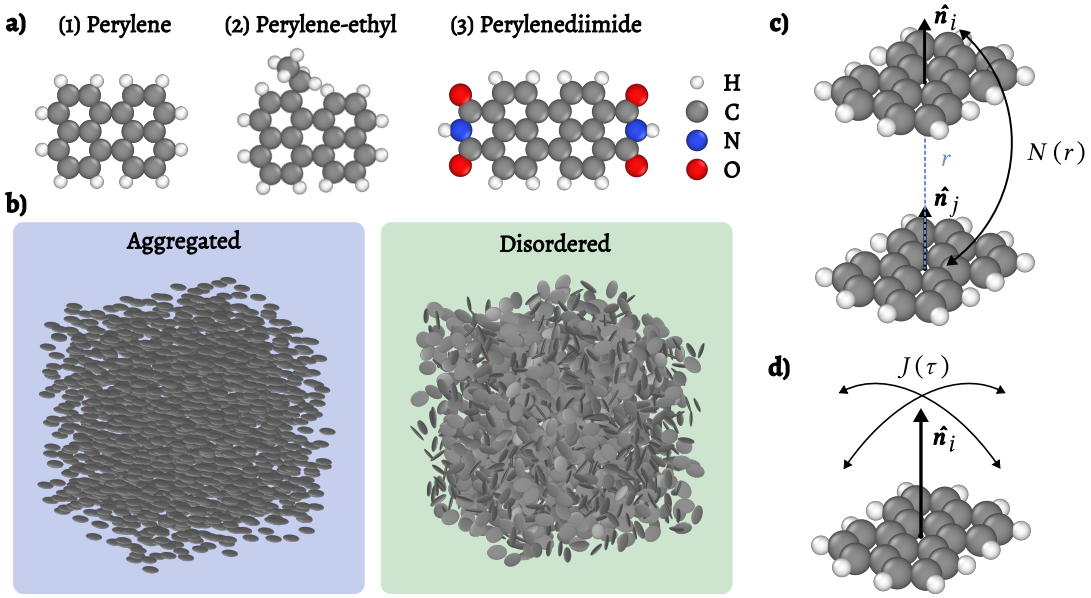


Figure 3.2: The three molecules (a) and the two structural models (b) that we studied in paper III, together with schematic visualizations of the two correlation functions that were calculated (c-d). The same-time normal vector correlation function in panel (c) describes the degree to which neighboring molecules are aligned, and the normal vector autocorrelation function (d) describes the movement of a single molecule as a function of time.

We will now focus on how to apply MD simulations for analyzing the structure and dynamics of systems of liquid chromophores. There are many possible observables that one can choose to study during an MD simulation, and one such observable that we studied in paper I are correlation functions. In this paper, we studied three different prototype systems consisting of ~1500 molecules of the same species. The three molecules were perylene, perylene-ethyl and perylenediimide. In particular we were interested in the stability of supramolecular aggregates of these molecules and changes in their dynamics as a function of temperature, in order to gain insight into which molecular mechanics affect aggregate stability. To this end, we conducted MD simulations at temperatures in the range 200 K to 600 K. At each temperature and molecule, a simulations were carried out starting from two different structural models — an aggregate structure and a disordered structure. The molecules and structural models are visualized in Fig. 3.2a and b, respectively.

From each simulation, we compute two normal vector correlation functions, defined

as

$$N(r) = \left\langle \sum_{i=1}^{N} \sum_{j \neq i} \delta(\boldsymbol{r} - \boldsymbol{r}_{ij}) \hat{\boldsymbol{n}}_i(t) \cdot \hat{\boldsymbol{n}}_j(t) \right\rangle_t \tag{3.13}$$
$$J(\tau) = \langle \hat{\boldsymbol{n}}_i(t) \cdot \hat{\boldsymbol{n}}_i(t+\tau) \rangle_{i,t} \,,$$

where $\boldsymbol{n}_i(t)$ is the normal vector of molecule i at time t and $\boldsymbol{r}_{ij}$ is the pairwise distance vector between molecules i and j. The normal vector is defined as pointing out of plane of the molecule, and is schematically visualized together with $N(r)$ and $J(\tau)$ in Fig. 3.2c-d. $N(r)$ is called the same-time normal vector correlation function, and describes the average relative orientation of molecules at a distance r. Typically, at large distances r, $N(r)$ is averaged over many molecules and approaches a limiting value that we call N_{lim}. If the neighboring molecules are oriented in the same direction $N_{\text{lim}} \simeq 1$, otherwise N_{lim} is small. $J(\tau)$ is the normal-vector auto-correlation function, and describes how the orientation of a molecule changes as a function of time. $J(\tau)$ exhibits a double-exponential decay, and decays more quickly the more rapidly the molecules change their orientation. By fitting a double exponential function $J(\tau)$,

$$J(\tau) = A_1 \exp(-t/\tau_1) + A_2 \exp(-t/\tau_2), \tag{3.14}$$

one can extract the typical timescales of the motion of the molecules, τ_1 and τ_2. The larger of these, τ_2, corresponds to reorientation of the molecules, which is the type of large-scale molecular motion we are interested in. By studying the temperature dependence of N_{lim} and τ_2, we can gain insight into the stability of the aggregate structure, and its dynamics.

In Fig. 3.3a we can see that the average orientation N_{lim} has a strong temperature dependence for perylene and perylene-ethyl for the aggregate structure. For low temperatures, N_{lim} is large which indicates that the molecules are oriented in the same direction and remain in the aggregate structure, but at 400 K N_{lim} decays quickly, indicating that the system transitions into a disordered structure. From these observations we may draw the conclusion that the aggregate structure is stable for temperatures <400 K for perylene and perylene-ethyl, at least on the timescale of 10 ns MD simulations. Perylenediimide, on the other hand, does not exhibit such a strong temperature dependence, as it remains in the aggregate structure throughout the temperature range. This can possibly be attributed to the larger and bulkier size of perylenediimide, which leads to stronger intermolecular interactions between neighboring molecules.

Turning to the speed of molecular reorientation as described by the reorientation time τ_2 in Fig. 3.3b and Eq. 3.14, we observe that it also increases by several orders of magnitude for perylene and perylene-ethyl as temperature is increased, from $\tau_2 \approx 10$ ns to $\tau_2 \approx 10$ ps, whilst the dynamics remain on the order of 10 ns for perylenediimide. Note that this effect is independent of the structural model, as both aggregate and disordered

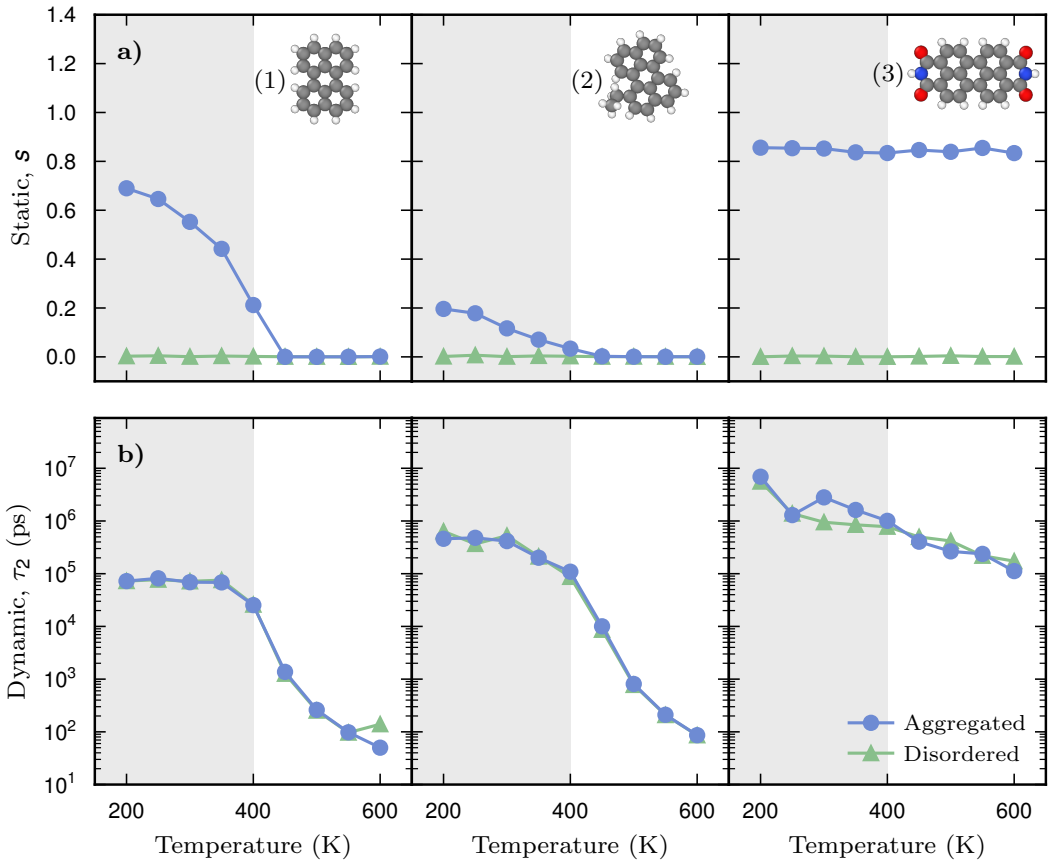


Figure 3.3: N_{lim} (a) and τ_2 (b) as a function of temperature for the three different molecular systems. Both N_{lim} and τ_2 exhibit a clear temperature dependence for the smaller perylene derivatives, perylene (1) and perylene-ethyl (2), with the aggregate structure becoming unstable at 400 K. Perylenediimide (3) remains in the aggregate structure regardless of temperature, which can be attributed to the stronger intermolecular interactions between the bulkier perylenediimide molecules.

systems exhibit a similar temperature dependence. The rapid increase in τ_2 by several orders of magnitude is reminiscent of glass transitions, which are characterized by the large-scale dynamics in the system becoming slower with decreasing temperature [47].

In short, this study exemplifies how MD simulations can be used to study liquid chromophores. However, MD simulations are limited by simulation length, which is particularly impactful in the study of glassy systems. The MD simulations conducted in this study had a simulation length of 10 ns. For comparison, when glass transitions and glassy systems are studied experimentally, one typically denotes the glass transition temperature as the temperature at which the dynamics in the system are on the order of the timescale of the experiment, 100 s. Because of this limitation, MD simu-

lations are all but guaranteed to overestimate the glass transition temperature, as the dynamics may appear frozen on the timescale of simulation but may be non-glassy at experimental conditions. With that being said, the role of simulations is not to serve as a substitute for experiments, but rather as a compliment, as one can in detail study the behavior of the system at an atomic or molecular level.

3.4 Takeaways

In this chapter, we have seen how MD simulations are performed and how they can be used to estimate various physical quantities. We have also seen that they have a clear link to the measured intensity in neutron scattering experiments via the time-dependent particle density (Eq. 3.12), which we can use to predict experimental neutron scattering results. Finally, we took an example from paper I in which we applied MD simulations to a system of perylene derivatives, and showed that it is indeed a useful technique for studying the structure and dynamics of liquid chromophores. However, the keen-eyed reader might have spotted one key ingredient in this simulation soup that I have systematically ignored, namely, the question of how one obtains the forces $\boldsymbol{f}_i$ acting on atom i that are required to solve the equations of motions in MD simulations (Eq. 3.1). The next chapter of this thesis will be fully dedicated to exploring this question.

4

Machine-learned Force Fields

> My job is to predict DFT, not reality.
>
> *Fredrik, 2022*

> I will primarily use unnecessarily complicated and highly inefficient but really huge neural network models to burn a lot of computer time.
>
> *Paul, 2024*

We have now arrived at the final, and possibly most important, piece of the puzzle that we need to solve in order to accurately simulate neutron scattering experiments with MD simulations, namely how one obtains the forces between atoms. The quality of the simulation depends on the quality of these forces, as they determine the dynamics. If the forces are inaccurate, so will the MD simulation be, and by extension, predicted observables such as neutron scattering spectra. We will begin this chapter by briefly discussing the two traditional methods that have been used for obtaining forces, which are DFT, the slow but accurate, and heuristic force fields (FFs), the fast but inaccurate. We will then dive headfirst into the world of machine-learned force fields (ML-FFs), which promise to combine the speed of heuristic FFs with the accuracy of DFT.

