Development of Algorithms for Classical and Semiclassical Dynamics

Michele Gandolfi

This dissertation is submitted for the degree of
Doctor of Philosophy
at the Chemistry Department of
Università degli Studi di Milano
Milano, Italy

February 2024

Acknowledgements

I extend my heartfelt gratitude to prof. Michele Ceotto, for his insightful guidance and constant encouragement throughout these challenging years. For his scientific sensibility, creative ideas, and profound humanity, he has been a model for the scientist I aspire to become. I express my sincere appreciation to all members, both present and past, of the Ceotto group. To Riccardo, Chiara, Marco, Fabio, Alessandro, Giacomo B., Giacomo M., Davide M., Cecilia, Davide B., Luca, and Paolo. You all have created a fantastic work environment, filled with memorable moments and professional growth.

I extend my deepest thanks to my friends and loved ones, for their unwavering support during the times it was needed the most. To my family - my mother, father, and sister - your ceaseless encouragement and support have been my main source of strength. *Grazie, siete i migliori.*

I gratefully acknowledge fundings from the European Research Council (Grant Agreement No. 647107—SEMICOMPLEX-ERC-2014-CoG and No. 101081361—SEMISOFT-ERC-2022-POC2) under the European Union's Horizon 2020 research and innovation programme, and from the Italian Ministry of Education, University, and Research (MIUR) (FARE programme R16KN7XBRB-project QURE).

Declaration

I confirm that the work described in this PhD dissertation was done entirely and solely by myself. All sources of information are included in the bibliography section.
Some of the work presented in this thesis has been published in the following peer-reviewed papers:

- M. Gandolfi, A. Rognoni, C. Aieta, R. Conte, and M. Ceotto, Machine learning for vibrational spectroscopy via divide-and-conquer semiclassical initial value representation molecular dynamics with application to N-methylacetamide, *J. Chem. Phys.*, **153**, 204104 (2020).
- M. Gandolfi, and M. Ceotto. Unsupervised machine learning neural gas algorithm for accurate evaluations of the Hessian matrix in molecular dynamics, *J. Chem. Theory Comp.* **17** 6733-6746 (2021).
- R. Conte, C. Aieta, G. Botti, M. Cazzaniga, M. Gandolfi, C. Lanzi, G. Mandelli, D. Moscato, and M. Ceotto, Anharmonicity and quantum nuclear effects in theoretical vibrational spectroscopy: a molecular tale of two cities. *Theor. Chem. Acc.* **142**, (2023).
- M. Gandolfi, and M. Michele Ceotto, Molecular Dynamics of Artificially Pair-Decoupled Systems: An Accurate Tool for Investigating the Importance of Intramolecular Couplings, *J. Chem. Theory Comp.* **19**, 6093–6108, (2023).

Contents

[1]This section is a rework of selected results from the article **R. Conte**, **C. Aieta**, **G. Botti**, **M. Cazzaniga**, **M. Gandolfi**, **C. Lanzi**, **G. Mandelli**, **D. Moscato**, and **M. Ceotto**, Anharmonicity and nuclear quantum effects in theoretical vibrational spectroscopy: a molecular tale of two cities. *Theor. Chem. Acc.*

[2]This chapter is a selection with minor modifications of the contents of the paper **Michele Gandolfi**, **Alessandro Rognoni**, **Chiara Aieta**, **Riccardo Conte**, and **Michele Ceotto**, "Machine learning for vibrational spectroscopy via divide-and-conquer semiclassical initial value representation molecular dynamics with application to N-methylacetamide"

[3]This chapter is a selection with minor modifications of the contents of the paper **Michele Gandolfi**, and **Michele Ceotto**, "An Unsupervised Machine Learning Neural Gas Algorithm for Accurate Evaluations of the Hessian matrix in Molecular Dynamics"

[4]This chapter is a selection with minor modifications of the content of the paper **Michele Gandolfi**, and **Michele Ceotto**, "Molecular Dynamics of Artificially Pair-Decoupled Systems: An Accurate Tool for Investigating the Importance of Intramolecular Couplings"

[5]This appendix is a reproduction with minor modifications of the content of the supporting information for the paper **Michele Gandolfi**, and **Michele Ceotto**, "Molecular Dynamics of Artificially Pair-Decoupled Systems: An Accurate Tool for Investigating the Importance of Intramolecular Couplings"

Chapter 1

Introduction

The work described in this dissertation is primarily focused on the development of new algorithms aimed at enhancing the comprehension of dynamical (time-dependent) interactions among the atoms in vibrating chemical systems. The allure of chemical science lies in the vast and uncharted complexity of weak and transient interactions that occur among the atoms. Take for example a system composed only by a small number of water molecules, that is, a small water cluster. Such a system exhibits a concerted "geared" motion of the water molecules that is governed by a complicated fluxional network of hydrogen bonds.[1–3] The motion of each water molecule must be cooperative and continuously well synchronized with the motion of all the others to keep the forces balanced and the motion harmonious. In case a portion of the cluster were to move asynchronously from the rest, then the chemistry of water clusters (and possibly of liquid water itself!) would be completely different. The same arguments apply to any other chemical system, with the added complexity that a larger variety of atoms are involved. In fact, when one aims to find a molecule that exhibits some desirable property (such as a compound effective in binding a specific protein), given a hint of the structural form of the molecule, the number of possible combinations and configurations of atoms is simply intimidating, and increases exponentially fast with the number of atoms. While the *chemical compound space* concept[4–6] was introduced to express this combinatorial complexity, it is really the amount of weak atom-atom interactions and dynamical couplings that makes the compound space so complicated, yet worth exploring. Just like a water cluster requires a correlated motion of its components to display its concerted geared motion, also organic molecules must display a well synchronized motion in each of their pieces. A lack of synchronization may imply structural changes and a fast redistribution of the energy within the molecule (as we will see in Chapter 6). If it happens that a portion of a molecule vibrates asynchronously, without disrupting the synchronized motion of the rest, then we have a truly separable system and the compound space could split into two smaller compound spaces, to be

explored independently. This separable situation is clearly a highly desirable property! But to asses whether or not a system is separable, we really need to investigate the vibrational couplings between the nuclear degrees of freedom with a time dependent approach.

Many computational methods that attempt to study large molecules are based on the sensible approximation that the interactions among the nuclei's degrees of freedom (as well as their dynamical couplings) are negligible or have a predefined (computationally convenient) mathematical expression.[7–10] In other cases the couplings among degrees of freedom are explicitly recovered by an expensive summation over many uncoupled configurations.[11] Failure to capture the relationships that link the potential with all the nuclear coordinates means looking at an artificially simplified version of the system, whose dynamics will be different from that of the real system. As far as we know, there is no obvious and rigorous way to determine the importance of a specific coupling for the long time dynamics of a molecule. Methods based on instantaneous normal mode analysis[12, 13] and the investigation of correlations functions[14, 15] can capture valuable information about the dynamical couplings and the intramolecular energy redistribution, but they do not answer questions of the kind: "will this process still occur if this coupling were a little bit weaker?" This basic question motivated most of the research detailed in this thesis.

The main instrument that we choose to carry out the investigation of dynamical couplings is computational vibrational spectroscopy. Molecular vibrations occur on a femtosecond timescale, and can be studied in the region of the light spectrum that spans between 10 to about 10000 cm^{-1}. Thus, the main experimental techniques employed to investigate this region of the spectrum are infrared (IR) and Raman spectroscopies. In IR and Raman spectroscopies the frequencies of the vibrational spectra are related to the structure, in particular to the presence of specific functional groups, which exhibit characteristic bands. Intramolecular couplings and weak interactions may influence the whole spectrum by shifting the signals from their expected frequencies. However, it is in the fingerprint region of the spectrum, encompassing the frequency range between 600 $cm^{-1}$and 1400 cm^{-1}, that the impact of intramolecular couplings becomes particularly relevant. Within this interval, the frequency signals exhibit notable variations in response to even minor structural changes.

In the dynamical approach to spectroscopy, which is extensively used in this work, one computes the vibrational spectrum as the Fourier transform of relevant correlation functions, such as the survival probability amplitude and the dipole-dipole correlation function. It turns out that in certain cases it is more computationally convenient to solve the time-dependent Schrödinger equation over the time independent one,[11] for the simple reason that propagating a wavepacket corresponds to solving

an initial value problem, which is mathematically simpler than solving the eigenvalue problem of the time-independent Schrödinger equation.[11, 16] A prominent exact quantum dynamical method for vibrational spectroscopy is the Multi-Configuration Time-Dependent-Hartree (MCTDH) method.[11] MCTDH uses combinations of many hartree products as the wavefunction *ansatz*. The equations of motion for the basis functions coefficients are obtained by applying the (time-dependent) Dirac-Frenkel variational principle to the *ansatz* and solving the equations of motion numerically. The presence of a sum over many configurations in the wavefunction *ansatz* allows recovering the instantaneous correlations among degrees of freedom. When some degrees of freedom can be assumed to be almost uncoupled there is no need to use many configurations to describe that specific coupling. Thus, in such cases the wavefunction is more compact and the MCTDH algorithm is more efficient. Conversely, if all the degrees of freedom are coupled with one another, the MCTDH algorithm does not provide any gain in terms of computational efficiency. The Multi-Layer (ML) version of MCTDH further extends the reach of the MCTDH algorithm to systems of 24 atoms and larger,[17] by writing the equations of motion in a hierarchical way, where the indexes of many degrees of freedom are grouped into collective indexes and represented by a single particle function. Another efficient and numerically exact quantum algorithm to describe vibrations is the vibrational version of the Density Matrix Renormalization Group (vDMRG) method, which can be used in both its time-independent and time-dependent versions [18] and can be applied to the ML-MCTDH state function *ansatz*. [19] Just like MCTDH, also DMRG is hampered by strongly coupled degrees of freedom, requiring a variational treatment of the low frequency, anharmonic modes. [18] Both these methods, as well as methods based on quantum trajectories on approximate quantum potentials,[20–22] strongly rely on choosing a representation of the coordinates,[23, 24] with the objective of minimizing their couplings, so that the wavefunction is expanded on a minimal basis.[25]