4.1 Electronic-structure methods and classical force fields

The goal of an FF is to provide a set of forces for a given configuration of atoms. Atomic systems behave according to the laws of quantum mechanics, and subsequently all the nuclei and electrons in the system obey the *many-body Schrödinger equation* [48, Chapter 3],

$$\begin{aligned} \hat{H}\Psi &= E\Psi \\ \hat{H} &= -\frac{1}{2}\sum_i \nabla_i^2 - \sum_I \frac{1}{2M_I}\nabla_I^2 - \sum_{i,I}\frac{Z_I}{|\boldsymbol{r}_i - \boldsymbol{R}_I|} \\ &\quad + \frac{1}{2}\sum_{i\neq j}\frac{1}{|\boldsymbol{r}_i - \boldsymbol{r}_j|} + \frac{1}{2}\sum_{I\neq J}\frac{Z_I Z_J}{|\boldsymbol{R}_I - \boldsymbol{R}_J|}, \end{aligned} \tag{4.1}$$

where we have used atomic units ($e = 4\pi\epsilon_0 = \hbar = 1$) for readability. Lower-case indices run over electrons and uppercase indices run over nuclei, which have mass M_I and atomic number Z_I. $\boldsymbol{r}_i$ denotes the position of electron i and $\boldsymbol{R}_J$ is the position of nucleus J. Solving Eq. 4.1 would give us a complete description of the system at the current moment in time, including all the forces between the atoms, which we need in order to propagate an MD simulation. However, the many-body Schrödinger equation is only analytically solvable in the simplest of cases involving a single electron, and numerically only for moderately sized or high-symmetry systems [49]. One major hurdle in solving Eq. 4.1 are the terms involving pairs of particles, denoted by the sums over $i \neq j$, i, I, and $I \neq J$, which scale exponentially in complexity as the number of particles in the system increases. DFT proposes to solve this problem by reformulating the Schrödinger equation. First, the slow-moving nuclei are assumed to be stationary relative to the electrons, which is known as the Born-Oppenheimer approximation [50]. Second, the fully-interacting electrons are replaced by N fictitious non-interacting electrons that still have the same ground-state density $n(\boldsymbol{r})$ through the Kohn-Sham (KS) equations ([49, 51]),

$$\begin{aligned} \left(-\frac{\nabla^2}{2} + v_s[n](\boldsymbol{r})\right)\varphi_j(\boldsymbol{r}) &= \epsilon_j\varphi_j(\boldsymbol{r}) \\ v_s[n](\boldsymbol{r}) &= v(\boldsymbol{r}) + \int d^3r' \frac{n(\boldsymbol{r'})}{|\boldsymbol{r} - \boldsymbol{r'}|} + v_{xc}[n](\boldsymbol{r}) \\ v_{xc}[n](\boldsymbol{r}) &= \frac{\delta E_{xc}[n]}{\delta n(\boldsymbol{r})} \\ n(\boldsymbol{r}) &= \sum_{j=1}^{N} |\varphi_j(\boldsymbol{r})|^2. \end{aligned} \tag{4.2}$$

$\varphi_j(\boldsymbol{r})$ is the KS orbital for state j of the non-interacting system, with associated energy ϵ_j, and $[n]$ denotes that a quantity is a *functional* of the electron density $n(\boldsymbol{r})$. A key feature of the KS equations is that the ground state density $n(\boldsymbol{r})$ is a function of $\{\varphi_j(\boldsymbol{r})\}_j$, which in turn also depends on the density. The equations thus have to be solved self-consistently, by iteratively solving for and updating the density $n(\boldsymbol{r})$ until it has converged. Once the ground state density $n(\boldsymbol{r})$ has been obtained one can compute relevant physical properties such as the forces between the nuclei. Note that the external potential $v(\boldsymbol{r})$ includes the Coulomb interaction from the nuclei acting on the electrons.

The KS equations are formally exact, and could be solved perfectly if the *exchange-correlation energy functional* $E_{xc}[n]$ and it's associated potential $v_{xc}[n](\boldsymbol{r})$ was known. Unfortunately, $E_{xc}[n]$ is in general unknown, leaving users of DFT with no other choice but to rely on approximations for $E_{xc}[n]$. There exists a plethora of DFT functionals, and depending on the choice of functional a DFT calculation can yield different results. As such, DFT calculations are inherently dependent on the choice of functional.

The KS equations (Eq. 4.2) are significantly easier to solve computationally than the many-body Schrödinger equation (Eq. 4.1), but the KS equations still scale poorly with the number of electrons in the system. In practice, DFT is often not used for systems with more than hundreds or possibly thousands of atoms, which is short of the tens of thousands to millions of atoms that are typically studied in MD simulations. Heuristic FFs takes a more computationally efficient but potentially less accurate approach by foregoing the Schrödinger equation entirely. Instead, one deals with an expansion of the system energy in terms depending on the positions $\boldsymbol{r}_i^N$ of the atoms in the system [42],

$$V(\boldsymbol{r}_i) = \sum_i V_1(\boldsymbol{r}_i) + \sum_i \sum_{j>i} V_2(\boldsymbol{r}_i, \boldsymbol{r}_j) + \sum_i \sum_{j>i} \sum_{k>j>i} V_2(\boldsymbol{r}_i, \boldsymbol{r}_j, \boldsymbol{r}_k) + \ldots \tag{4.3}$$

where the first term, V_1, corresponds to interactions of all N atoms with an external potential, the second term with V_2 corresponds to interactions between pairs of particles, the third sum to interactions between triplets and so forth. Including more terms in Eq. 4.3 gives in general a more accurate but less computationally efficient model. Truncating the expansion at the V_2 term gives a class of potentials known as *pair potentials*, the most well-known of which is the *Lennard-Jones* potential [42],

$$V^{LJ}(\boldsymbol{r}_i) = \sum_i \sum_{j>i} 4\epsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^{6} \right]. \tag{4.4}$$

The Lennard-Jones potential involves two parameters, ϵ and σ, which control the shape of the potential. These parameters are adjusted such that the potential gives reasonable estimates for various physical properties such as, e.g., the density of the system under study when used in an MD simulation, which yields a potential that is tailored for a specific system. This is a common strategy for heuristic FFs in general, where the main

difference between different potentials is the type of interactions considered and the number of free parameters available in the model. Once optimized, heuristic FFs such as the Lennard-Jones potential can be readily evaluated for the large number of atoms in typical MD simulations, but at the cost of not being as accurate as DFT calculations.

4.2 Machine-learned force fields and neural network potentials

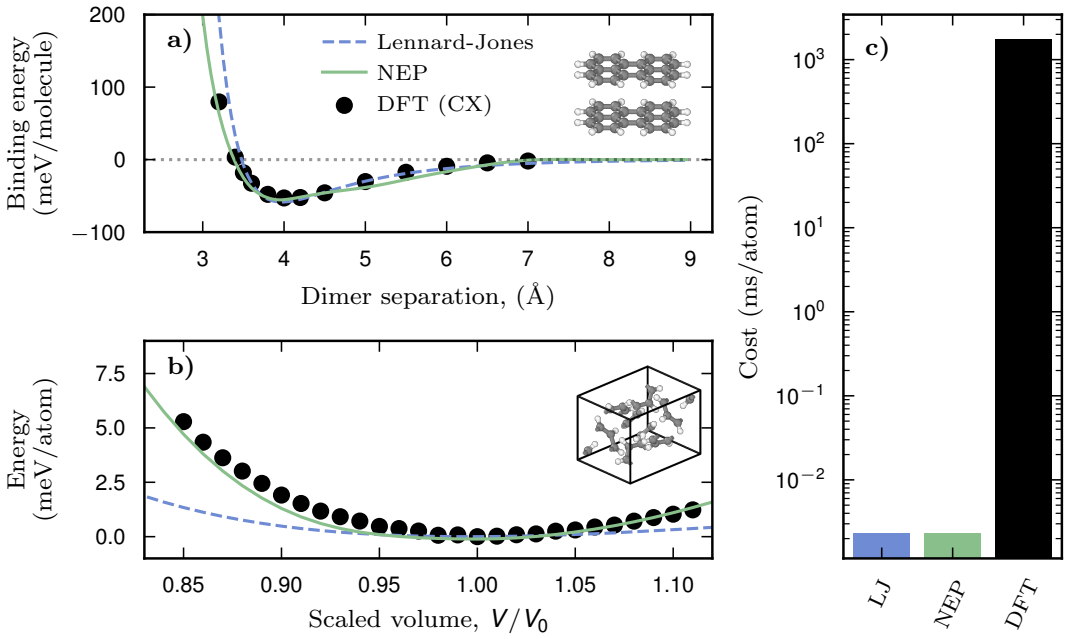


Figure 4.1: Calculation of **a)** the binding energy between two benzene molecules as a function of distance, and **b)** the change in potential energy as the unit cell of crystalline benzene is scaled, using three different methods; a Lennard-Jones potential, NEP which is an ML-FF, and DFT using the CX functional. Panel **c)** compares the computational cost of evaluating the energies with each method. The ML-FF, NEP, accurately captures the target curves from DFT, while still having the same low computational cost as the Lennard-Jones potential.

Machine-learned force fields (ML-FFs) take a similar approach as heuristic force fields (FFs), with the difference being that ML-FFs typically have a larges number of parameters which makes them very flexible and thus possibly more accurate. Here we will give a brief overview of ML-FFs; a more comprehensive review can for instance be found in reference [52].

An ML-FF is often trained to reproduce energies and forces from a training dataset of atomic configurations with reference energies and forces from DFT, with the aim of obtaining a model that is as accurate as DFT but has similar computational cost as a heuristic FF. This is demonstrated in Fig. 4.1, where the binding energy between two benzene molecules as a function of separation (Fig. 4.1a) and the change in potential energy as the axes of the unit cell of crystalline benzene are scaled (Fig. 4.1b), have been computed using a Lennard-Jones potential, an ML-FF called NEP that we will discuss in detail in Sect. 4.3, and DFT. The computational cost of evaluating the energies with each method are given in Fig. 4.1c, with DFT being about five orders of magnitude slower than the FFs. Both the Lennard-Jones potential and the NEP model predict the binding energy accurately, but the Lennard-Jones potential fails for the more complex case of predicting the energy-volume curve. The NEP model on the other hand accurately captures the energy-volume curve, whilst retaining the high computational performance the Lennard-Jones potential.

Like with heuristic FFs, there exists a large number of different types of ML-FFs based on different functional forms. These can be divided into two main categories, namely *kernel-based methods* and *neural-network based methods*.

4.2.1 Kernel-based methods

Kernel-based methods typically employ a probabilistic approach, with the assumption that the energy E as a function of the positions of N atoms takes the form of a gaussian process (GP) [53, Chapter 2],

$$\begin{aligned} E(\boldsymbol{q}) &\sim \mathcal{GP}\left(m(\boldsymbol{q}), k(\boldsymbol{q}, \boldsymbol{q'})\right) \\ m(\boldsymbol{q}) &= \mathbb{E}\left[E(\boldsymbol{q})\right] \\ k(\boldsymbol{q}, \boldsymbol{q'}) &= \mathbb{E}\left[\left(E(\boldsymbol{q}) - m(\boldsymbol{q})\right)\left(E(\boldsymbol{q'}) - m(\boldsymbol{q'})\right)\right]. \end{aligned} \tag{4.5}$$

By modeling the potential energy as a GP we restrict the potential energy function $E(\boldsymbol{q})$ to belong to a family of functions in which all the points are jointly Gaussian distributed. The GP is fully determined by its mean function $m(\boldsymbol{q})$ and its covariance function $k(\boldsymbol{q}, \boldsymbol{q'})$, with the vector $\boldsymbol{q} \in R^{3N}$ representing a specific configuration of the N atoms in the system. The covariance function can be interpreted as encoding the similarity between two configurations $\boldsymbol{q}$ and $\boldsymbol{q'}$.