Classical mechanical methods for vibrational spectroscopy are all based on the inaccurate assumption that the nuclei obey Newton's laws of motion. This assumption is indeed incorrect, but often it represents a good enough approximation for the study of molecules and solids. Often one resorts to use of classical mechanics because of computational convenience. In fact, simulations of several hundreds atoms is feasible for classical mechanics, as long as a reliable potential energy surface is available. Thus, in general, ease of use and clear interpretation might be preferred over high accuracy, when the latter is not crucial. However, studying vibrational spectroscopy with classical mechanics is limited to investigating the fundamental vibrations of the atoms in a potential well, which, in turn, depends on the curvature of the Born-Oppenheimer Potential Energy Surface (BO-PES). In classical mechanics the energy is not quantized

and all the quantum mechanical concepts that arise from the quantum picture, such as overtones, combination bands, populations, etc. cannot, at any degree, be captured by a classical mechanics calculation. Even though the classical picture of vibrations is much simpler than the quantum one, accurate prediction of the classical vibrational frequencies remain challenging, because they depend on the (non-local) anharmonic shape of the BO-PES. One may resort to the harmonic approximation, which, however, is a crude representation of the real system and may lead to a large disagreement with experimental measurements. Various *ad hoc* weighting schemes may be employed to scale the harmonic frequencies to mitigate this disagreement.[26] The main disadvantage of scaling the harmonic frequencies is that the optimal scaling factors depend on the system, level of theory/basis, and on the type of motion. More consistent results can be obtained by a sensible, classical dynamic simulation. The Quasi-Classical-Trajectory method (QCT)[27] samples initial conditions from a system of quantized harmonic oscillators and then evolves a swarm of independent trajectories, thus accounting for anharmonicity.[28] Path Integral Molecular Dynamics (PIMD) methods, such as Ring Polymer Molecular Dynamics (RPMD) [29, 30] and Centroid Molecular Dynamics (CMD), [31] along with their many variants, are other popular approaches for vibrational spectroscopy. They have the advantage of a convenient computational cost (still greater than traditional molecular dynamics), but each has weaknesses and caveats that must be accounted for.[32]

A practical approach to combine the advantages of classical mechanics with a quantum mechanical calculation is the SemiClassical (SC) approximation. Semiclassical methods are based on a stationary phase approximation [33] (equivalent to the imaginary time Steepest Descent integration method) of the quantum mechanical propagator in real time. The action functionals of the most contributing (classical) pathways are expanded to second order and integrated analytically. This corresponds to approximating the propagator integral to a sum over those classical paths which join the initial and final states. A variety of methodologies that implement semiclassical theories [34–40] have been proven useful in applications of vibrational spectroscopy of isolated molecules, clusters, molecules in solution and at interfaces. [41–48] In particular the Divide-and-Conquer (DC) method, recently proposed by Ceotto, Di Liberto, and Conte,[49] allows to investigate the vibrational features of complicated molecules, at a semiclassical level, by a projection of the Herman-Kluk propagator variables to lower dimensional subspaces. In this thesis we present some methodological advancements for semiclassical methods, particularly for the Divide-and-Conquer (DC) method.[49–51]

The purpose of the work presented in this thesis is dual and goes beyond the study of vibrational spectroscopy with computational methods. Firstly, we aim to improve and challenge those techniques that rely on uncoupled description of the nuclei's motion.

Secondly, we aim to give instruments to measure the dynamical effects of vibrational couplings. We demonstrate on both model and realistic systems how the new ideas and methods we developed can be used to get (often) unexpected physical and chemical insight into middle-sized organic molecules.

Chapter 2

Classical and Semiclassical Vibrational Spectroscopy

To assess the vibrational behavior of a molecule one can employ the harmonic approximation. After obtaining the equilibrium molecular geometry, diagonalization of the mass-scaled Hessian matrix provides the harmonic frequencies of vibration. The corresponding eigenvectors define the normal modes of vibration. This procedure involves only electronic structure calculations at a particular molecular geometry and therefore it is based on a local investigation of the potential energy surface (PES). The PES, however, is not a quadratic function of the nuclear coordinates, thus the harmonic approximation breaks down as soon as the molecule moves away from the equilibrium geometry. As a consequence, the spectrum of an anharmonic system is characterized by shifted frequencies compared to its harmonic approximation. Strategies to account for anharmonicities include, for instance, scaling the harmonic frequencies by *ad hoc* tabulated coefficients. These coefficients, which depend on the specific electronic structure and basis utilized, serve a dual purpose. They aim to rectify the limitations of electronic structure methods, such as the approximate treatment of the electron electron interaction energy, and they incorporate the anharmonic contributions of the potential energy surface. To achieve this dual purpose, the coefficients are determined by an optimization process, which aims to minimize the deviations between the scaled harmonic frequencies and the experimental measurements, all within the context of the chosen electronic structure method and basis set. [26]

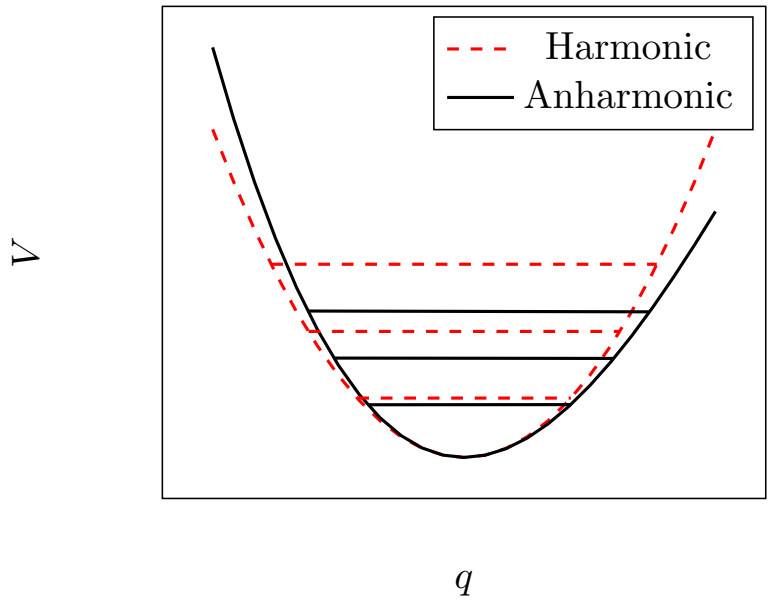


Figure 2.1: Harmonic vs anharmonic oscillator potentials with energy levels

The strategy of computing the velocity autocorrelation function is more expensive yet more reliable. It perfectly accounts for the anharmonic shape of the potential without using *ad hoc* scaling coefficients. However, in classical molecular dynamics simulations the energy is not quantized, and the spectroscopic signals do not correspond to a transition between two states, and, finally, the temperature dependence of the simulation has a different interpretation from that of a quantum mechanical system. This becomes manifest when one tries to reproduce an experiment at low temperature with a classical simulation at the same experimental temperature. As a matter of fact, independently of temperature, the molecules vibrate (and keep vibrating) at least with the Zero-Point vibrational Energy (ZPE), which is higher than the bottom of the potential well. If one simulates the classical dynamics at low temperature, it often happens that the simulation remains close to the bottom of the well and below the ZPE. If we describe the nuclei as classical particles, we can compute the power spectrum as the finite time FT of the momentum-momentum correlation function, which can be rearranged as [52]

$$I(E) = \frac{1}{2T}\left|\int_0^T e^{i\omega t} p(t)dt\right|^2 . \tag{2.1}$$

The data necessary to apply Eq. 2.1 are gathered by the time propagation of a system with initial conditions $z(0) = (q(0), p(0))$. To get the spectrum of a single molecule in the void one can run a single-trajectory calculation, using the initial conditions of the

approximate harmonic oscillators model in mass-scaled coordinates, that is

$$\begin{cases} q_i(0) = q_{i,eq} \\ p_i(0) = [(2n_i + 1)\omega_i]^{1/2}, \end{cases} \tag{2.2}$$

where $q_{i,eq}$ is the equilibrium geometry, ω_i, n_i are the harmonic frequency and quantum number of i^{th} degree of freedom. An ensemble of trajectories can also be employed. In this case, Eq. 2.1 becomes an ensemble average

$$I(E) = \iint dp_0 dq_0 \frac{1}{2T} \left| \int_0^T e^{i\omega t} p(t) dt \right|^2, \tag{2.3}$$

where the initial conditions p_0 and q_0 are sampled from the representative statistical ensemble, and the integral is computed in the Monte Carlo way. If one aims to simulate the microcanonical ensemble, the initial conditions can be sampled from the surface of the ellipse of the Hamiltonian in harmonic approximation, that is

$$\begin{cases} q_i = \quad [(2n_i + 1)/\omega_i]^{1/2} \cos(\phi_i) \\ p_i = -\left[(2n_i + 1)\ \ \omega_i\right]^{1/2} \sin(\phi_i) \end{cases}, \tag{2.4}$$

where ϕ_i is a uniformly distributed random number between 0 and 2π. This procedure is called Quasi-Classical-Trajectory (QCT)[28]. The nuclei are assumed classical, but the initial conditions are sampled from the energy shell of a collection of quantized harmonic oscillators, with quantum numbers $\{n_i\}$. This simple method has the advantage of being easy to implement and computationally as expensive as standard molecular dynamics simulations. The spectrum resulting from Eq. 2.1 describes the fundamental vibrational frequency, accounting for anharmonicity and mode couplings.

2.1 Combinations, Anharmonicity and Temperature Dependence[1]

Experiments do not give direct evidence of the zero-point energy, but they display other features that can be explained as nuclear quantum effects. For instance, combination bands appear when transitions associated to the simultaneous excitation of two or more modes occur.[3] Clearly there is no equivalence to combination bands in classical spectroscopy. Classical combinations of spectroscopic peaks can be seen at linear

[1] This section is a rework of selected results from the article **R. Conte**, **C. Aieta**, **G. Botti**, **M. Cazzaniga**, **M. Gandolfi**, **C. Lanzi**, **G. Mandelli**, **D. Moscato**, and **M. Ceotto**, Anharmonicity and nuclear quantum effects in theoretical vibrational spectroscopy: a molecular tale of two cities. *Theor. Chem. Acc.*

combinations of the vibrational frequencies with integer coefficients, such as $n\omega_a \pm m\omega_b$, for small values of $n, m \in \mathbb{N}$. Their occurrence is due to the anharmonicities and couplings of the potential. It can be seen by a perturbative expansion of the position variables and frequencies.[53] Another difference between the classical and quantum description of spectroscopy lies in the dependence of the vibrational frequencies on the temperature. In the classical world, the frequencies of vibration are in general a continuous function of the energy of the system, i.e., $\omega = \omega(E)$, and so they are a function of temperature. The quantum picture is described instead by the Schrödinger equation, and in the Schrödinger equation there is no temperature! In few words, the discrete Hamiltonian eigenvalues do not depend on temperature, and so the transition energy gaps between them. This means that quantum mechanical vibrational frequencies are temperature independent. As a matter of fact, temperature affects only the populations of the discrete energy levels, which influences the intensity and resolution of the spectral lines of the spectrum, leaving their position unchanged. In classical mechanics, the dependence of the vibrational frequencies on the simulation temperature is related only to the landscape of the potential explored during the simulation. The higher the simulation temperature, the higher the energy of the system, the more anharmonic the potential energy landscape. To prove these points we run simulations of the Formaldehyde molecule (CH_2O) vibrations at various temperatures and energies, from which we collect the vibrational spectra for comparison. The potential energy function of CH_2O is a quartic force-field fitted on *ab initio* data at CCSD(T)/cc-pVTZ level of theory. [54]

Our classical temperature dependent Molecular Dynamics (MD) and QCT runs differ in the way the dynamics is performed but they both rely on the mathematical formalism of the Fourier-transformed velocity–velocity autocorrelation function. The initial distributions of velocities for the MD simulations are obtained from the Maxwell–Boltzmann distribution. Then, a thermalization run at the desired temperature is performed by means of a stochastic velocity rescaling algorithm applied to an evolution of 200000 a.u. [55] followed by a production run 30000 a.u. long in an NVE ensemble (i.e., in a microcanonical ensemble where the number of particles, volume and total energy are held fixed). The production dynamics is used to collect the data necessary for calculating the vibrational frequencies [56]. QCT simulations start the trajectories at a target and harmonically quantized energy [28]. The starting configuration of all trajectories is chosen to be the equilibrium geometry, while initial atomic linear momenta are extracted from a Husimi distribution and then rescaled in a way that the total energy matches the target one, i.e., the harmonic zero point energy or a multiple of it. QCT trajectories are evolved in an NVE ensemble for a total time of 25000 a.u. For both MD and QCT a 4th order symplectic algorithm is adopted for

evolving the trajectories.