In less formal terms, by modeling the potential energy surface as a GP we constrain it to be smoothly varying as the positions of the atoms represented by the vector $\boldsymbol{q}$ are changed, with each configuration having an associated predicted mean energy $E(\boldsymbol{q})$ with standard deviation $\sigma_E(\boldsymbol{q})$. This leads us to the main benefit of GP-based methods: the uncertainty in the predictions can readily be extracted as the predicted variance $\sigma_E(\boldsymbol{q})$. By monitoring the uncertainty during a MD simulation, configurations of atoms for which the model gives an inaccurate prediction can be identified. These uncertain structures

can then be included in the training dataset to improve the model, which is known as *active learning* [54, 55]. The disadvantage of kernel-based methods is that they are computationally expensive to evaluate, with the cost of evaluating the model scaling as $\mathcal{O}(n^3)$, where n are the number of data points in the training dataset, although some strategies exist for partially mitigating this limitation [56].

A specific example of a GP-based ML-FF is the Gradient-Domain Machine-Learning model (GDML, [57]), which is trained to predict the forces $\boldsymbol{F}(\boldsymbol{q})$, with the potential model being obtained by integration, $E(\boldsymbol{q}) = \int_0^{\boldsymbol{q}} \boldsymbol{F}(\boldsymbol{q'})d\boldsymbol{q'}$. sGDML is an extension of GDML in which relevant symmetries are incorporated to improve the efficiency of the model [58]. Another example of a kernel-based method is the Gaussian Approximation Potential (GAP) [59],

4.2.2 Neural network-based methods

Popularized by Behler and Parrinello [60, 61], a neural network (NN) potential denoted U with weights $\boldsymbol{w}$ can be used to predict the energies for each atom i in a system of N atoms, and can be written as follows,

$$\begin{aligned} E_i &= U(\boldsymbol{w}, \boldsymbol{q}(\{\boldsymbol{r}_{ij}\})) \\ \rightarrow \boldsymbol{F}_i &= \nabla_{\boldsymbol{r}_i} U(\boldsymbol{w}, \boldsymbol{q}(\{\boldsymbol{r}_{ij}\})). \end{aligned} \tag{4.6}$$

E_i and $\boldsymbol{F}_i$ is the per-atom potential energy and the forces acting on atom i respectively. The per-atom potential energies are summed up to yield $E = \sum_i^N E_i$, as only the total potential energy for a structure is defined, and U can thus be seen as a model for the potential energy surface of the system. The forces $\boldsymbol{F}_i$ are obtained as the gradient of U with respect to the coordinates of atom i, $\boldsymbol{r}_i$.

Uncertainty estimates cannot be as easily extracted from NN-based as from GP-based ML-FFs, but they are generally more computationally efficient. One way to estimate the uncertainty for NN ML-FFs is to train an ensemble of models and use each model to compute the forces. This gives a distribution of force predictions over the ensemble, which can be used to estimate the uncertainty [62].

Other examples of neural network-based methods include Deep Potential (DP) [63], Embedded Atom Neural Network (EANN) [64], ANI-1 [65], SchNet [66], and NEP which we will return to in Sect. 4.3.

4.2.3 Using descriptors to represent atomic structures

Both kernel-based and neural network-based ML-FFs do not typically have the Cartesian coordinates of a set of atoms, $\{\boldsymbol{r}_i\}$, as input, but rather a so-called *descriptor vector* $\boldsymbol{q}(\{\boldsymbol{r}_{ij}\})$. This is a function of the relative positions of atom i and all neighboring atoms j, and can be thought of as a chemical fingerprint describing the environment around atom

i. The reason for introducing this seemingly cumbersome descriptor vector is that it guarantees that the model fulfills certain symmetries dictated by physics.

First of all, there are a set of *invariances*, transformations of the atomic system that should not change the predicted energies of forces. These include invariance under translations and rotations of the system, and permutations of atoms with the same element [67]. Recently, *equivariances* have additionally started to be incorporated in ML-FFs [68, 69]. An equivariant transformation can, for instance, be a rotation; when the input structure is rotated by a certain amount, the output forces should be rotated accordingly. Specifically, both descriptors and the model are designed to transform equivariantly under SO(3) rotations of the input. These are often implemented using equivariant graph neural networks [70].

Understanding the descriptor vector is key to understanding the success of ML-FFs in recent years [71]. An ML model in general, and a neural network in particular, is just a "dumb" mathematical function that takes numbers as input and blindly outputs (or "predicts") other numbers. During training, the parameters of this function are adjusted such that it best mimics the examples of inputs and corresponding target outputs in the training data set. I want to stress that the model *mimics* the training data; saying that an ML "model learns" is to some extent a misnomer. Crucially, this means that an ML-FF on its own does not know anything about physics. Furthermore, as ML models typically involve many parameters interacting non-linearly, it is often difficult to know beforehand what the output will be for a specific input, a problem that becomes exponentially more difficult as the size of the model increases. Taken together, this means that an ML model can produce unexpected outputs when presented with an input that is different from the examples in the training data set, a problem that is known as out-of-distribution prediction and is widely researched in the ML literature [72]. A famous recent example of, at least in part, out-of-distribution predictions are the hallucinations of large-language models like ChatGPT [73], leading to possibly non-factual responses [74, 75]. An ML-FF can to some extent be protected from the issues of out-of-distribution prediction by incorporating a descriptor vector that ensures that the input to the ML-FF is physically meaningful. An added benefit of using a descriptor is that the model automatically fulfills the relevant symmetries, which means that these do not need to be "learnt" during training. However, even with a descriptor vector an ML-FF still suffers from the issues of out-of-distribution predictions, for instance if structures encountered during simulations are vastly different from the ones in the training data set. Like with any other ML model, ML-FFs can be hardened against this problem by ensuring that the training data set samples the chemical space of interest well, for example by using entropy-maximized datasets [76], but in practice one can never *guarantee* the robustness of the predictions of the ML-FF.

The points discussed in this section are the general considerations that go into crafting a physically accurate descriptor, the exact implementation of which can change for different ML-FFs. Examples of often-used descriptors are atom-centered symmetry

functions (ACSF) [60, 61], smooth overlap of atomic orbitals (SOAP) [77], spherical harmonics [78] and the many-body tensor representation [79]. In the next section, we will take a look at a specific implementation of an NN-based ML-FF, and discuss its associated descriptor vector in detail.

4.3 Neuroevolution potentials

We will now turn to a specific type of neural network-based machine-learned potential, namely the neuroevolution potential (NEP) developed and implemented in the GPUMD package by Zheyong Fan to which I am a contributing developer, presented in paper I. In this section, we will describe NEP models in detail, starting with the formalism.

4.3.1 The NEP formalism

Similarly to the Behler-Parinello NN-ML-FFs described in Eq. 4.6, in NEP the energy of atom i, E_i, is predicted as a function of a descriptor vector with N_{des} components, denoted $\boldsymbol{q}_i$. The NN consists of a single fully connected hidden layer, and the predicted energy takes the form

$$E_i = U\left(\boldsymbol{w}, \boldsymbol{q}\right) = \sum_{\mu=1}^{N_{\text{neu}}} w_\mu^{(1)} \tanh\left(\sum_{\nu=1}^{N_{\text{des}}} w_{\mu\nu}^{(0)} q_\nu^i - b_\mu^{(0)}\right) - b^{(1)}. \tag{4.7}$$

$\boldsymbol{w}^{(0)}$ and $\boldsymbol{b}^{(0)}$ are the weight matrix from the input descriptor vector to the hidden layer, and $\boldsymbol{w}^{(1)}$ and $b^{(1)}$ the weights and bias term from the hidden layer to the single output neuron. The activation function for the hidden layer is tanh.

Note that the parameters for the model in Eq. 4.7, which for historical reasons is known as a NEP3 model, are shared between all atoms in the system. This has the crucial benefit that the model does not increase in size as the number of atoms in the simulated system increases, which could otherwise lead to a model that is computationally impossible to evaluate for the millions of atoms in an MD simulation. However, sharing weights between all atoms in a system can lead to a insufficiently flexible model, especially in systems with many different atomic species and thus a potentially large input space of possible descriptor vectors $\boldsymbol{q}$. NEP4 increases the flexibility of NEP models by having an individual network for each atomic species α in the system, $U\left(\boldsymbol{w}^\alpha, \boldsymbol{q}\right)$ [80], yielding a possibly more accurate model.

The descriptor vector $\boldsymbol{q}$ takes the same shape for both NEP3 and NEP4, and is comprised of a radial part and an angular part (Fig. 4.2a). The radial part has $n_{\text{max}}^{\text{R}} + 1$ components and is defined as

$$q_n^i = \sum_{j \neq i} g_n(r_{ij}). \tag{4.8}$$

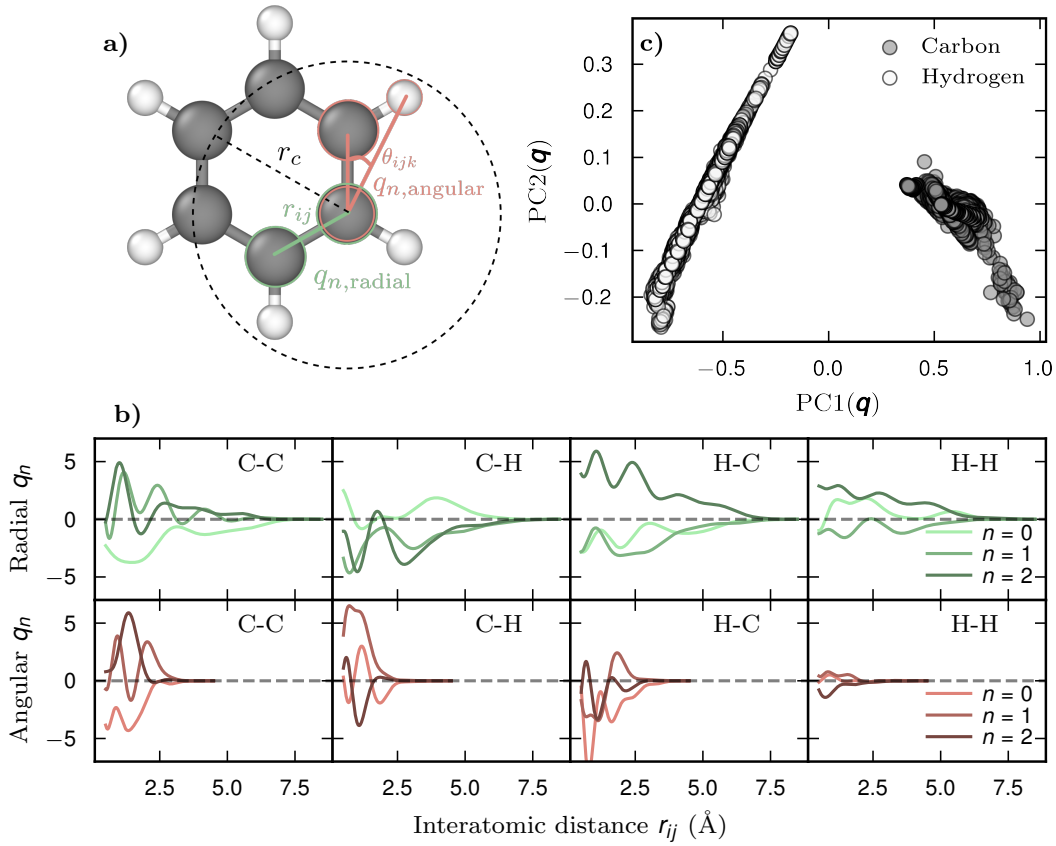


Figure 4.2: a) Schematic of radial (green) and angular (red) descriptor components. **b)** Visualization of three radial and angular descriptor components. Note that the peak position of descriptor component n varies. **c)** principal component analysis (PCA) plot of the two principal descriptor components for ~ 900 structures of crystalline benzene, colored by atomic species.