We focus on the highest fundamental frequency, labeled ν_6 in Table 2.1, because it is the frequency presenting the largest shift (-154 cm^{-1}). The normal mode associated to it, mode 6, is related to the $C-H_2$ asymmetric stretch.

Table 2.1: Quantum mechanical (QM) and harmonic (HARM) estimates of the fundamental vibrational frequencies of H_2CO

Frequency cm^{-1}	QM	HARM	Δ
ν_1	1171	1192	-21
ν_2	1253	1275	-22
ν_3	1509	1543	-34
ν_4	1750	1781	-31
ν_5	2783	2929	-146
ν_6	2842	2996	-154

From Fig. 2.2 it is evident that the estimated frequency is changing with the temperature or energy employed. More precisely, when MD is performed at very low temperature (10 K) or QCT is undertaken at an energy which is just 5% of the harmonic zero-point energy, the simulations return a frequency value which is basically the harmonic one.

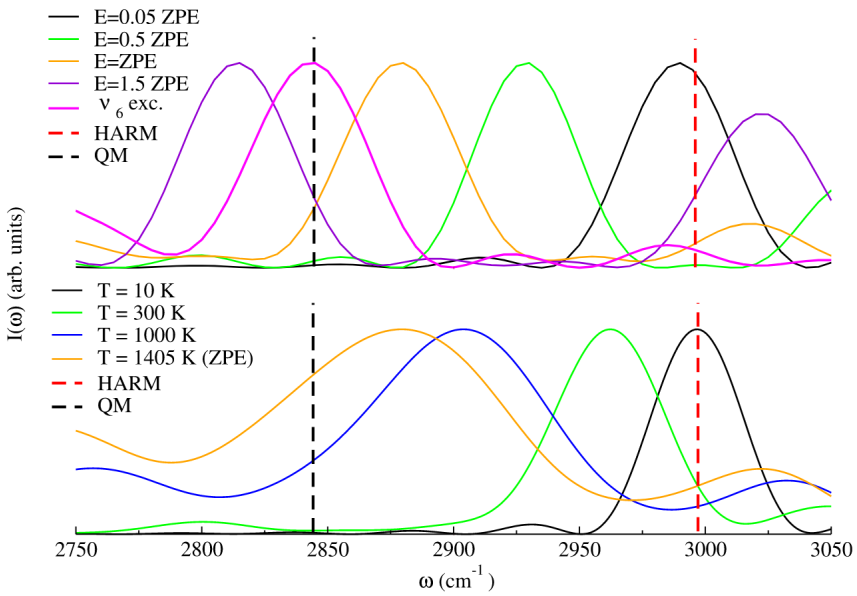


Figure 2.2: QCT (upper panel) and MD (lower panel) simulations of ν_6 in H_2CO. The maximum intensity has been set equal to 1 for all simulations

At such low temperature or energy the classical trajectory can sample only a region of the PES close to the equilibrium position. In this region the parabolic (harmonic)

approximation to the PES is valid and a frequency value close to the harmonic one is obtained. Remarkably, for the same reason, the MD simulation at room temperature (300 K) fails badly in estimating the frequency of vibration. Moving to higher energies (or temperature) we first notice that when the energy equals the harmonic ZPE, then the frequency estimate gets closer to the quantum mechanical benchmark (it is about 30 cm^{-1} away). Secondly and remarkably, the ν_6 exc. and 1.5 ZPE simulations are characterized by about the same energy (just a few wavenumbers difference) but while the ν_6 exc. trajectory is on the mark, the 1.5 ZPE trajectory is clearly underestimating the QM frequency.

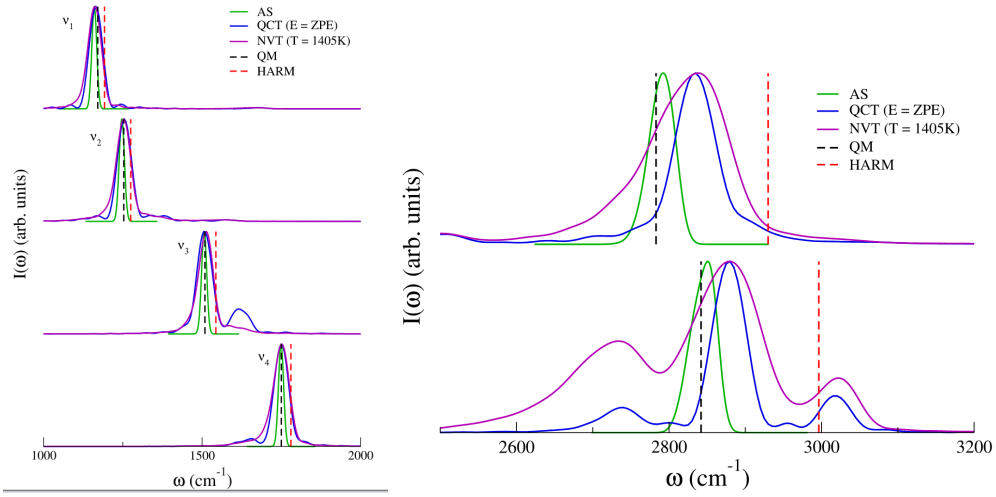


Figure 2.3: Spectra of modes ν_1, ν_2, ν_3, ν_4 (left), and ν_5, ν_6 (right) of CH_2O.

This is a demonstration that not only the total energy (temperature) of the trajectory but also the way it is distributed in the molecule may affect the outcome of a classical simulation. The fact that with the ν_6 exc. trajectory a classical method like QCT is able to reproduce the quantum frequency demonstrates that there are no sizeable quantum effects in play and anharmonicity can be detected by means of classical approaches. The nuclear quantum effects are recovered using the Semiclassical Initial Value Representation method (see section 2.2), with initial conditions sampled with the Adiabatic Switching (AS) technique. [57] Finally, we notice that for the trajectories run at high energy or temperature some spectroscopic features appear above 3000 cm^{-1}. Those are the higher harmonics associated to lower-frequency modes, in particular mode 3 and mode 4. It is not surprising that their intensity is quite high because we are not simulating dipole (IR) spectra, so intensities in Fig. 2.2 are not related to the actual absorption intensities but to the recurrence of trajectories in the phase space.

2.2 Semiclassical Initial Value Representation

As seen in the previous section, classical mechanics cannot capture quantum nuclear effects, such as the Zero Point Energy, overtones and combination bands. To capture these features we can resort to Semiclassical methods. In particular, to compute the power spectrum of a molecular system, we are interested in the survival probability amplitude. The survival probability amplitude $\langle\Psi(0)|\Psi(t)\rangle$ is a correlation function that measures the overlap between the moving wavepacket $|\Psi(t)\rangle$ and its initial state $|\Psi(0)\rangle$. Given the representation of $|\Psi(t)\rangle$ in the basis of eigenstates of the Hamiltonian, that is $|\Psi(t)\rangle = \sum_i c_i(t)\,|\psi_i\rangle$, with $\mathcal{H}\,|\psi_i\rangle = E_i\,|\psi_i\rangle$, the power spectrum is simply

$$
\begin{aligned}
I(E) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{iEt}\,\langle\Psi(0)|\Psi(t)\rangle\,dt && (2.5)\\
&= \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{iEt}\,\left\langle\Psi(0)|e^{-i\mathcal{H}t}\Psi(0)\right\rangle dt && (2.6)\\
&= \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{iEt}\left(\sum_i c_i(0)\,|\psi_i\rangle\right)^{\star}\left(\sum_j e^{-iE_jt}c_j(0)\,|\psi_j\rangle\right) dt\\
&= \frac{1}{2\pi}\sum_{i,j} c_i^{\star}(0)c_j(0)\,\langle\psi_i|\psi_j\rangle\int_{-\infty}^{\infty} e^{i(E-E_j)t}dt\\
&= \sum_i |c_j(0)|^2\,\delta(E-E_j). && (2.7)
\end{aligned}
$$

Eqs. 2.5 and 2.7, while being equivalent, have different pictorial interpretations. Eq. 2.7 sheds light on the fact that the power spectrum is a time-independent collection of Dirac delta functions centered on all the vibrational energy levels E_j, each of which is proportional to squared modulus of the basis expansion coefficient. Eq. 2.5 shows that the spectrum depends on the time-evolution of the wavepacket. Starting from the initial state $|\Psi(0)\rangle$ the wavepacket evolves in time and moves away from the initial configuration space. It moves back to a configuration arbitrarily close to the initial one after a period of time. The arbitrariness of how close it moves back is captured by the time-dependent overlap function $\langle\Psi(0)|\Psi(t)\rangle$. Thus the power spectrum is interpreted as the energies E at which the wavepacket oscillates from its initial state. The time-dependent approach to spectroscopy focuses on the power spectrum as defined in Eq. 2.6.

In the following paragraphs we summarize the ingredients required for the semiclassical version of Eq. 2.5, while we leave part of the demonstration in Appendix A. In section 2.2.1 we present the Divide-and-Conquer algorithm, that, within the semiclassical representation, splits the full-dimensional overlap $\langle\Psi(0)|\Psi(t)\rangle$ into a product of lower dimensional ones.

The semiclassical version of Eq. 2.5, using the time-averaging semiclassical initial value representation (TA-SCIVR) method is given in Eq. 2.8, [36, 58] where a time-averaging filter is applied to the semiclassical Heller-Herman-Kluk-Kay (HHKK) propagator [34, 35, 59–63].

$$I(E) = \left(\frac{1}{2\pi}\right)^{N_{vib}} \iint d\mathbf{p}_0 d\mathbf{q}_0 \frac{1}{2\pi T} \left| \int_0^T \mathrm{d}t \, \langle \chi | \mathbf{p}_t \mathbf{q}_t \rangle \, e^{i(S_t(\mathbf{p}_0,\mathbf{q}_0)+\phi_t(\mathbf{p}_0,\mathbf{q}_0)+Et)} \right|^2, \quad (2.8)$$

where T is the total simulation time, $S_t(\mathbf{p}_0, \mathbf{q}_0)$ the instantaneous action of the classically evolved trajectory $(\mathbf{p}_t, \mathbf{q}_t)$, and the phase-space integration is performed on the initial trajectory momenta $\mathbf{p}_0$ and positions $\mathbf{q}_0$. In Eq. 2.8, $|\mathbf{p}_t \mathbf{q}_t\rangle$ are coherent states with the following expression in position representation[64]

$$\langle \mathbf{x} | \mathbf{p}_t \mathbf{q}_t \rangle = \left(\frac{\det(\gamma)}{\pi^F}\right)^{\frac{1}{4}} \exp\left[-\frac{1}{2}(\mathbf{x}-\mathbf{q}_t)^{\mathrm{T}} \gamma (\mathbf{x}-\mathbf{q}_t) + i\mathbf{p}_t^{\mathrm{T}}(\mathbf{x}-\mathbf{q}_t)\right], \quad (2.9)$$

where γ is an $N_{vib} \times N_{vib}$ diagonal matrix, whose elements are equal to the harmonic frequencies of the system. In Eq 2.8, $\phi_t(\mathbf{p}_0, \mathbf{q}_0)$ is the phase of the HHKK prefactor[65]

$$\phi_t(\mathbf{p}_0, \mathbf{q}_0) = \mathrm{phase}\left[\sqrt{\frac{1}{2^{N_{vib}}} |\mathbf{M_{qq}}(t) + \gamma^{-1}\mathbf{M_{pp}}(t)\gamma - i\mathbf{M_{qp}}(t)\gamma + i\gamma^{-1}\mathbf{M_{pq}}(t)|}\right], \quad (2.10)$$

where $\mathbf{M_{ij}}$, with $\mathbf{i}, \mathbf{j} = \mathbf{p}, \mathbf{q}$, are the elements of the symplectic monodromy (or stability) $2N_{vib} \times 2N_{vib}$ matrix

$$\mathbf{M}(t) = \begin{pmatrix} \mathbf{M_{pp}}(t), & \mathbf{M_{pq}}(t) \\ \mathbf{M_{qp}}(t), & \mathbf{M_{qq}}(t) \end{pmatrix} = \begin{pmatrix} \dfrac{\partial \mathbf{p}_t}{\partial \mathbf{p}_0}, & \dfrac{\partial \mathbf{p}_t}{\partial \mathbf{q}_0} \\ \dfrac{\partial \mathbf{q}_t}{\partial \mathbf{p}_0}, & \dfrac{\partial \mathbf{q}_t}{\partial \mathbf{q}_0} \end{pmatrix}. \quad (2.11)$$

The monodromy matrix represents the dependence of the classical system variables $\mathbf{p}_t, \mathbf{q}_t$ from the initial conditions. Since it is a Jacobian matrix of the system coordinates, its determinant must be equal to 1 at all times. More precisely, $\mathbf{M}(t)$ must remain symplectic during the propagation, which requires a symplectic integration algorithm to be used for semiclassical dynamics. Furthermore, the time evolution of the monodromy matrix, following Hamilton's equations, is

$$\frac{d}{dt}\mathbf{M}(t) = \mathbf{A} \cdot \mathbf{M}(t) \quad (2.12)$$

where

$$\mathbf{A} = \begin{pmatrix} 0 & -\mathbf{H}(t) \\ 1/\mathbf{m} & 0 \end{pmatrix} \quad (2.13)$$

and where $\mathbf{H}(t)$ is the Hessian matrix at time t. Thus, it is necessary an accurate Hessian calculation to have an accurate $\mathbf{M}(t)$ matrix and an accurate vibrational power spectrum. For those reasons, during semiclassical dynamics calculations, one must monitor the accuracy of $\mathbf{M}(t)$ by exploiting its symplectic property and checking the deviation of its determinant (or better, the determinant of the positive-definite matrix $\mathbf{M}^T(t)\,\mathbf{M}(t)$) from unity. As we discuss in chapter 4 the calculation of the monodromy matrix is the bottleneck of semiclassical IVR calculations, but it can be tamed using approximate techniques.

To beat the curse of dimensionality we reduced the phase space integration of Eq. 2.8 to a few, or just one, trajectory simulations, where each trajectory starts from the global minimum and with an energy equal to the harmonic vibrational energy level that one is looking for.[66] This method is called Multiple Coherent SCIVR (MC SCIVR) [38, 67] and it exploits the fact that during the simulation, the delocalization of the coherent states will account for anharmonicity and reproduce the anharmonic vibrational peak position. [43] The method allows also to identify each mode contribution by selecting a suitable combination of coherent states obtained by changing the sign of the momentum part of the coherent state. [68]

2.2.1 Divide-and-Conquer Approximation and Multiple Coherent states initial conditions

The Divide and Conquer (DC) basic idea is to calculate the full dimensional spectrum as the composition of lower dimensional ones using Eq. 2.8 but with reduced dimensionality phase space quantities. In Eq. 2.8 the potential, which is part of the action, is the only quantity that can not be exactly projected in a reduced dimensionality space. For this reason, we have introduced the following approximation for the partial M-dimensional spectrum

$$\tilde{S}_t(\tilde{\mathbf{p}}_0, \tilde{\mathbf{q}}_0) = \int_0^t dt' \left[\frac{1}{2} \tilde{\mathbf{p}}_{t'}^T \tilde{\mathbf{p}}_{t'} - \left(V(\tilde{\mathbf{q}}_{t'}, \mathbf{q}_{t'}^{(N_{vib}-M)}) - V(\tilde{\mathbf{q}}_{eq}, \mathbf{q}_{t'}^{(N_{vib}-M)}) \right) \right], \qquad (2.14)$$

where the tilde $\sim$, in this context, indicates a M-dimension quantity, $V(\tilde{\mathbf{q}}_{t'}, \mathbf{q}_{t'}^{(N_{vib}-M)})$ is the full dimensional potential and $V(\tilde{\mathbf{q}}_{eq}, \mathbf{q}_{t'}^{(N_{vib}-M)})$ the one obtained by fixing at equilibrium the coordinates in the M-dimensional subspace. This approach has been successfully applied to several high dimensional complex systems, including fluxional ones, like small water clusters[44] and the protonated water dimer.[46] In the DC-SCIVR method one needs to properly partition the full dimensional vibrational space. One can reach this goal by coarse-graining the time-averaged Hessian matrix or by splitting the Jacobian (Monodromy) matrix $\mathbf{M}(t)$ in square blocks, such that the determinant of each block is as close as possible to 1, in partial satisfaction of Liouville's theorem. In

either case, the result is a block diagonalized matrix, where each block represents a vibrational subspace. If one chooses to use the Jacobian approach, it is convenient to use the Probability Graph-Evolutionary Algorithm (PG-EA) discussed chapter 3. [51]

When combined with MC SCIVR by projecting few full-dimensional classical trajectories into sub-dimensional phase space components, we obtain the MC-DC SCIVR, which can deal with very high-dimensional systems. The MC SCIVR method relies on the idea that the most important contribution to the spectrum comes from the trajectories whose energies are as close as possible to the quantum mechanical eigenvalues.[66] Thus, in MC SCIVR, the phase-space integral of Eq. (2.8) is formally replaced by a sum over the most relevant trajectories, i.e. those corresponding to the spectral signals of interest. The MC-SCIVR initial conditions for the mass-scaled j-th degree of freedom are[38, 67]

$$\begin{cases} q_0^{(j)} = q_{eq}^{(j)} \\ p_0^{(j)} = \sqrt{(2n_j+1)\omega_j} \end{cases} , \tag{2.15}$$

being $q_{eq}^{(j)}$ the equilibrium position of the j-th mode, and ω_j its harmonic frequency. Our reference states are combinations of coherent states of the type

$$|\chi\rangle = \prod_j^F \left(|p_0^{(j)}, q_0^{(j)}\rangle + \epsilon_j | - p_0^{(j)}, q_0^{(j)}\rangle \right), \tag{2.16}$$

where $\epsilon_j = \pm 1$ according to which spectroscopic signal one wants to enhance. For example, a collection of $+1$ values allows one to enhance the ZPE signal (together with the even transitions), while a selected $\epsilon_j = -1$ and the remaining $\epsilon_{i \neq j} = +1$ enhances the odd transition of the j-th mode.

Chapter 3

Divide-and-Conquer Jacobian Divisibility Criterion[1]

In this chapter we discuss in more detail the implementation of an evolutionary algorithm that aims to enforce the "Jacobian" criterion for DC-SCIVR. The objective is to partition the nuclear degrees of freedom into disjoint sets called subspaces. The optimal choice of the subspaces is such that Liouville theorem is approximately satisfied for each set. This translates to the search of a partition of normal modes, such that the monodromy matrix of each subspace has a determinant close to 1. Unfortunately, the number of ways one can partition a set into subsets grows as the Bell's numbers B_n

$$\begin{cases} B_0 = 1 \\ B_1 = 1 \\ B_n = \sum_{k=0}^{n} \binom{n}{k} B_k, \end{cases} \tag{3.1}$$

that is a super-exponential scaling series. Because of this very unfavorable scaling, the optimal choice of subspaces becomes virtually unreachable for medium and large systems. Thus it may become more appropriate to first identify mid-sized subspaces (say of 20 dimensions) according to a more heuristic criterion, and then split the subspace with the PG-EA algorithm.