The summation runs over all neighboring atoms j to atom i, where r_{ij} is the distance between them. The contribution from each neighbor, $g_n(r_{ij})$, is in turn computed from $N_{\text{bas}}^{\text{R}} + 1$ basis functions,

$$
\begin{aligned}
g_n(r_{ij}) &= \sum_{k=0}^{N_{\text{bas}}^{\text{R}}} c_{nk}^{ij} f_k(r_{ij}) \\
f_k(r_{ij}) &= \frac{1}{2}\left[T_k\left(2\left(\frac{r_{ij}}{r_c^{\text{R}}-1}\right)^2 - 1\right) + 1\right] f_c(r_{ij})
\end{aligned}
\tag{4.9}
$$

where $T_k(...)$ is the kth-order Chebyshev polynomial of the first kind. $f_c(r_{ij})$ is a cutoff function that ensures that the contribution $g_n(r_{ij})$ from atom j decreases smoothly to

zero as the distance r_{ij} approaches the radial cutoff distance r_c^{R}, and is defined as

$$f_c(r_{ij}) = \begin{cases} \frac{1}{2}\left[1 + \cos\left(\pi \frac{r_{ij}}{r_c^{\mathrm{R}}}\right)\right], & r_{ij} \leq r_c^{\mathrm{R}} \\ 0, & r_{ij} > r_c^{\mathrm{R}}. \end{cases} \tag{4.10}$$

$n_{\mathrm{max}}^{\mathrm{R}}$, $N_{\mathrm{bas}}^{\mathrm{R}}$, and r_c^{R} are hyperparameters that are set before training. A key feature of the NEP formalism is that the coefficients c_{nk}^{ij} in the radial basis expansion are free parameters that are optimized in conjunction with the weights $\boldsymbol{w}$ and biases $\boldsymbol{b}$ of the NN. These coefficients depend on the species of atom i and j, which allows NEP to tailor the message $g_n(r_{ij})$ from each neighbor, increasing the flexibility of the model.

The angular descriptor vector are similarly defined as

$$q_{nl}^{i} = \frac{2l+1}{4\pi} \sum_{j \neq i} \sum_{k \neq i} g_n(r_{ij}) g_n(r_{ik}) P_l\left(\cos\theta_{ijk}\right) \tag{4.11}$$

with $0 < n < n_{\mathrm{max}}^{A}$ and $1 \leq l \leq l_{\mathrm{max}}^{\mathrm{3b}}$ as hyperparameters that control the size of the basis expansion. $P_l(...)$ is the Legendre polynomial of order l, and θ_{ijk} is the angle formed between the two pairs of atoms, ij and ik. This expression is a three-body descriptor as it involves three atoms; the central atom i and two neighboring atoms j and k. Higher order terms, such as four-body or five-body interactions, can additionally be included in the NEP formalism, but we will not describe those in detail here as the notation becomes rather cumbersome. Please see paper I for details.

The radial and angular components of the descriptor vectors for a benzene molecule are visualized as a function of interatomic distance r_{ij} in Fig. 4.2b. The peaks of the basis functions are in different positions for different descriptor components, which can be interpreted as the different components probing different regions of the chemical environment. Typically, a NEP model encounters a large number of chemical environments with their own descriptor vectors. This is visualized as a PCA plot in Fig. 4.2c, where the descriptors for ~900 structures of crystalline benzene are plotted. Although the descriptors for the atoms of the same species fall into similar regions, the descriptors still vary dramatically within these regions.

The descriptor vector as defined fulfills the invariance requirements we discussed in the previous section. Invariance under translation and rotation of the system is fulfilled as the descriptors only depend on the relative distance r_{ij} between pairs of atoms, as well as the angle θ_{ijk} between triplets of atoms. Furthermore, invariance under permutations of atoms of the same species is guaranteed by the summation over neighbors in Eq. 4.8 and Eq. 4.11.

It is straightforward to compute the partial force acting on atom i using the chain rule,

as both the expressions for the model and the descriptor vector are entirely analytical,

$$\begin{aligned}\frac{\partial E_i}{\partial \boldsymbol{r}_{ij}} = &\sum_{n=0}^{n_{\max}^{\mathrm{R}}} \frac{\partial E_i}{\partial q_n^i} \frac{\partial q_n^i}{\partial \boldsymbol{r}_{ij}} + \sum_{n=0}^{n_{\max}^{\mathrm{A}}} \sum_{l=1}^{l_{\max}^{\mathrm{3b}}} \frac{\partial E_i}{\partial q_{nl}^i} \frac{\partial q_{nl}^i}{\partial \boldsymbol{r}_{ij}} \\ &+ \sum_{n=0}^{n_{\max}^{\mathrm{A}}} \sum_{l=1}^{l_{\max}^{\mathrm{4b}}} \frac{\partial E_i}{\partial q_{nlll}^i} \frac{\partial q_{nlll}^i}{\partial \boldsymbol{r}_{ij}} + \sum_{n=0}^{n_{\max}^{\mathrm{A}}} \sum_{l=1}^{l_{\max}^{\mathrm{5b}}} \frac{\partial E_i}{\partial q_{nllll}^i} \frac{\partial q_{nllll}^i}{\partial \boldsymbol{r}_{ij}},\end{aligned} \tag{4.12}$$

where we have additionally included the four- and five-body angular descriptors. Note that the derivative is taken with regard to the distance vector between atoms i and j, $\boldsymbol{r}_{ij}$. We can now construct the force acting on atom i from atom j to respect Newton's third law, $\boldsymbol{F}_{ij} = -\boldsymbol{F}_{ji}$, as $\boldsymbol{F}_{ij} = \partial E_i/\partial \boldsymbol{r}_{ij} - \partial E_i/\partial \boldsymbol{r}_{ji}$. The total force acting on atom i from all neighboring atoms can be obtained by direct summation,

$$\boldsymbol{F}_i = \sum_{i \neq j} \boldsymbol{F}_{ij}. \tag{4.13}$$

The per-atom virial, from which properties such as stress and heat-current can be derived, can also be defined in terms of the partial force,

$$\boldsymbol{W}_i = \sum_{j \neq i} \boldsymbol{r}_{ij} \otimes \frac{\partial U_j}{\partial \boldsymbol{r}_{ji}}. \tag{4.14}$$

These analytical expressions for the energies, forces and virials are computationally cheap to evaluate, and since Eq. 4.7, Eq. 4.13 and Eq. 4.14 can be evaluated for all atoms in the system in parallel the NEP formalism can be very efficiently implemented on graphics processing units (GPUs).

4.3.2 Training a NEP

NEPs are trained by minimizing the following loss function, where the first three terms are the root mean squared error (RMSE) loss with regards to energy, forces and virials

respectively,

$$
\begin{aligned}
L(\boldsymbol{z}) = \lambda_e & \left(\frac{1}{N_{\text{str}}} \sum_{n=1}^{N_{\text{str}}} \left(E^{\text{NEP}}(n, \boldsymbol{z}) - E^{\text{tar}} \right)^2 \right)^{1/2} \\
& + \lambda_f \left(\frac{1}{3N} \sum_{i=1}^{N} \left(F_i^{\text{NEP}}(\boldsymbol{z}) - F_i^{\text{tar}} \right)^2 \right)^{1/2} \\
& + \lambda_v \left(\frac{1}{6N_{\text{str}}} \sum_{n=1}^{N_{\text{str}}} \sum_{\mu\nu} \left(W_{\mu\nu}^{\text{NEP}}(n, \boldsymbol{z}) - W_{\mu\nu}^{\text{tar}} \right)^2 \right)^{1/2} \\
& + \lambda_1 \frac{1}{N_{\text{par}}} \sum_{n=1}^{N_{\text{par}}} |z_n| + \lambda_2 \left(\frac{1}{N_{\text{par}}} \sum_{n=1}^{N_{\text{par}}} z_n^2 \right)^{1/2}
\end{aligned}
\tag{4.15}
$$

with $\boldsymbol{z}$ denoting the trainable parameters of the model, N_{str} the number of structures in the current batch, and N the total number of atoms in the current batch. Superscripts *NEP* and *tar* represents predicted and target values, respectively. The last two terms, weighted by factors λ_1 and λ_2, are $L1$ and $L2$-regularization terms, which makes Eq. 4.15 an elastic-net loss [81]. An elastic net combines the benefits of L1 (Lasso) and L2 (ridge) regression, and yields a sparse model.

The loss in Eq. 4.15 is minimized through a separable natural evolution strategy (SNES) [82], which is form of genetic optimization algorithm. The general idea of SNES is to optimize a distribution for each parameter, instead of a single value as in most other optimization techniques. This scheme is implemented as follows [83]. Let the parameters $\boldsymbol{z}$ be distributed according to a joint N_{par}-dimensional Gaussian distribution, $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{s})$, where $\boldsymbol{m}$ and $\boldsymbol{s}$ is the mean and standard deviation vector respectively. This parameter distribution is iteratively updated according to the natural gradient of the *fitness* $J(\boldsymbol{z})$,

$$
J(\boldsymbol{z}) = \mathbb{E}[-L(\boldsymbol{z})] = -\int L(\boldsymbol{z}) p(\boldsymbol{z}|\boldsymbol{m}, \boldsymbol{s}) d\boldsymbol{z}, \tag{4.16}
$$

which is the expected value of the loss function under the search parameter distribution, $p(\boldsymbol{z}|\boldsymbol{m}, \boldsymbol{s})$. The minus sign in Eq. 4.16 comes from SNES being a maximization procedure, but we want to minimize $L(\boldsymbol{z})$. First, N_{pop} samples $\boldsymbol{z}_k$ are drawn from the distribution, $\boldsymbol{z}_k = \boldsymbol{m} + \boldsymbol{s} \odot \boldsymbol{r}_k$ where $\boldsymbol{r}_k \sim \mathcal{N}(0, 1)$, which each can be seen as an instance of the NEP model. The symbol $\odot$ denotes the Hadamard (element-wise) product. Second, $L(\boldsymbol{z}_k)$ is evaluated for each of the N_{pop} models in the current generation, the models are sorted in ascending order of the loss score, and each of the models is assigned a value u_k according to its rank (see [84] for explicit values of u_k). Third, the natural gradient of the fitness

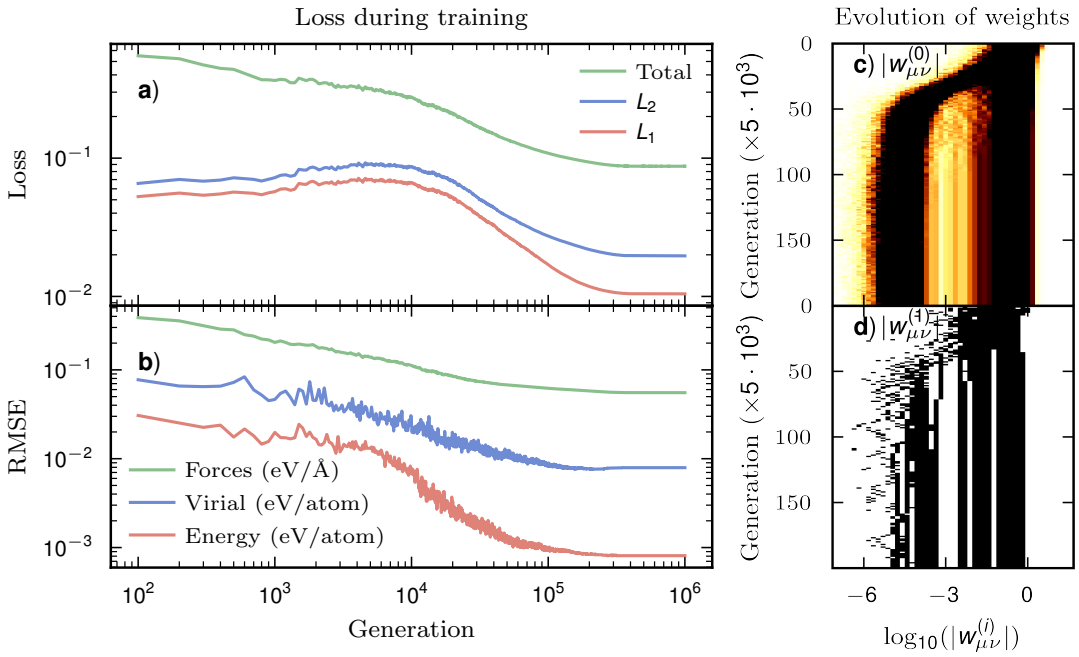


Figure 4.3: An example of how the loss function in Eq. 4.15 (panels a-b) and the magnitude of the weights in the hidden and output layers of the NN change (panels c-d) as a function of training generation. The effect of the regularization can clearly be observed in many of the model weights decreasing in magnitude as the training progresses, with start around 50 000 generations.

with regards to $\boldsymbol{m}$ and $\boldsymbol{s}$ are computed,

$$\begin{aligned} \nabla_{\boldsymbol{m}} J(\boldsymbol{z}) &= \sum_{k=1}^{N_{\text{pop}}} u_k \boldsymbol{r}_k \\ \nabla_{\boldsymbol{s}} J(\boldsymbol{z}) &= \sum_{k=1}^{N_{\text{pop}}} u_k (\boldsymbol{r}_k \odot \boldsymbol{r}_k - 1), \end{aligned} \tag{4.17}$$

which, finally, are used to update the mean and standard deviations of the parameter distribution,

$$\begin{aligned} \boldsymbol{m} &\leftarrow \boldsymbol{m} + \eta_{\boldsymbol{m}} (\boldsymbol{s} \odot \nabla_{\boldsymbol{m}} J(\boldsymbol{z})) \\ \boldsymbol{s} &\leftarrow \boldsymbol{s} \odot \exp\left(\frac{\eta_{\boldsymbol{s}}}{2} \nabla_{\boldsymbol{s}} J(\boldsymbol{z})\right). \end{aligned} \tag{4.18}$$

$\eta_{\boldsymbol{m}}$ and $\eta_{\boldsymbol{s}}$ are the equivalent of learning rates, and are set to $\eta_{\boldsymbol{m}} = 1$ and $\eta_{\boldsymbol{s}} = (3 + \ln N_{\text{par}})/5\sqrt{N_{\text{par}}}$ as suggested by [84].