3.1 Probability Graph-Evolutionary Algorithms (PG-EA)

Here we introduce a combined Probability Graph and Evolutionary Algorithm (PG-EA) approach to find the best vibrational space subdivision according to Liouville's

[1] This chapter is a selection with minor modifications of the contents of the paper **Michele Gandolfi**, **Alessandro Rognoni**, **Chiara Aieta**, **Riccardo Conte**, and **Michele Ceotto**, "Machine learning for vibrational spectroscopy via divide-and-conquer semiclassical initial value representation molecular dynamics with application to N-methylacetamide"

criterion explained above. Evolutionary algorithms emulate the natural selection of an initial population, where the "fittest" individual is the most likely to survive and its genes to be inherited by the next generation. In GAs jargon, the population is composed of chromosomes, which are collections of fitness parameters, each one called a gene. There is no obvious or required way to represent the genes and there are many valid choices. At each epoch, all chromosomes are evaluated and sorted according to their fitness score. First, a fraction of the best individuals give birth to a set of newborn chromosomes by mixing and mutating genes during the crossover and mutation processes. Then, the new chromosomes take the place of those individuals that are least fit to survive, so that the next epoch would be enriched by the more fitted chromosomes.

In our case, each chromosome represents a possible clustering of vibrational degrees of freedom into subspaces to compose the full vibrational space. The collection of all the chromosomes provides many possible subdivisions of the full-dimensional vibrational space. Each chromosome is evaluated by an appropriate score function that rates the individual's fitness. The fitness function is evaluated after time evolution of the Jacobian matrix along a test trajectory, which in our case is the trajectory that evolves from the equilibrium geometry with the energy of the vibrational ground state. We propose the following fitness function for a possible collection C of normal modes subdivisions

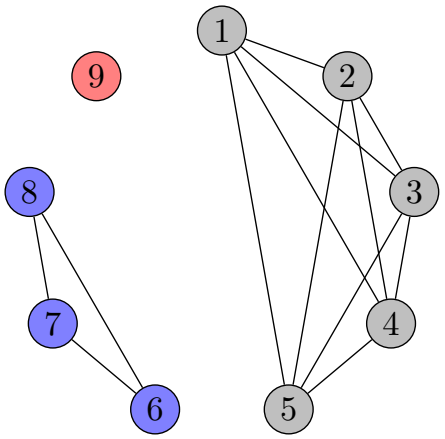
$$f(C) = \frac{1}{N} \sum_{\text{steps}}^{N} \sum_{s \in C} \left| 1 - \left| \det(\tilde{\mathbf{J}}_s) \right| \right|, \tag{3.2}$$

where the external sum is over the N time steps. We prefer Eq. (3.2) with respect to a possible fitness function, such as $|1 - \prod_i^{n_s} \det \tilde{\mathbf{J}}_i|$ as employed in Ref.[50], because in the latter case there could be a compensation of error that the internal sum in Eq. (3.2) avoids. More specifically, the optimal criterion in Eq. (3.2) is satisfied when all the subspaces have the determinant of the Jacobian closest to $+1$ in modulus, at every trajectory step. Furthermore, this function rewards preferentially a chromosome made of few large subspaces over one made of several small subspaces because any new term in the internal sum over C is addictive and positive. We prefer to have large subspaces to account for as many normal modes couplings as possible.

Once an initial guess of possible chromosomes is given, we need a probability distribution function to generate the new chromosomes, i.e. the new mode subdivision into subspaces. We propose our own customized Evolutionary Algorithm inspired to GAs for updating the probability distribution $\mathbf{\Phi}(\tau)$ from which newborn chromosomes are sampled at a certain epoch τ.

First, we represent a chromosome as the adjacency matrix of an unweighted cluster graph, that is a graph where each connected component is a clique (i.e. a fully connected subgraph), as reported in Figure 3.1. The vertexes are the normal modes that are

connected only if they fall into the same subspace. This representation minimizes the redundancy of information, since cliques are invariant to vertex permutation and a cluster graph is invariant to clique permutations.



$$\mathbf{C} = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Figure 3.1: Chromosome expressed in a cluster graph representation. Here, 9 normal modes, corresponding to the graph vertexes, are grouped into three subspaces (gray, blue, and red), containing 5, 3, and 1 modes, respectively. Each subspace is a fully connected graph (a clique), and there is no interaction between any two subspaces. On the right, the corresponding adjacency matrix is colored accordingly.

All the information about the subspaces is codified in the adjacency matrix $\mathbf{C}$. It is defined by $C_{ij} = 1$ only if modes i and j are in the same subspace, otherwise $C_{ij} = 0$. $C_{ii} = 1$ only if mode i is in a one-dimensional subspace. In the end, we have increased the problem variables from the F-dimensional redundant representation (such as in the linear representation (1,2,3,4,5)(6,7,8)(9)=(2,4,1,5,3)(8,7,6)(9)=...) to the $F(F+1)/2$-dimensional non-redundant one. The F-dimensional representation is redundant because any mode permutation within a given subspace leaves the subspace unaffected. Our adjacency matrix representation is invariant to row and column permutation within the subspace block. Hence, every subspace configuration has a unique adjacency matrix representation. Furthermore, our adjacency matrix is symmetric, thus completely defined by its F(F+1)/2 lower (upper) triangular elements.

Secondly, we customize the crossover and mutation operators to mix the chromosomes with simple arithmetic rules and store the genetic information in a matrix of weights $\mathbf{\Phi}(\tau)$, which represents the probability distribution of the mixed chromosomes at the evolution epoch τ. We define the crossover $\mathcal{X}$ of a couple of chromosomes C and C' as a weighted average of their adjacency matrices

$$\mathcal{X} = \frac{1}{w_C + w_{C'}} \left(w_C \mathbf{C} + w_{C'} \mathbf{C}' \right), \tag{3.3}$$

where $\mathbf{C}$ and $\mathbf{C}'$ are adjacency matrices and the weights w_C and $w_{C'}$ depend on the

chromosome fitnesses. The pure mutation $\mathcal{M}$ of a chromosome C with probability μ is

$$\mathcal{M} = \frac{1}{1+F\mu}\left(\mathbf{1}\mu + \mathbf{C}\right), \tag{3.4}$$

where $\mathbf{1}$ is the square matrix of ones and F is the number of genes. In our approach each vibrational normal mode corresponds to a gene, as illustrated in the previous example of Fig. 3.1. Both crossover and mutation have the basic property of scattering the gene probability. More specifically, the crossover distributes the probability among the genes expressed in C and C', depending on their fitness, and the mutation distributes it among every possible outcome, independently of the fitness and the expression. The combination of the crossover and mutation processes is obtained by the subsequent application of the two operations: the mutation can be equivalently applied to the single chromosomes previous to undergoing crossover, or to the result of the crossover process. In addition, considering that we use m chromosomes for the optimization, only an elite fraction $0 < \eta < 1$ of these are the fittest chromosomes that undergo the crossover and mutation processes. Eventually, the resulting probability distribution at epoch τ, $\mathbf{\Phi}(\tau)$, is:

$$\mathbf{\Phi}(\tau) = \frac{1}{1+F\mu}\left[\mathbf{1}\mu + \frac{1}{\sum_i^{m\eta} w_i}\sum_{i=1}^{m\eta} w_i \mathbf{C}_i\right], \tag{3.5}$$

where C_i is the i-th chromosome adjacency matrix and the new probability distribution is essentially generated by suitably mixing the adjacency matrices of the previous generation. This is the machine learning part, where the algorithm, epoch by epoch, learns the optimal probability distribution from an evolving population of chromosomes. For this work we use the simple weighting scheme $w_i = (m\eta - i)/m\eta$ with the resulting normalization constant $\sum_i^{m\eta} w_i = (m\eta - 1)/2$, the elite fraction is $\eta = 0.4$, while the mutation probability μ and the number of chromosomes will be specified below case by case. $\mathbf{\Phi}(\tau)$ is updated at every epoch and contains the average genetic material of the previous generation of chromosomes according to Eq. (3.5).

To sample new chromosomes from the probability distribution, $\mathbf{\Phi}(\tau)$ must be normalized. For the expression in Eq. (3.5), this means that $\mathbf{\Phi}(\tau)$ has to be symmetric and doubly stochastic, i.e. with rows and columns summing up to 1, so that we can consider $\mathbf{\Phi}(\tau)$ a weighted undirected graph to sample from. To enforce the doubly stochastic property and, at the same time, retain the symmetry, we rely on Sinkhorn's theorem,[69] which ensures that there exist two diagonal matrices $\mathbf{R}$ and $\mathbf{S}$ such that $\mathbf{R\Phi}(\tau)\mathbf{S}$ is doubly stochastic. R_{ii} and S_{jj} are found by repeatedly and alternatively

normalizing the rows and the columns of $\mathbf{\Phi}(\tau)$, according to the updates

$$\begin{aligned} R_{ii} &= \frac{1}{\sum_j \Phi_{ij}(\tau) S_{jj}} \quad \forall\ i \\ S_{jj} &= \frac{1}{\sum_i \Phi_{ij}(\tau) R_{ii}} \quad \forall\ j\ , \end{aligned} \tag{3.6}$$

with S_{jj} initialized to 1 for all j. $\mathbf{R}$ and $\mathbf{S}$ will converge, up to a small threshold ε, after an unspecified number of iterations[69]. In all the applications described below we use $\varepsilon = 10^{-8}$ on each element of $\mathbf{R}$ as and $\mathbf{S}$, which is always satisfied in less than 100 iterations.

Third, we need to elaborate a procedure for obtaining the newborn chromosomes from the symmetric and doubly stochastic probability matrix $\mathbf{\Phi}(\tau)$. To sample representative cluster graphs from $\mathbf{\Phi}(\tau)$, we propose a sampling procedure to generate a population which reflects the original distribution:

1. generate the random numbers $r_i, i = 1, 2, \ldots F$ and sample independently the chances of each mode to be in a subspace alone. If $r_i < \Phi_{ii}(\tau)$ then the normal mode i is in a subspace alone;
2. iterate on the leftover modes in a random order: if the k-th mode is already joined with another mode, then continue with the next one, otherwise sample the edge between modes k and $j \neq k$ with a random number and join them with a probability given by the matrix element $\Phi_{jk}(\tau)$. If the k-th cannot be joined to any j (for instance, because each of them is in a one-dimensional subspace) it stays in a subspace by itself;
3. identify the connected components of the sampled graph and complete them, obtaining the cluster graph for a newborn chromosome.

Note that before step 3, the procedure samples tree graphs, which means that there are no redundant sampling steps. Each chromosome sampled with this procedure may be weakly biased anyway and the random shuffle of the mode order in step 2 is required to make the sampled population representative and the overall sampling unbiased. To achieve step 3 we look for a basis of the Laplacian matrix Kernel, with the Laplacian matrix defined as $L_{ij} = -C_{ij} + \delta_{ij} \sum_j C_{ij}$. Since $\sum_j L_{ij} = 0$ by definition, the vector of ones always belongs to $Ker(\mathbf{L})$, i.e. to the collection of vectors $\mathbf{x}$ such that $\mathbf{Lx} = \mathbf{0}$. Furthermore, if the graph is disconnected, $\mathbf{L}$ can be rearranged to be block diagonal by swapping row and column indexes, with each block being the Laplacian of the corresponding connected subgraph. Hence each basis vector of $Ker(\mathbf{L})$ is 1 on the entries of the connected vertices and 0 elsewhere. The sum of all basis vectors is the vector of ones. To practically find a basis for $Ker(\mathbf{L})$ we solve the linear equation $\mathbf{Lx} = \mathbf{0}$ by applying Gaussian Elimination[70] to the augmented matrix $\mathbf{L}|\mathbf{I}$, with output $\mathbf{L_{rref}}|\mathbf{B}$, where $\mathbf{L_{rref}}$ is the reduced row echelon form of $\mathbf{L}$. The rows $\mathbf{x}$ in $\mathbf{B}$ corresponding to

the row indices where $\mathbf{L_{rref}} = \mathbf{0}$ do solve the linear equation and hence form a basis for the kernel.

To measure the likelihood of a subspace s of size D sampled from $\mathbf{\Phi}(\tau)$, we sum the edge products of all the possible trees that span the clique (subspace), as

$$p(s,\tau) = \frac{(D-1)^{D-1}}{D^{D-2}} \sum_{\mathbf{T}\in span(\tilde{\mathbf{\Phi}}_s(\tau))} \prod_{e=1}^{D} T_e, \tag{3.7}$$

where T_e is the edge of the tree graph $\mathbf{T}$, which spans the subspace probability distribution $\tilde{\mathbf{\Phi}}_s(\tau)$ (that is the probability distribution considering only the modes in s). The first factor is a normalization constant so that $p(s,\tau)$ does not depend on the subspace size and it is maximized to 1 when $\tilde{\mathbf{\Phi}}_s(\tau)$ is uniform. D^{D-2} is the number of spanning trees for a clique according to Cayley's formula.[71] $p(s,\tau)$ measures the degree of convergence towards the chosen subspace s, such that, as $p(s,\tau)$ approaches unity, the population becomes more and more uniform and eventually the algorithm stops learning. The brute force application of Eq. (3.7) is out of reach for large subspaces ($D >\approx 10$), therefore, we use it only to check the algorithm progression towards an optimal solution of the small systems described below. Furthermore, GAs in general, and PG-EA in particular, do not require a full convergence of the population for the solution to be satisfactory. On the contrary, if the solution is unknown or hard to find, a homogeneous population is undesirable, as it kills diversity and damps the optimization.

In Fig. (3.2) we report a four normal mode example to show how the PG-EA algorithm works in practice. First, as we do in all our simulations, the initial probability distribution is set as $\Phi_{ij}(0) = 1/F \;\forall\; i,j$. Then, we generate the initial chromosome population according to the sampling procedure described at points 1-3 above. Each chromosome graph is reported together with the corresponding adjacency matrix in panel 1 in the left side of Fig. (3.2). The chromosomes C_j are ordered by the fitness score, which is calculated using the function $f(C)$ (pay attention that according to our definition of the fitness function, Eq. 3.2, the preferred chromosomes are those with lower fitness score). The chromosome fraction $m\eta$ undergoes crossings and mutations. We reject (i.e. apply an infinite penalty) to any subspace which has a dimension larger than the largest subspace value $\bar{D}$ that one fixes *a priori*. Since assigning the fitness score is the most expensive step, it is advisable to build a score database, *i.e.* a list of already known chromosomes with their fitness value, so that, whenever a known chromosome is encountered its score does not have to be recomputed. After undertaking crossovers and mutations according to Eq. (3.5), a new (non-normalized) probability distribution is generated (panel 2 on the right side of Fig. 3.2). After transforming the new probability distribution into a symmetric and doubly stochastic distribution matrix using Sinkhorn's algorithm, we generate the newborn chromosomes according to the sampling procedure

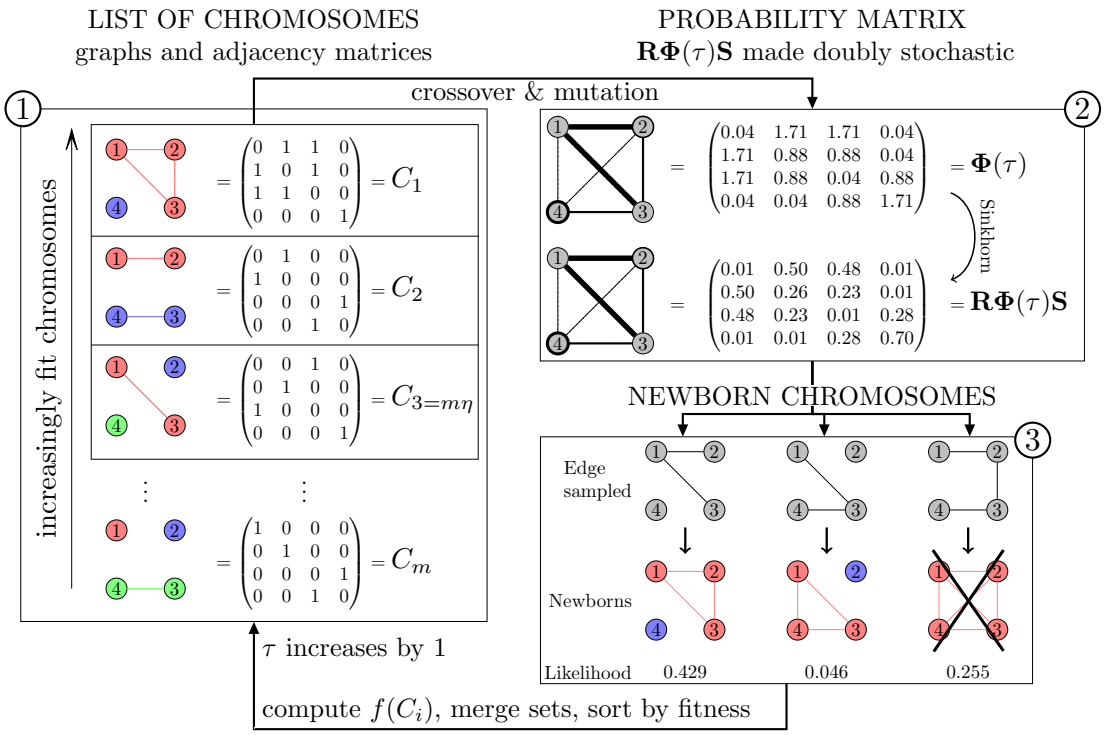


Figure 3.2: A numerical example of the PG-EA for a four dimensional system. The procedure is broken down into 3 steps. The inner rectangle in panel 1 includes the $m\eta$ elite chromosomes which will be employed for mutation and crossover in panel 2.

of points 1-3 described above. A likelihood coefficient is calculated according to Eq. (3.7). In the lower right part of Fig. 3.2, the 4-dimensional solution is rejected because its dimensionality is greater than the largest subspace value $\bar{D}$, which in this numerical example has been fixed to be $\bar{D} = 3$. Finally, a fitness coefficient is attributed to each chromosome and we are back to panel 1 for the next iteration. At each epoch, the fittest i-th chromosome, i.e. the one with the lowest $f\,(C_i)$ value, provides the graph with the so far optimal normal modes arrangement into subspaces.

3.2 2-mode interaction method

As an alternative, we propose an approximate and computationally cheaper method to deal with large molecules when the computational cost of PG-GA is prohibitive or in instances in which one can reasonably assume that for each normal mode the coupling is mainly due to the interaction with just a second mode.

In this alternative approach, we first compute a 2-mode coupling network, where each vertex i, j is a normal mode and each edge is weighted by the two-mode Jacobian determinant $G_{ij} = \det(\tilde{\mathbf{J}}_{ij})$. $\tilde{\mathbf{J}}_{ij}$ is a 4×4 matrix containing all the partial derivatives between the phase-space momenta and positions of modes i and j with respect to the

initial conditions. For each full-dimensional Jacobian matrix **J** along the trajectory, we evaluate the determinant of every two-mode combination, G_{ij}. Then, we compute the error matrix $\mathbf{E} = |\mathbf{G} - \mathbf{1}|$, which measures how large is the error done by assuming that the relevant interaction is only between the couple (i,j) of modes while other interactions are disregarded. Specifically, when $E_{ij} = 0$, then modes i and j are fully correlated and uncoupled to any other mode. **E** is computed at every time step of the test trajectory and all **E** matrices are averaged into a single error matrix representative of the whole trajectory.

Then, we employ an agglomerative hierarchical clustering technique called Weighted Pair Group Method with Arithmetic mean (WPGMA) [72] to cluster the normal modes using the information encoded in **E**. The algorithm produces a dendrogram where each branching is an optimal subspace. Among the several hierarchical clustering techniques available in the literature, we choose WPGMA because it provides results which are the closest to the exact ones for the model systems considered below. WPGMA clustering is iterative and hierarchical. To start, each mode is in a subspace by itself. Then, at every iteration, the two "closest" subspaces are merged into one and the dendrogram profile shows a link. The error between the newly formed subspace $(j \cup k)$ and a given subspace i is calculated as the arithmetic mean of the errors from the newly merged subspaces j and k:

$$E'_{i(j\cup k)} = \frac{E_{ij} + E_{ik}}{2}. \tag{3.8}$$

The procedure goes on until a maximum error criterion has been met, i.e. until all modes fall into one large subspace.

The whole process is represented by a dendrogram, where each node corresponds to a subspace and each edge represents the link between two subspaces. At the root of the dendrogram there is the full-dimensional system, which contains all modes. The leaves are the subspaces containing one mode only. The error E_{ij} of every update is a measure of how close the linked subspaces are. Finally, this process generates a number of arrangements of normal modes at different level of the tree, and for every such arrangement we measure the fitness score, i.e. the full-dimensional Jacobian factorization error with Eq. (3.2), along with E_{ij}.

3.3 Results

This section presents our results and it can be divided into three parts. In Sec. 3.3.1, we show how PG-EA and the 2-mode interaction method are effective when applied to model systems like coupled Morse oscillators with non-trivial coupling topologies but obvious mode separations. In subsection 3.3.2, we show that we can improve spectra accuracy with respect to previous calculations where the hierarchical subspace

optimization originally proposed was adopted.[50] Finally, in subsection 3.3.3 we show that PG-EA allows us to apply the DC-SCIVR method and select the subspaces with the Jacobi criterion even for simulation of mid-large molecules such as the 12-atom *trans*-N-Methylacetamide. Remarkably, it would not have been possible to accomplish this task with a brute force combinatorial approach.