Optimizing the loss function using the natural gradient instead of, e.g., the Euclidian gradient as in regular steepest descent optimization is beneficial, as the natural gradi-

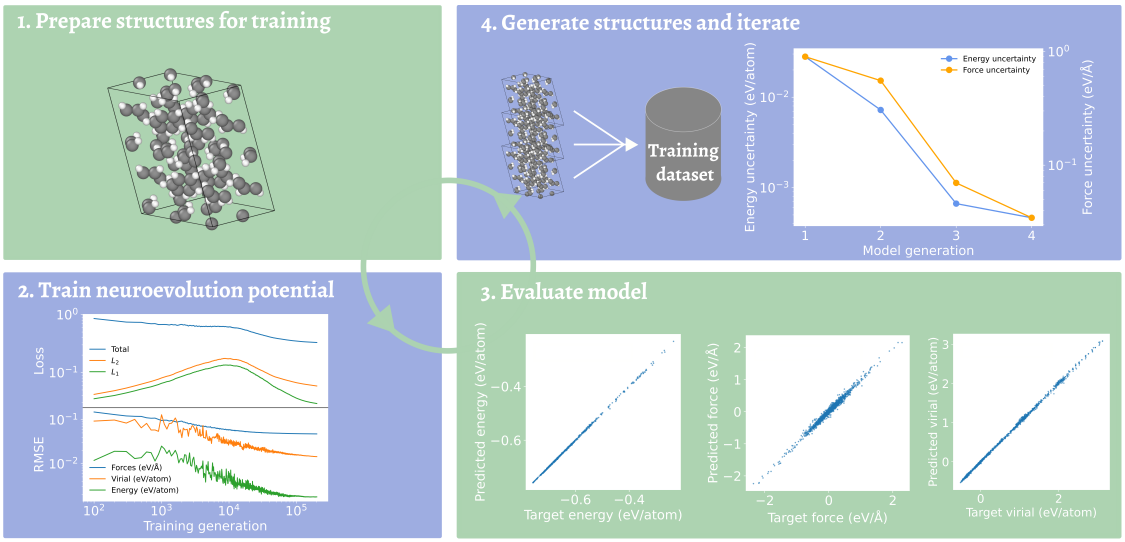


Figure 4.4: The active learning scheme often used when training NEP models. The initial dataset $\mathscr{D}_0$ consists of rattled and strained structures, after which an ensemble of models are trained and new structures are added to the training dataset for the next generation, $\mathscr{D}_{i+1}$ through active learning. This procedure is repeated until the force uncertainty $\sigma_{\boldsymbol{F}_i}$ decreases below the force RMSE of the model.

ent takes the curvature of the loss landscape into consideration. This feature, in conjunction with the elastic-net regularization and the genetic form of the algorithm, leads to a very efficient optimization scheme that yields both an accurate and sparse final model. See Fig. 4.3 for an example of how the loss function and parameter distribution evolves as training proceeds. As can be seen in Fig. 4.3b, many parameters in the trained model are small in magnitude. Such sparsity is desirable as it decreases the effect of *overfitting*, the effect where the model parameters are adjusted too tightly to the training set. Overfitting can lead to the model predicting unphysical forces when it encounters an atomic configuration which was not in the training dataset, which in turn affects the accuracy of the MD trajectory. A sparse model is less likely to predict wildly unphysical forces when extrapolating to such structures.

However, the most efficient method for minimizing the risk of the model extrapolating to unknown structures is to have a comprehensive training set, so that as much of the relevant configuration space is covered. To achieve this, the training dataset $\mathscr{D}_i$ is augmented using active learning, similar to what was touched upon in Sect. 4.2.1. Typically, an ensemble of $N_{\text{ens}} = 5$ models is trained on various random subsets of the training dataset $\mathscr{D}_i$ through a process known as *bagging* [85]. Then, a short MD simulation is run with one of the models, and for each atomic configuration the forces are predicted with each of the N_{ens} models. The uncertainty of each structure is then estimated as the force on atom i with the maximum standard deviation $\sigma_{\boldsymbol{F}_i}$ over the ensemble mod-

els. After the simulation, the structures with the largest uncertainties are selected and target energies, forces and virials are computed using DFT, after which they are added to the dataset to yield an augmented dataset $\mathscr{D}_{i+1}$. Then a new ensemble of models can be trained using $\mathscr{D}_{i+1}$. This procedure is then updated around 3-10 times, until the uncertainty of the structures encountered during MD fall beneath the force RMSE over the entire dataset, which can be seen as the noise in the predictions of the model. See Fig. 4.4 for a schematic of this active learning scheme. The initial dataset, $\mathscr{D}_0$, is typically comprised of various rattled and strained structures, starting from a reference structure via DFT.

4.3.3 NEP in practice

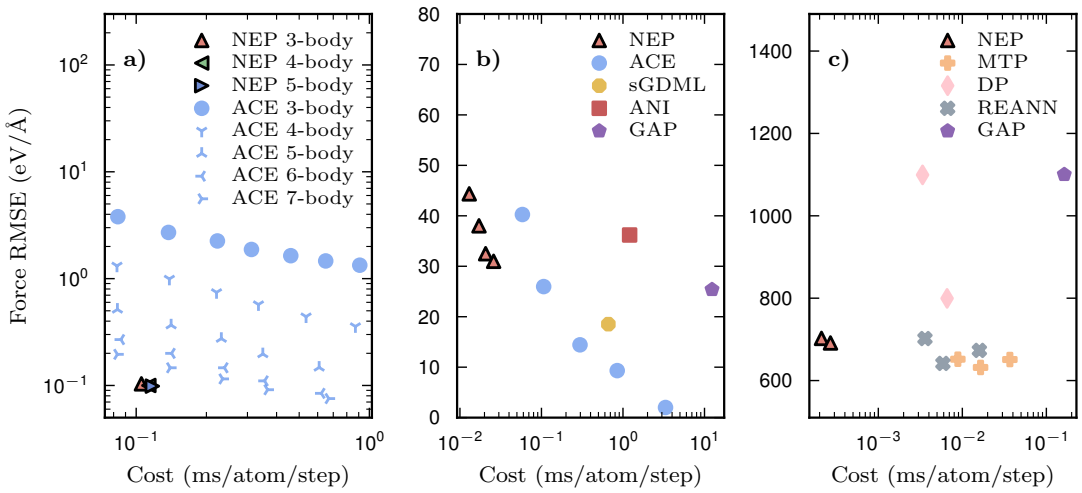


Figure 4.5: Benchmarks from paper I comparing NEP to various other state-of-the-art models as of 2022, both when running on a central processing unit (CPU) and on a GPU. The systems under study are silicon (a), azobenzene (b) and carbon (c). Benchmarks a) and b) are performed on a CPU, and benchmark c) on a GPU. Accuracy as measured by the force RMSE is on the y-axis, with speed in ms/atom/step. In all instances, NEP reaches state-of-the-art-accuracy, whilst often being at least an order of magnitude faster. Reproduced with permission from the publisher.

The formalism and optimization procedure outlined in the two previous subsections are key for the success of the NEP approach. In paper I we benchmark NEP against several state-of-the-art methods as of 2022, including ACE [86–88], ANI [65], GAP [59], sGDML [58], MTP [89, 90], DP [63, 91, 92] and REANN [93, 94]. Fig. 4.5 panels a) and b) compare NEP models for silicon (Fig. 4.5a) and azobenzene (Fig. 4.5b) prototype systems when the simulation is run on a CPU. NEP reaches state-of-the-art accuracy for both datasets, whilst often being about an order of magnitude faster in inference speed than

the competition, as measured in ms/atom/step. The inference speed of NEP increases by another order of magnitude when inference is performed on GPUs, as illustrated for a carbon dataset in Fig. 4.5c.

These benchmarks highlight the design principles of NEP, boosting computational efficiency whilst retaining accuracy. The high computational efficiency is enabled by GPUMD being implemented in native C++/CUDA and running primarily on GPUs, without almost any external dependencies. GPUMD is controlled via a text-based interface using input files. The NEP models are also defined in a proprietary text-based format. In paper II we present the CALORINE package, a Python toolbox for GPUMD simulations and NEP construction. CALORINE aims to simplify the user experience as well as make developed NEP models transferable to other workflows implemented in Python, in order to make the NEP approach easily accessible for the broader materials research community.

4.4 Takeaways

In this chapter we have familiarized ourselves with the world of machine-learned force fields (ML-FFs), machine learning models that enable large-scale MD simulations with the accuracy of computationally much more expensive quantum mechanical methods, such as DFT. We have specifically focused on the NEP approach, which provides very computationally efficient ML-FFs with state-of-the-art accuracy. This is exactly what we requested at the end of the last chapter (Chapter 3) as a necessity for running MD simulations with the accuracy required to predict neutron scattering experiments described in Chapter 2.

5

Summary of papers

Paper I

Structural stability and dynamics of liquid chromophore aggregates

In paper I we take a deep-dive into the structural stability of supramolecular aggregates of three perylene derivatives. By performing MD simulations using a classical FF and calculating correlations functions between neighboring molecules we study how the structure and dynamics of these aggregates change as a function of temperature. We find that the supramolecular aggregates are unstable for smaller perylene derivatives, but that a larger derivative remains ordered even for high temperatures, which we attribute to the increase in sterical forces between the bulkier molecules. Furthermore, we find all derivatives to be frozen into what we call an *artificially glassy state* for low temperatures; on the timescale of the MD simulation the molecules remain oriented in their original direction. As the temperature is increased, the speed of the reorientation increases for the smaller perylene derivatives, regardless of if the system is ordered into supramolecular aggregates or not. The larger derivative remains in the artificially installed glassy state, which we yet again attribute to the larger sterical hindrances in this system.

Paper II

GPUMD: A package for constructing accurate machine-learned potentials and performing highly efficient atomistic simulations

In paper II we present the GPUMD package, which is both a tool for creating NEPs with state-of-the-art accuracy as well as performing highly efficient MD simulations, thanks to GPUMD harnessing the power of modern GPUs. NEPs are Behler-Parinello-style neural network potentials, based on decomposing the total potential energy into per-atom contributions E_i,

$$E_i = U\left(\boldsymbol{w}, \boldsymbol{q}\right) = \sum_{\mu=1}^{N_{neu}} w_\mu^{(1)} \tanh\left(\sum_{\nu=1}^{N_{des}} w_{\mu\nu}^{(0)} q_\nu^i - b_\mu^{(0)}\right) - b^{(1)}. \tag{5.1}$$

$\boldsymbol{w}^{(0)}$ and $\boldsymbol{b}^{(0)}$ are the weight matrix from the input descriptor vector to the hidden layer. The hidden layer has a tanh activation function, and $\boldsymbol{w}^{(1)}$ and $b^{(1)}$ are weights and bias from the hidden layer to the single output node. Combining this functional form with a loss function that combines predicted energies, forces, virials and both $L1$ and $L2$-regularization, yields both a computationally efficient as well as a highly accurate implementation. This allows NEP to achieve similar or better accuracy compared to other common approaches, whilst being at least an order of magnitude faster. Additionally, we demonstrate the capabilities of NEP models in a variety of applications, including calculating lattice constants, tensile loading, quenching, and heat-capacity calculations. An active learning scheme for generating a diverse dataset based on farthest-point sampling is also presented.