3.3.1 Model systems

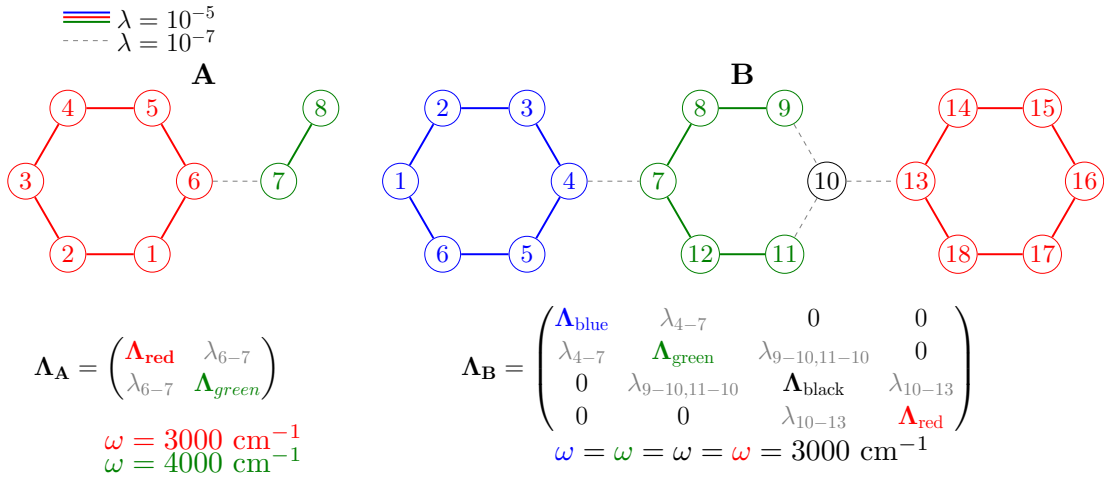


Figure 3.3: Toy model systems with different coupling topologies. The circled numbers represent Morse oscillators and the edges their couplings. When two oscillators are not connected by an edge, the relative coupling matrix element λ_{ij} is zero. The oscillator frequencies ω, reported below each coupling matrix, are 3000 cm^{-1}except for modes 7 and 8 of topology **A**, for which $\omega = 4000$ cm^{-1}. Note that for topology **A** reported on the left, oscillators 1 and 5 are equivalent and they produce the same signal, as well as 2 and 4. Similar symmetry considerations can be applied to topology **B**.

To preliminarily test our algorithms, we consider the arrangements of F coupled Morse oscillators **A** and **B** in figure 3.3. Each oscillator experiences the following Morse-type potential

$$V^{morse} = \sum_{i=1}^{F} D_e \left(1 - e^{-\omega_i (2D_e)^{-1/2}(q_i - q_{eq,i})}\right)^2 + \sum_{i=1}^{F-1} \sum_{j>i}^{F} \lambda_{ij}(q_i - q_{eq,i})(q_j - q_{eq,j}), \tag{3.9}$$

where the dissociation energy $D_e = 38293$ cm^{-1}and the equilibrium position $q_{eq} = 1.4$ a.u. are valid for all F degrees of freedom. According to the coupling graphs and matrices $\mathbf{\Lambda}$ schematically represented in Figure 3.3, the oscillators might be uncoupled (no edge), weakly coupled ($\lambda = 10^{-7}$ a.u., dashed edge) or strongly coupled ($\lambda =$

10^{-5} a.u., solid edge). We devise two topologies (**A** and **B**) to provide non-trivial examples. **A** has oscillator frequency $\omega = 3000\,\mathrm{cm}^{-1}$ for the oscillators 1 through 6 and $\omega = 4000\,\mathrm{cm}^{-1}$ for oscillators 7 and 8; **B** has all oscillators with the same frequency $\omega = 3000\,\mathrm{cm}^{-1}$. In both cases the correct separation into subspaces is unique.

Both PG-EA and the 2-mode interaction method separate system **A** correctly. In PG-EA we use $m = 50$ chromosomes, mutation probability $\mu = 0.001$, crossover fraction $\eta = 0.4$ and 70 epochs, with the constraint that the maximum dimension is $\bar{D} = 6$. The likelihood of the optimal subspaces calculated using Eq. (3.7) is plotted against the epochs in panel (a) at the top left of Fig. 3.4, showing that the population quickly converges to the unique global optimum represented by the continuous lines. The 2-mode interaction method provides three choices for the subspaces, the best of which is the global optimum (1,2,3,4,5,6)(7,8), reported in the first branching of the dendrogram in panel (b) at the top right of Fig. 3.4. This global optimum has a Jacobian factorization error of about $3.37 \cdot 10^{-6}$. The example of topology **B** is much more challenging, as it is an 18-dimensional system divided in 4 loosely connected regions, one of which containing a single mode. As expected, it turns out that the best subspace division has oscillator 10 (black subspace in Fig. 3.3) joined together with five oscillators (7-12, green subspace), since there is one term less in the fitness function summation with respect to the case in which oscillator 10 is left isolated. In panel (c) at the bottom left of Fig. (3.4), PG-EA provides the optimal desired solution using $m = 300$ chromosomes, $\mu = 0.1$, $\eta = 0.4$ and 1000 epochs, with the constraint that the largest subspace dimension is $\bar{D} = 10$. In panel (d) at the bottom right of Fig. 3.4, the 2-mode interaction method also provides the optimal solution, as shown in the upper part of the dendrogram with the smallest Jacobian factorization error.

Now, one may wonder if these subdivisions are indeed the most suitable ones for DC-SCIVR spectroscopic calculations. In Figure 3.5 we show that DC SCIVR can account properly for most of the spectral features of these systems, if the subspaces are chosen accordingly to the algorithms described above. For example when choosing the subspace separation (1,3,5,8)(2,4,6,7), which is the least fit for case **A** i.e. it has the largest fitness score in case of requiring 2 subspaces only, the corresponding spectra are quite noisy as shown in Figure 3.5. Furthermore, phantom signals are observed, for example at $2686\,\mathrm{cm}^{-1}$. Conversely, the spectra of the subspaces suggested by both our algorithms, which are reported with green and red lines, are without noise to the naked eye. However, we notice that the signal originated from a combination band of modes from different subspaces at $7006\,cm^{-1}$ is too weak in this scale to be observed. This is not a drawback of the algorithms proposed in this work, but a known feature of the DC-SCIVR method in predicting mixed overtones originated from modes belonging to different subspaces.

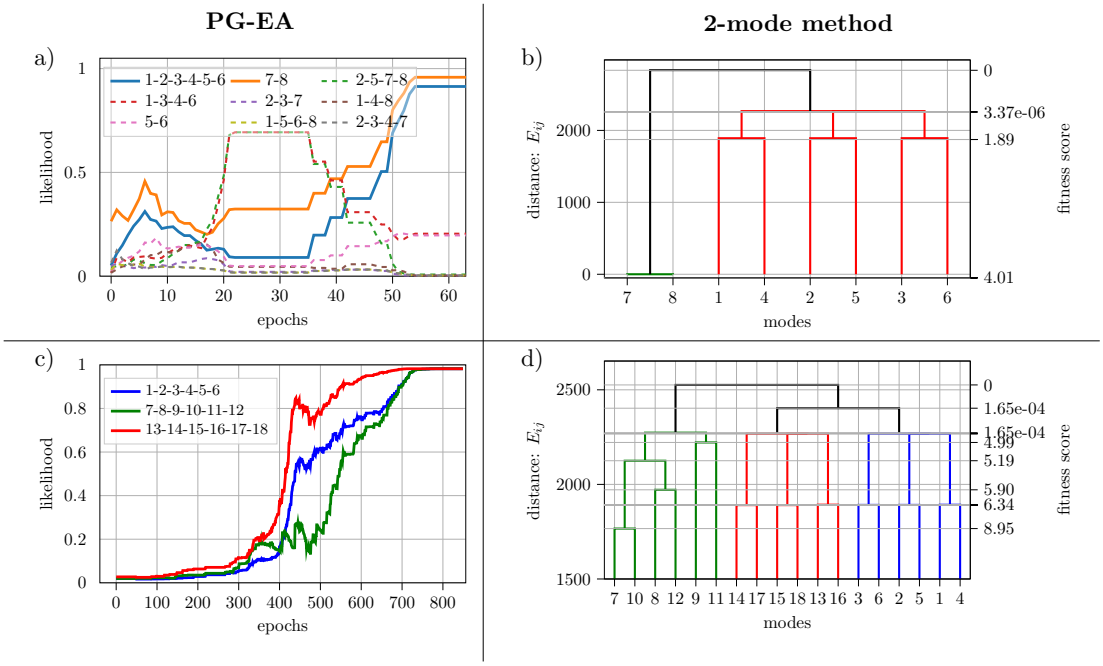


Figure 3.4: Vibrational modes subdivision optimizations of the model systems in Fig. (3.3) using PG-EA (left panels (a) and (c)) and the 2-modes interaction method (right panels (b) and (d)). In the top panels ((a) and (b)) the subspace optimizations of topology **A** is effectively achieved using both methods. The dendrogram (panel (b)) is colored to highlight the least error branching. In the bottom panels (panels (c) and (d)) the subspace optimization is effectively reached by both methods for topology **B**.

On the right part of Fig. 3.5 we show the optimal subspace spectra for system **B**. In this case, the system is too large to have a well-converged semiclassical full-dimensional TA-SCIVR spectrum as shown by the black continuous line spectrum[68]. Instead, it is possible to recover the most significant spectroscopic features of the system with a DC-SCIVR calculation based on the optimal subspaces suggested by the algorithms.