Paper III

calorine: A Python package for constructing and sampling neuroevolution potential models

CALORINE is a Python toolbox that acts as an interface for users of GPUMD. The package includes convenience functions for setting up and running MD simulations, as well as training, analyzing, and modifying NEP models. In addition, CALORINE provides two ASE calculators [95]. ASE is a popular framework for atomistic modeling within the broader computational materials community, and these calculators thus make NEP models more widely accessible and interoperable with other workflows. The documentation and tutorials for CALORINE can be found at the following URL: `https://calorine.materialsmodeling.org/`.

6

Conclusions and outlook

Experiment är inte fysik

Jakub, 2024

In the beginning of this thesis we set out to answer two main research questions, namely

- To what extent can the developed simulation protocol capture the structure and dynamics of aggregates of perylene derivatives?
- How well can neutron scattering experiments be predicted using the simulation protocol?

The simulation protocol that I have presented in Chapter 3 and Chapter 4 is based on running accurate MD simulations with NEP models, and extracting from the resulting trajectory experimental observables, such as the dynamic structure factor measured in neutron scattering experiments as described in Chapter 2. Although I have yet to combine all three parts of the simulation protocol to comprehensively study a liquid chromophore system, I have demonstrated the capability of each part of the protocol individually throughout the thesis.

In Chapter 3, I presented paper I in which we use MD simulations with a classical FF to study the aggregation behavior of large systems of perylene derivatives, a type of chromophore. We found that the stability of the supramolecular aggregates are heavily dependent on temperature, but also on the strength of intermolecular interactions as demonstrated by the more bulky perylenediimide molecule remaining aggregated even at high temperatures. This clearly demonstrates the insights that can be gained from large-scale MD simulations.

> In conclusion, MD simulations with a classical FF can be used to describe the changes in structure and dynamics of large systems of perylene derivatives.

Paper II presents the NEP formalism, which I describe in detail in Chapter 4. NEP not only yields a prediction accuracy on par with state-of-the-art approaches in the field, but it does so whilst being much more computationally efficient owing to the implementation within GPUMD efficiently harnessing the powers of modern GPUs. Computational efficiency is key, as it makes the long MD simulations necessary to accurately describe the dynamics of chromophores feasible. Furthermore, thanks to the CALORINE package described in paper III, creating specialized NEP models for different types of chromophores such as perylene derivatives is straightforward. Hence, NEP models are well suited for accurately modeling large systems of liquid chromophores, and we may draw the following conclusion:

> Using an ML-FF such as NEP is expected to increase the accuracy of simulations of chromophores even further.

From these accurate MD simulations with NEPs I can then compute the dynamic structure factor as described in Chapter 2, in order to predict neutron scattering experiments for chromophore systems. However, this is still a work in progress, and we may thus only answer the second research question tentatively.

> Neutron scattering experiments should be able to be accurately predicted using the simulation protocol.

6.1 Limitations

Studying chromophore systems computationally is challenging, and although the simulation protocol described in this thesis is designed with this in mind there are still some inherent limitations to the methodology.

The main limitation is that of limited simulation length. Due to the time step in a MD simulation typically being on the order of fs, total simulation times are limited to at most 1 μs, even with a computationally efficient ML-FF such as NEP. This makes the simulation protocol inherently unable to describe certain slow dynamic process, such as the glass formation of certain chromophores as mentioned in [35] which takes place on experimental time scales, on the order of 100 s.

Another limitation is the small system size in a MD simulation compared to experiments. Even though PBCs are used to mimic an effectively infinite system, large supramolecular aggregates may simply not fit in a MD simulation.

A third limitation is the choice of training NEPs from reference data obtained from DFT. DFT only gives approximate solutions to the Schrödinger equation as described

in the beginning of Chapter 4, where crucially the choice of exchange-correlation functional affects the results. This means that the developed NEP and subsequently the results of the MD simulation are conditional on the choice of functional.

Taken together, these limitations may seem crippling to an all-encompassing computational description of realistic systems of chromophores. However, that loses track of the goal of developing the computational framework presented in this thesis. The goal of the simulation protocol is not to replace experiments, but rather complement them by aiding in interpreting the results. After all, experimental studies are the best probe we scientists have for understanding the world around us. The role of computer simulations and theory in this context is to improve the efficiency of experiments, and the insight gained from them.

6.2 Outlook

As I have already alluded to, the next step is to put the whole simulation protocol into practice by predicting the neutron scattering experiments for a chromophore system. One possible challenge that may appear is faithfully reproducing the details of the experimental setup. The full dynamic structure factor $S(\boldsymbol{q}, \omega)$ is not measured immediately; rather, the measured intensity is a function of the dynamic structure factor as well as the experimental resolution function, among other things. This resolution function is unique to each experimental setup, and will need to be reproduced in order to match the results from that particular instrument.

Another extension to the simulation protocol could be to include predictions of other experimental techniques. For example, Raman spectroscopy would be a good complement to neutron scattering, as Raman is sensitive to vibrations of higher energy such as intramolecular bonds. Predicting Raman spectroscopy requires predicting the dynamic susceptibility of a configuration of atoms. In a recent manuscript we extend the NEP formalism to predict tensorial properties, such as the susceptibility [96].

Finally, I also aim to study the structure and dynamics of chromophore systems in even greater detail. Mixtures of perylene derivatives and the glassy dynamics within would be particularly interesting to study, and would be a good fit for an application of the simulation protocol.

Acknowledgments

Now comes the part where I want to thank the people that have made this thesis possible. First of all, I would like to thank my supervisor Paul Erhart, for your support, both scientific and mental. You might just make a researcher of me yet. Second, I'd like to thank my co-supervisors Jan Swenson, Christian Müller, Sanghamitra Mukhopadhyay and Thomas Holm-Rod, as well as my scientific collaborator Adam Jackson, for their help in making sense of my simulations. I'm still hoping to absorb some of your immense knowledge through osmosis. Third, I want to thank my office-mate Erik Fransson for your patience in answering my questions on molecular dynamics and correlations functions, as well as for putting up with my excessively loud typing. Jakub Fojt, thank you for having coached me through the process, both with regards to writing the thesis and at the gym. A special thank you also goes out to each and everyone of my colleagues at the Condensed Matter and Materials Theory division at Chalmers. The sense of community and friendliness around the lunch table truly is something special. I'd also like to thank SwedNess for funding me, and the super-computing centers at Kungliga Tekniska Högskolan (PDC), Linköpings Tekniska Universitet (NSC), Uppsala (UPPMAX) and Chalmers (C3SE) for providing the computational resources I need for my research. To my family and friends, thank you for all your support and love, and a big thank you to my partner Amanda Djäknegren. You always help me pick up the pieces when I'm broken and at my wit's end. And finally, I want to thank you, the reader, for reading my thesis. I guess it must have been a real page-turner for you to make it this far.

Tack och hej,
Eric Lindgren, 2024

Bibliography

[1] L. Yu, D. Qian, S. Marina, F. A. A. Nugroho, A. Sharma, S. Hultmark, A. I. Hofmann, R. Kroon, J. Benduhn, D.-M. Smilgies, K. Vandewal, M. R. Andersson, C. Langhammer, J. Martín, F. Gao, and C. Müller, *Diffusion-Limited Crystallization: A Rationale for the Thermal Stability of Non-Fullerene Solar Cells,* ACS Applied Materials & Interfaces **11**, 21766 (2019). doi:10.1021/acsami.9b04554.

[2] A. Diacon, O. Krupka, and P. Hudhomme, *Fullerene-Perylenediimide (C60-PDI) Based Systems: An Overview and Synthesis of a Versatile Platform for Their Anchor Engineering,* Molecules **27**, 6522 (2022). doi:10.3390/molecules27196522.

[3] J. Zhang, Y. Li, J. Huang, H. Hu, G. Zhang, T. Ma, P. C. Y. Chow, H. Ade, D. Pan, and H. Yan, *Ring-Fusion of Perylene Diimide Acceptor Enabling Efficient Nonfullerene Organic Solar Cells with a Small Voltage Loss,* Journal of the American Chemical Society **139**, 16092 (2017). doi:10.1021/jacs.7b09998.

[4] A. Ghosh and T. Nakanishi, *Frontiers of Solvent-Free Functional Molecular Liquids,* Chemical Communications **53**, 10344 (2017). doi:10.1039/C7CC05883G.

[5] S. S. Babu, M. J. Hollamby, J. Aimi, H. Ozawa, A. Saeki, S. Seki, K. Kobayashi, K. Hagiwara, M. Yoshizawa, H. Möhwald, and T. Nakanishi, *Nonvolatile Liquid Anthracenes for Facile Full-Colour Luminescence Tuning at Single Blue-Light Excitation,* Nature Communications **4**, 1969 (2013). doi:10.1038/ncomms2969.

[6] B. C. Freitas-Dörr, C. O. Machado, A. C. Pinheiro, A. B. Fernandes, F. A. Dörr, E. Pinto, M. Lopes-Ferreira, M. Abdellah, J. Sá, L. C. Russo, F. L. Forti, L. C. P. Gonçalves, and E. L. Bastos, *A Metal-Free Blue Chromophore Derived from Plant Pigments,* Science Advances **6**, eaaz0421 (2020). doi:10.1126/sciadv.aaz0421.

[7] N. Kobayashi, T. Kasahara, T. Edura, J. Oshima, R. Ishimatsu, M. Tsuwaki, T. Imato, S. Shoji, and J. Mizuno, *Microfluidic White Organic Light-Emitting Diode Based on Integrated Patterns of Greenish-Blue and Yellow Solvent-Free Liquid Emitters,* Scientific Reports **5**, 14822 (2015). doi:10.1038/srep14822.

[8] L. Yao, S. Zhang, R. Wang, W. Li, F. Shen, B. Yang, and Y. Ma, *Highly Efficient Near-Infrared Organic Light-Emitting Diode Based on a Butterfly-Shaped Donor–Acceptor Chromophore with Strong Solid-State Fluorescence and a Large Proportion of Radiative Excitons,* Angewandte Chemie **126**, 2151 (2014). doi:10.1002/ange.201308486.

[9] V. Venunath Patil, K. Hyung Lee, and J. Yeob Lee, *Isomeric Fused Benzocarbazole as a Chromophore for Blue Fluorescent Organic Light-Emitting Diodes,* Journal of Materials Chemistry C **8**, 8320 (2020). doi:10.1039/D0TC01268H.

[10] S. Wang, T.-P. Ruoko, G. Wang, S. Riera-Galindo, S. Hultmark, Y. Puttisong, F. Moro, H. Yan, W. M. Chen, M. Berggren, C. Müller, and S. Fabiano, *Sequential Doping of Ladder-Type Conjugated Polymers for Thermally Stable n-Type Organic Conductors,* ACS Applied Materials & Interfaces **12**, 53003 (2020). doi:10.1021/acsami.0c16254.

[11] M. E. Gemayel, K. Börjesson, M. Herder, D. T. Duong, J. A. Hutchison, C. Ruzié, G. Schweicher, A. Salleo, Y. Geerts, S. Hecht, E. Orgiu, and P. Samorì, *Optically Switchable Transistors by Simple Incorporation of Photochromic Systems into Small-Molecule Semiconducting Matrices,* Nature Communications **6**, 6330 (2015). doi:10.1038/ncomms7330.

[12] M. Gsänger, D. Bialas, L. Huang, M. Stolte, and F. Würthner, *Organic Semiconductors Based on Dyes and Color Pigments,* Advanced Materials **28**, 3615 (2016). doi:10.1002/adma.201505440.

[13] K. Börjesson, D. Dzebo, B. Albinsson, and K. Moth-Poulsen, *Photon Upconversion Facilitated Molecular Solar Energy Storage,* Journal of Materials Chemistry A **1**, 8521 (2013). doi:10.1039/C3TA12002C.

[14] K. Stranius and K. Börjesson, *Determining the Photoisomerization Quantum Yield of Photoswitchable Molecules in Solution and in the Solid State,* Scientific Reports **7**, 41145 (2017). doi:10.1038/srep41145.

[15] M. Jevric, A. U. Petersen, M. Mansø, S. Kumar Singh, Z. Wang, A. Dreos, C. Sumby, M. B. Nielsen, K. Börjesson, P. Erhart, and K. Moth-Poulsen, *Norbornadiene-Based Photoswitches with Exceptional Combination of Solar Spectrum Match and Long-Term Energy Storage,* Chemistry – A European Journal **24**, 12767 (2018). doi:10.1002/chem.201802932.

[16] V. Gray, A. Dreos, P. Erhart, B. Albinsson, K. Moth-Poulsen, and M. Abrahamsson, *Loss Channels in Triplet–Triplet Annihilation Photon Upconversion: Importance of Annihilator Singlet and Triplet Surface Shapes,* Physical Chemistry Chemical Physics **19**, 10931 (2017). doi:10.1039/C7CP01368J.

[17] M. Quant, A. Hamrin, A. Lennartson, P. Erhart, and K. Moth-Poulsen, *Solvent Effects on the Absorption Profile, Kinetic Stability, and Photoisomerization Process of the Norbornadiene–Quadricyclanes System,* The Journal of Physical Chemistry C **123**, 7081 (2019). doi:10.1021/acs.jpcc.9b02111.

[18] G. Charalambidis, E. Georgilis, M. K. Panda, C. E. Anson, A. K. Powell, S. Doyle, D. Moss, T. Jochum, P. N. Horton, S. J. Coles, M. Linares, D. Beljonne, J.-V. Naubron, J. Conradt, H. Kalt, A. Mitraki, A. G. Coutsolelos, and T. S. Balaban, *A Switchable Self-Assembling and Disassembling Chiral System Based on a Porphyrin-Substituted Phenylalanine–Phenylalanine Motif,* Nature Communications **7**, 12657 (2016). doi:10.1038/ncomms12657.

[19] J. Clayden, N. Greeves, and S. G. Warren, *Organic Chemistry* (Oxford University Press, 2012). ISBN 978-0-19-166621-6.

[20] A. Laiho, B. M. Smarsly, C. F. J. Faul, and O. Ikkala, *Macroscopically Aligned Ionic Self-Assembled Perylene-Surfactant Complexes within a Polymer Matrix,* Advanced Functional Materials **18**, 1890 (2008). doi:10.1002/adfm.200701496.

[21] X. Li, L. E. Sinks, B. Rybtchinski, and M. R. Wasielewski, *Ultrafast Aggregate-to-Aggregate Energy Transfer within Self-assembled Light-Harvesting Columns of Zinc Phthalocyanine*

Tetrakis(Perylenediimide), Journal of the American Chemical Society **126**, 10810 (2004). doi: 10.1021/ja047176b.

[22] S. Herbst, B. Soberats, P. Leowanawat, M. Lehmann, and F. Würthner, *A Columnar Liquid-Crystal Phase Formed by Hydrogen-Bonded Perylene Bisimide J-Aggregates,* Angewandte Chemie International Edition **56**, 2162 (2017). doi:10.1002/anie.201612047.

[23] F. J. M. Hoeben, P. Jonkheijm, E. W. Meijer, and A. P. H. J. Schenning, *About Supramolecular Assemblies of π-Conjugated Systems,* Chemical Reviews **105**, 1491 (2005). doi:10.1021/cr030070z.

[24] P. J. Collings, E. J. Gibbs, T. E. Starr, O. Vafek, C. Yee, L. A. Pomerance, and R. F. Pasternack, *Resonance Light Scattering and Its Application in Determining the Size, Shape, and Aggregation Number for Supramolecular Assemblies of Chromophores,* The Journal of Physical Chemistry B **103**, 8474 (1999). doi:10.1021/jp991610s.

[25] Y. Deng, W. Yuan, Z. Jia, and G. Liu, *H- and J-Aggregation of Fluorene-Based Chromophores,* The Journal of Physical Chemistry B **118**, 14536 (2014). doi:10.1021/jp510520m.

[26] Z. Chen, A. Lohr, C. R. Saha-Möller, and F. Würthner, *Self-Assembled π-Stacks of Functional Dyes in Solution: Structural and Thermodynamic Features,* Chemical Society Reviews **38**, 564 (2009). doi:10.1039/B809359H.

[27] S. Chen, P. Slattum, C. Wang, and L. Zang, *Self-Assembly of Perylene Imide Molecules into 1D Nanostructures: Methods, Morphologies, and Applications,* Chemical Reviews **115**, 11967 (2015). doi:10.1021/acs.chemrev.5b00312.

[28] M. M. Safont-Sempere, P. Osswald, M. Stolte, M. Grüne, M. Renz, M. Kaupp, K. Radacki, H. Braunschweig, and F. Würthner, *Impact of Molecular Flexibility on Binding Strength and Self-Sorting of Chiral π-Surfaces,* Journal of the American Chemical Society **133**, 9580 (2011). doi: 10.1021/ja202696d.

[29] F. Lu, T. Takaya, K. Iwata, I. Kawamura, A. Saeki, M. Ishii, K. Nagura, and T. Nakanishi, *A Guide to Design Functional Molecular Liquids with Tailorable Properties Using Pyrene-Fluorescence as a Probe,* Scientific Reports **7**, 3416 (2017). doi:10.1038/s41598-017-03584-1.

[30] K. Kushwaha, L. Yu, K. Stranius, S. K. Singh, S. Hultmark, M. N. Iqbal, L. Eriksson, E. Johnston, P. Erhart, C. Müller, and K. Börjesson, *A Record Chromophore Density in High-Entropy Liquids of Two Low-Melting Perylenes: A New Strategy for Liquid Chromophores,* Advanced Science **6**, 1801650 (2019). doi:10.1002/advs.201801650.

[31] C. Ye, V. Gray, K. Kushwaha, S. Kumar Singh, P. Erhart, and K. Börjesson, *Optimizing Photon Upconversion by Decoupling Excimer Formation and Triplet Triplet Annihilation,* Physical Chemistry Chemical Physics **22**, 1715 (2020). doi:10.1039/C9CP06561J.

[32] S. Hultmark, S. H. K. Paleti, A. Harillo, S. Marina, F. A. A. Nugroho, Y. Liu, L. K. E. Ericsson, R. Li, J. Martín, J. Bergqvist, C. Langhammer, F. Zhang, L. Yu, M. Campoy-Quiles, E. Moons, D. Baran, and C. Müller, *Suppressing Co-Crystallization of Halogenated Non-Fullerene Acceptors for Thermally Stable Ternary Solar Cells,* Advanced Functional Materials **30**, 2005462 (2020). doi:10.1002/adfm.202005462.

[33] S. H. K. Paleti, S. Hultmark, J. Han, Y. Wen, H. Xu, S. Chen, E. Järsvall, I. Jalan, D. R. Villalva, A. Sharma, J. I. Khan, E. Moons, R. Li, L. Yu, J. Gorenflot, F. Laquai, C. Müller, and

D. Baran, *Hexanary Blends: A Strategy towards Thermally Stable Organic Photovoltaics,* Nature Communications **14**, 4608 (2023). doi:10.1038/s41467-023-39830-6.

[34] M.-Y. Ni, S.-F. Leng, H. Liu, Y.-K. Yang, Q.-H. Li, C.-Q. Sheng, X. Lu, F. Liu, and J.-H. Wan, *Ternary Organic Solar Cells with 16.88% Efficiency Enabled by a Twisted Perylene Diimide Derivative to Enhance the Open-Circuit Voltage,* Journal of Materials Chemistry C **9**, 3826 (2021). doi:10.1039/D0TC05691J.

[35] S. Hultmark, A. Cravcenco, K. Kushwaha, S. Mallick, P. Erhart, K. Börjesson, and C. Müller, *Vitrification of Octonary Perylene Mixtures with Ultralow Fragility,* Science Advances **7**, eabi4659 (2021). doi:10.1126/sciadv.abi4659.

[36] J. Brede, M. Linares, S. Kuck, J. Schwöbel, A. Scarfato, S.-H. Chang, G. Hoffmann, R. Wiesendanger, R. Lensen, P. H. J. Kouwer, J. Hoogboom, A. E. Rowan, M. Bröring, M. Funk, S. Stafström, F. Zerbetto, and R. Lazzaroni, *Dynamics of Molecular Self-Ordering in Tetraphenyl Porphyrin Monolayers on Metallic Substrates,* Nanotechnology **20**, 275602 (2009). doi:10.1088/0957-4484/20/27/275602.

[37] P. Iavicoli, H. Xu, L. N. Feldborg, M. Linares, M. Paradinas, S. Stafström, C. Ocal, B. Nieto-Ortega, J. Casado, J. T. López Navarrete, R. Lazzaroni, S. D. Feyter, and D. B. Amabilino, *Tuning the Supramolecular Chirality of One- and Two-Dimensional Aggregates with the Number of Stereogenic Centers in the Component Porphyrins,* Journal of the American Chemical Society **132**, 9350 (2010). doi:10.1021/ja101533j.

[38] A. Beigbeder, M. Linares, M. Devalckenaere, P. Degée, M. Claes, D. Beljonne, R. Lazzaroni, and P. Dubois, *CH-π Interactions as the Driving Force for Silicone-Based Nanocomposites with Exceptional Properties,* Advanced Materials **20**, 1003 (2008). doi:10.1002/adma.200701497.

[39] T. Shikata, T. Nishida, B. Isare, M. Linares, R. Lazzaroni, and L. Bouteiller, *Structure and Dynamics of a Bisurea-Based Supramolecular Polymer in n-Dodecane,* The Journal of Physical Chemistry B **112**, 8459 (2008). doi:10.1021/jp800495v.

[40] J. Sjöqvist, J. Maria, R. A. Simon, M. Linares, P. Norman, K. P. R. Nilsson, and M. Lindgren, *Toward a Molecular Understanding of the Detection of Amyloid Proteins with Flexible Conjugated Oligothiophenes,* The Journal of Physical Chemistry A **118**, 9820 (2014). doi:10.1021/jp506797j.

[41] G. L. Squires, *Introduction to the Theory of Thermal Neutron Scattering* (Cambridge: Cambridge University Press, 2012). ISBN 978-1-107-64406-9. doi:10.1017/CBO9781139107808.

[42] M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Oxford University Press, 1987).

[43] W. C. Swope, H. C. Andersen, P. H. Berens, and K. R. Wilson, *A Computer Simulation Method for the Calculation of Equilibrium Constants for the Formation of Physical Clusters of Molecules: Application to Small Water Clusters,* The Journal of Chemical Physics **76**, 637 (1982). doi:10.1063/1.442716.

[44] R. Bowley, M. Sanchez, R. Bowley, and M. Sanchez, *Introductory Statistical Mechanics* (Oxford, New York: Oxford University Press, 1999). ISBN 978-0-19-850576-1.

[45] G. Bussi, D. Donadio, and M. Parrinello, *Canonical Sampling through Velocity Rescaling,* The Journal of Chemical Physics **126**, 014101 (2007). doi:10.1063/1.2408420.

[46] M. Parrinello and A. Rahman, *Crystal Structure and Pair Potentials: A Molecular-Dynamics Study,* Physical Review Letters **45**, 1196 (1980). doi:10.1103/PhysRevLett.45.1196.

[47] C. Müller, *On the Glass Transition of Polymer Semiconductors and Its Impact on Polymer Solar Cell Stability,* Chemistry of Materials **27**, 2740 (2015). doi:10.1021/acs.chemmater.5b00024.

[48] R. M. Martin, *Electronic Structure: Basic Theory and Practical Methods* (Cambridge: Cambridge University Press, 2004). doi:10.1017/CBO9780511805769.

[49] C. A. Ullrich, *Time-Dependent Density-Functional Theory: Concepts and Applications* (Oxford University Press, 2011). ISBN 978-0-19-177513-0. doi:10.1093/acprof:oso/9780199563029.001.0001.

[50] M. Born and R. Oppenheimer, *Zur Quantentheorie Der Molekeln,* Annalen der Physik **389**, 457 (1927). doi:10.1002/andp.19273892002.

[51] W. Kohn and L. J. Sham, *Self-Consistent Equations Including Exchange and Correlation Effects,* Physical Review **140**, A1133 (1965). doi:10.1103/PhysRev.140.A1133.

[52] O. T. Unke, S. Chmiela, H. E. Sauceda, M. Gastegger, I. Poltavsky, K. T. Schütt, A. Tkatchenko, and K.-R. Müller, *Machine Learning Force Fields,* Chemical Reviews **121**, 10142 (2021). doi:10.1021/acs.chemrev.0c01111.

[53] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning* (Cambridge, Mass: MIT Press, 2006). ISBN 978-0-262-18253-9.

[54] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, *Active Learning with Statistical Models,* Journal of Artificial Intelligence Research **4**, 129 (1996). doi:10.1613/jair.295.

[55] H. Liu, Y.-S. Ong, and J. Cai, *A Survey of Adaptive Sampling for Global Metamodeling in Support of Simulation-Based Complex Engineering Design,* Structural and Multidisciplinary Optimization **57**, 393 (2018). doi:10.1007/s00158-017-1739-8.

[56] E. Gilboa, Y. Saatçi, and J. P. Cunningham, *Scaling Multidimensional Inference for Structured Gaussian Processes,* IEEE Transactions on Pattern Analysis and Machine Intelligence **37**, 424 (2015). doi:10.1109/TPAMI.2013.192.

[57] S. Chmiela, A. Tkatchenko, H. E. Sauceda, I. Poltavsky, K. T. Schütt, and K.-R. Müller, *Machine Learning of Accurate Energy-Conserving Molecular Force Fields,* Science Advances **3**, e1603015 (2017). doi:10.1126/sciadv.1603015.

[58] S. Chmiela, H. E. Sauceda, K.-R. Müller, and A. Tkatchenko, *Towards Exact Molecular Dynamics Simulations with Machine-Learned Force Fields,* Nature Communications **9**, 3887 (2018). doi:10.1038/s41467-018-06169-2.

[59] V. L. Deringer, A. P. Bartók, N. Bernstein, D. M. Wilkins, M. Ceriotti, and G. Csányi, *Gaussian Process Regression for Materials and Molecules,* Chemical Reviews **121**, 10073 (2021). doi:10.1021/acs.chemrev.1c00022.

[60] J. Behler and M. Parrinello, *Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces,* Physical Review Letters **98**, 146401 (2007). doi:10.1103/PhysRevLett.98.146401.

[61] J. Behler, *Atom-Centered Symmetry Functions for Constructing High-Dimensional Neural Network Potentials,* The Journal of Chemical Physics **134**, 074106 (2011). doi:10.1063/1.3553717.

[62] M. Gastegger, J. Behler, and P. Marquetand, *Machine Learning Molecular Dynamics for the Simulation of Infrared Spectra,* Chemical Science **8**, 6924 (2017). doi:10.1039/C7SC02267K.

[63] H. Wang, L. Zhang, J. Han, and W. E, *DeePMD-kit: A Deep Learning Package for Many-Body Potential Energy Representation and Molecular Dynamics,* Computer Physics Communications **228**, 178 (2018). doi:10.1016/j.cpc.2018.03.016.

[64] Y. Zhang, C. Hu, and B. Jiang, *Embedded Atom Neural Network Potentials: Efficient and Accurate Machine Learning with a Physically Inspired Representation,* The Journal of Physical Chemistry Letters **10**, 4962 (2019). doi:10.1021/acs.jpclett.9b02037.

[65] J. S. Smith, O. Isayev, and A. E. Roitberg, *ANI-1: An Extensible Neural Network Potential with DFT Accuracy at Force Field Computational Cost,* Chemical Science **8**, 3192 (2017). doi:10.1039/C6SC05720A.

[66] K. T. Schütt, H. E. Sauceda, P.-J. Kindermans, A. Tkatchenko, and K.-R. Müller, *SchNet – A Deep Learning Architecture for Molecules and Materials,* The Journal of Chemical Physics **148**, 241722 (2018). doi:10.1063/1.5019779.

[67] J. Behler, *Constructing High-Dimensional Neural Network Potentials: A Tutorial Review,* International Journal of Quantum Chemistry **115**, 1032 (2015). doi:10.1002/qua.24890.

[68] I. Batatia, D. P. Kovacs, G. Simm, C. Ortner, and G. Csanyi, *MACE: Higher Order Equivariant Message Passing Neural Networks for Fast and Accurate Force Fields,* Advances in Neural Information Processing Systems **35**, 11423 (2022).

[69] M. Thürlemann, L. Böselt, and S. Riniker, *Learning Atomic Multipoles: Prediction of the Electrostatic Potential with Equivariant Graph Neural Networks,* Journal of Chemical Theory and Computation **18**, 1701 (2022). doi:10.1021/acs.jctc.1c01021.

[70] V. G. Satorras, E. Hoogeboom, and M. Welling, *E(n) Equivariant Graph Neural Networks,* 2022. doi:10.48550/arXiv.2102.09844.

[71] J. Behler, *Four Generations of High-Dimensional Neural Network Potentials,* Chemical Reviews **121**, 10037 (2021). doi:10.1021/acs.chemrev.0c00868.

[72] J. Liu, Z. Shen, Y. He, X. Zhang, R. Xu, H. Yu, and P. Cui, *Towards Out-Of-Distribution Generalization: A Survey,* 2023. doi:10.48550/arXiv.2108.13624.

[73] J.-Y. Yao, K.-P. Ning, Z.-H. Liu, M.-N. Ning, and L. Yuan, *LLM Lies: Hallucinations Are Not Bugs, but Features as Adversarial Examples,* 2023. doi:10.48550/arXiv.2310.01469.

[74] H. Alkaissi and S. I. McFarlane, *Artificial Hallucinations in ChatGPT: Implications in Scientific Writing,* Cureus **15**, e35179 (2023). doi:10.7759/cureus.35179.

[75] K. C. Siontis, Z. I. Attia, S. J. Asirvatham, and P. A. Friedman, *ChatGPT Hallucinating: Can It Get Any More Humanlike?,* European Heart Journal **45**, 321 (2024). doi:10.1093/eurheartj/ehad766.

[76] M. Karabin and D. Perez, *An Entropy-Maximization Approach to Automated Training Set Generation for Interatomic Potentials,* The Journal of Chemical Physics **153**, 094110 (2020). doi:10.1063/5.0013059.

[77] A. P. Bartók, R. Kondor, and G. Csányi, *On Representing Chemical Environments,* Physical Review B **87**, 184115 (2013). doi:10.1103/PhysRevB.87.184115.

[78] S. Jindal, S. Chiriki, and S. S. Bulusu, *Spherical Harmonics Based Descriptor for Neural Network Potentials: Structure and Dynamics of Au147 Nanocluster,* The Journal of Chemical Physics **146**, 204301 (2017). doi:10.1063/1.4983392.

[79] H. Huo and M. Rupp, *Unified Representation of Molecules and Crystals for Machine Learning*, Machine Learning: Science and Technology **3**, 045017 (2022). doi:10.1088/2632-2153/aca005.

[80] K. Song, R. Zhao, J. Liu, Y. Wang, E. Lindgren, Y. Wang, S. Chen, K. Xu, T. Liang, P. Ying, N. Xu, Z. Zhao, J. Shi, J. Wang, S. Lyu, Z. Zeng, S. Liang, H. Dong, L. Sun, Y. Chen, Z. Zhang, W. Guo, P. Qian, J. Sun, P. Erhart, T. Ala-Nissila, Y. Su, and Z. Fan, *General-Purpose Machine-Learned Potential for 16 Elemental Metals and Their Alloys*, 2023. doi:10.48550/arXiv.2311.04732.

[81] H. Zou and T. Hastie, *Regularization and Variable Selection Via the Elastic Net*, Journal of the Royal Statistical Society Series B: Statistical Methodology **67**, 301 (2005). doi:10.1111/j.1467-9868.2005.00503.x.

[82] Y. Sun, F. Gomez, T. Schaul, and J. Schmidhuber, *A Linear Time Natural Evolution Strategy for Non-Separable Functions*, 2011. doi:10.48550/arXiv.1106.1998.

[83] Z. Fan, Z. Zeng, C. Zhang, Y. Wang, K. Song, H. Dong, Y. Chen, and T. Ala-Nissila, *Neuroevolution Machine Learning Potentials: Combining High Accuracy and Low Cost in Atomistic Simulations and Application to Heat Transport*, Physical Review B **104**, 104309 (2021). doi:10.1103/PhysRevB.104.104309.

[84] T. Glasmachers, T. Schaul, S. Yi, D. Wierstra, and J. Schmidhuber, *Exponential Natural Evolution Strategies*, in *Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation - GECCO '10*, (Portland, Oregon, USA), 393, ACM Press, 2010. doi:10.1145/1830483.1830557.

[85] L. Breiman, *Bagging Predictors*, Machine Learning **24**, 123 (1996). doi:10.1007/BF00058655.

[86] R. Drautz, *Atomic Cluster Expansion for Accurate and Transferable Interatomic Potentials*, Physical Review B **99**, 014104 (2019). doi:10.1103/PhysRevB.99.014104.

[87] D. P. Kovács, C. van der Oord, J. Kucera, A. E. A. Allen, D. J. Cole, C. Ortner, and G. Csányi, *Linear Atomic Cluster Expansion Force Fields for Organic Molecules: Beyond RMSE*, Journal of Chemical Theory and Computation **17**, 7696 (2021). doi:10.1021/acs.jctc.1c00647.

[88] G. Dusson, M. Bachmayr, G. Csányi, R. Drautz, S. Etter, C. van der Oord, and C. Ortner, *Atomic Cluster Expansion: Completeness, Efficiency and Stability*, Journal of Computational Physics **454**, 110946 (2022). doi:10.1016/j.jcp.2022.110946.

[89] A. V. Shapeev, *Moment Tensor Potentials: A Class of Systematically Improvable Interatomic Potentials*, Multiscale Modeling & Simulation **14**, 1153 (2016). doi:10.1137/15M1054183.

[90] I. S. Novikov, K. Gubaev, E. V. Podryabinkin, and A. V. Shapeev, *The MLIP Package: Moment Tensor Potentials with MPI and Active Learning*, Machine Learning: Science and Technology **2**, 025002 (2020). doi:10.1088/2632-2153/abc9fe.

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

[92] L. Zhang, J. Han, H. Wang, W. Saidi, R. Car, and W. E, *End-to-End Symmetry Preserving Interatomic Potential Energy Model for Finite and Extended Systems*, in *Advances in Neural Information Processing Systems*, vol. 31, Curran Associates, Inc., 2018.

[93] Y. Zhang, J. Xia, and B. Jiang, *REANN: A PyTorch-based End-to-End Multi-Functional Deep Neural Network Package for Molecular, Reactive, and Periodic Systems,* The Journal of Chemical Physics **156**, 114801 (2022). doi:10.1063/5.0080766.

[94] Y. Zhang, J. Xia, and B. Jiang, *Physically Motivated Recursively Embedded Atom Neural Networks: Incorporating Local Completeness and Nonlocality,* Physical Review Letters **127**, 156002 (2021). doi:10.1103/PhysRevLett.127.156002.

[95] A. H. Larsen, J. J. Mortensen, J. Blomqvist, I. E. Castelli, R. Christensen, M. Dułak, J. Friis, M. N. Groves, B. Hammer, C. Hargus, E. D. Hermes, P. C. Jennings, P. B. Jensen, J. Kermode, J. R. Kitchin, E. L. Kolsbjerg, J. Kubal, K. Kaasbjerg, S. Lysgaard, J. B. Maronsson, T. Maxson, T. Olsen, L. Pastewka, A. Peterson, C. Rostgaard, J. Schiøtz, O. Schütt, M. Strange, K. S. Thygesen, T. Vegge, L. Vilhelmsen, M. Walter, Z. Zeng, and K. W. Jacobsen, *The Atomic Simulation Environment—a Python Library for Working with Atoms,* Journal of Physics: Condensed Matter **29**, 273002 (2017). doi:10.1088/1361-648X/aa680e.

[96] N. Xu, P. Rosander, C. Schäfer, E. Lindgren, N. Österbacka, M. Fang, W. Chen, Y. He, Z. Fan, and P. Erhart, *Tensorial properties via the neuroevolution potential framework: Fast simulation of infrared and Raman spectra,* 2023.