3.3.2 The CH_4 Molecule

This section further confirms the ability of the proposed algorithms to find optimal subspace separations for DC-SCIVR calculations when applied to real systems. We show that our techniques can reproduce and improve the DC-SCIVR spectra even for small molecules. We consider CH_4 as the case system. Methane vibrational spectrum is a tough challenge for DC SCIVR because the molecule is characterized by highly chaotic dynamics and high symmetry which is difficult to recover if a proper subspace partition is not implemented. We simulate 180000 trajectories 30000 a.u. long and each trajectory is rejected if during the dynamics $||\det(\mathbf{J}^T\mathbf{J})| - 1| > 10^{-5}$. The initial trajectory conditions are sampled from the Husimi distribution centered in phase space

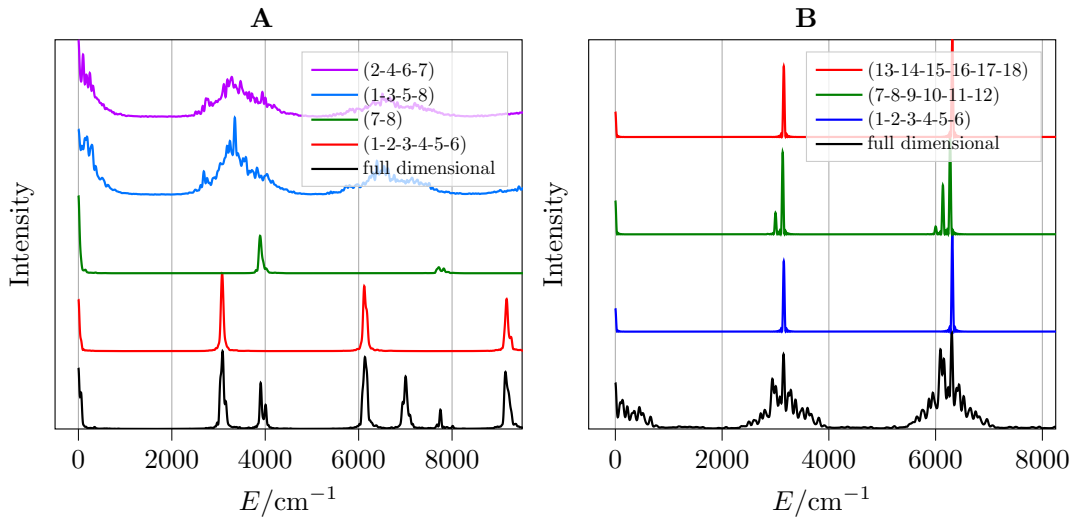


Figure 3.5: Spectra of the coupling topologies **A** and **B** of Morse oscillators with ZPE signals shifted to 0 cm^{-1}. Left panel for system **A**: in black the full-dimensional spectrum; in red and green the best subspaces (ring and segment respectively with reference to figure 3.3), while the blue and purple are the two worst subspaces. Right panel for system **B**: in blue, dark green and red the best subspaces; in black the full-dimensional calculation.

at $(\sqrt{\boldsymbol{\omega}}, \mathbf{q}_{eq})$, while gradients and Hessian matrices are computed by finite differences with infinitesimal displacements equal to 10^{-3} a.u. for all modes.

We use the Force-Field by Lee, Martin and Taylor [73] which takes into account the symmetry relations of cubic and quartic force constants. [74, 75] The same PES and the hierarchical subspace optimization with the Jacobi method was employed in a previous work of the group.[49] For this system PG-EA successfully converges with the constraint $\bar{D} \leq 7$, which leads to the optimal couple of subspaces (2,5)(1,3,4,6,7,8,9), with a fitness score of about 0.71. The three subspaces (1)(2,3)(4,5,6,7,8,9) were selected in a previous work of the group [49] using the Jacobi criterion but looking for optimal subspaces with a brute force hierarchical approach and constraining the largest subspace to be six dimensional. These three subspaces have an associated fitness score of about 0.91. For methane PG-EA converges using $m = 100$ chromosomes, $\eta = 0.4$, $\mu = 0.01$ and 50 epochs. The Likelihood plot is represented in panel (a) in the left part of figure 3.6. As shown in Fig. 3.6 (panel (b)), the 2-mode approximation leads, in this case, to a poor subspace separation: the best branch in the dendrogram is the colored (1,3,4)(2,5,6)(7)(8)(9) with a score of 4.84, leading to a bigger error than PG-EA for the Jacobian factorization.

Methane, in absence of a preliminary adiabatic switching sampling,[57] is known to be characterized by highly chaotic dynamics, [73] thus we employed 180000 trajectories

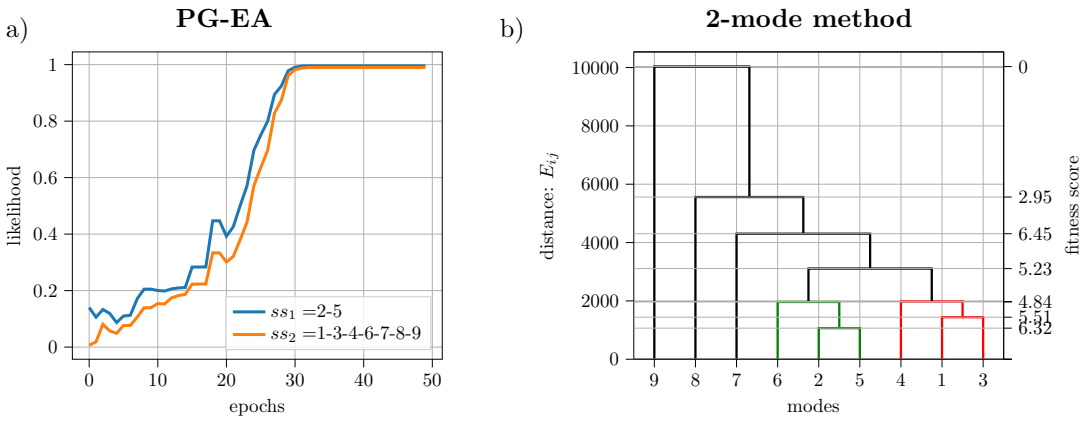


Figure 3.6: Likelihood of the optimal subspaces of methane during PG-EA optimization (panel (a)) and dendrogram for separation with 2-mode approximation method (panel (b)). Note that the 2-mode approximation leads to a very different result.

to provide convergent results, with a rejection rate of about 90%, keeping nearly 2000 trajectories per degree of freedom. Fig. 3.7 reports both the full-dimensional spectrum (red continuous line) and the partial dimensional ones (green and blue), according to the PG-EA vibrational space sub-division found above. All spectroscopic features are properly reproduced. Note that degenerate modes belonging to different subspaces give rise to spectral lines at the same energy (see for instance 1_1 and 2_1 signals displayed in both subspace spectra in Fig. 3.7). In these cases, we consider more accurate the peaks appearing in the largest subspace, as more mode interactions are taken into account, even if frequencies of degenerate modes in different subspaces are very similar and cannot be distinguished by naked eye. In Fig. 3.7 we employ an incremental notation for the spectral features, so that degenerate signals are collected together under the same label. For a deeper insight, we report in Tab. 3.1 the value in wavenumbers of each spectral peak frequency.

In conclusion, PG-EA provides a subdivision of the vibrational space appropriate for DC-SCIVR spectroscopic calculations and we can move to apply it to larger systems where previous recipes for vibrational space subdivision are impracticle.

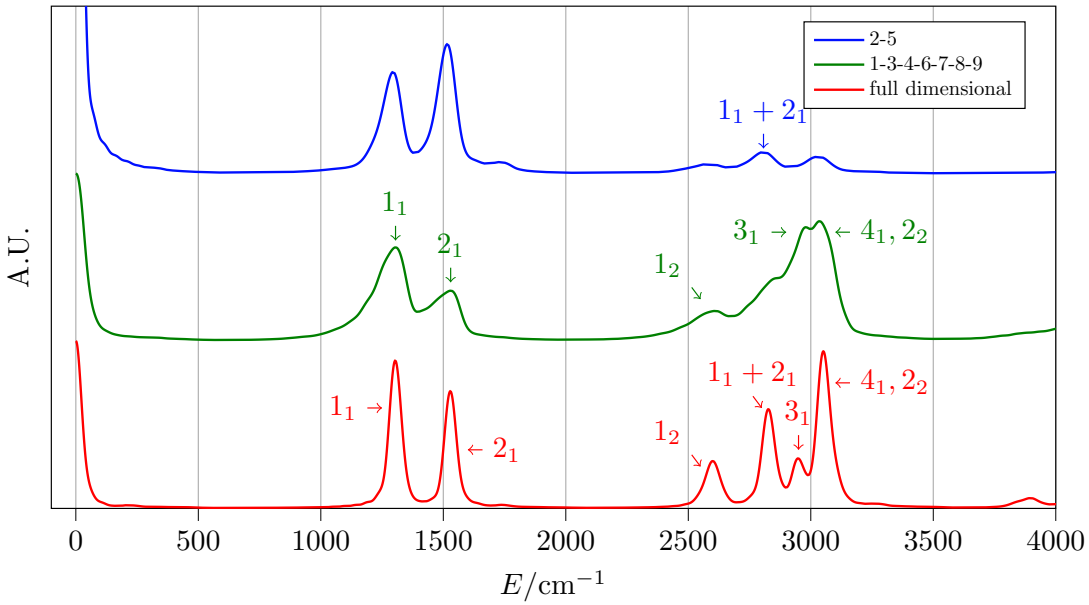


Figure 3.7: Spectrum of methane: the full-dimensional spectrum is in red; the reduced dimensionality spectra chosen according to PG-EA are in green and blue.

3.3.3 *Trans*-N-Methylacetamide

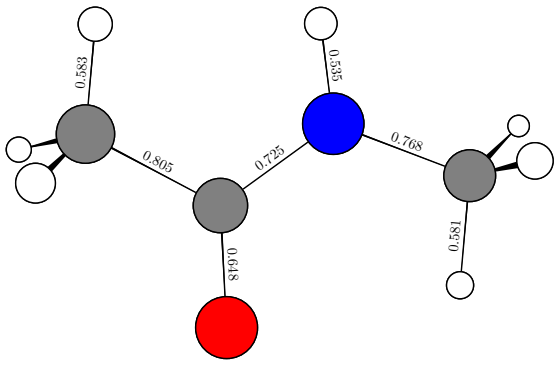


Figure 3.8: *trans*-N-Methylacetamide equilibrium geometry on the Potential Energy Surface by Nandi, Qu, and Bowman.[77] Representative bond lengths of the equilibrium geometry are shown in Ångström. The HNCO peptide bond is of fundamental importance for the dynamics of peptides.

Here we present the subspace optimization and the associated DC-SCIVR spectroscopic calculations for the 30-mode *trans*-N-Methylacetamide (NMA) molecule represented in Fig. 3.8. NMA has been studied thoroughly both computationally [78–80] and experimentally [78, 80–83], as it is one of the simplest examples of a molecule featuring the HNCO peptide bond. We use the full-PES by Nandi, Qu and Bowman[84], which has been designed both for *cis*- and *trans*- NMA and that accounts for the three-fold

Table 3.1: Quantum frequencies of vibration of the methane molecule in cm^{-1} calculated on the PES by Lee, Martin and Taylor [73] using full-dimensional TA SCIVR, DC-SCIVR based on PG-EA subspace partition, and discrete variable representation calculations (Exact). MAE stands for Mean Absolute Error, calculated using exact (MAE (Exact)) or full-dimensional semiclassical values (MAE (TA SCIVR)) as reference. In the fourth column, we report the DC-SCIVR frequencies obtained from the subdivision proposed in Ref.[50] where a different approach for the Jacobi method was employed.

Incremental label	Modes (symmetry)	Exact [76]	TA SCIVR	DC SCIVR (1)(2,3)(4-9)[50]	DC SCIVR PG-EA (2,5)(1,3,4,5-9) [sub][a]
1_1	1,2,3 (F_2)	1313	1304	1287	1305 [G]
2_1	4,5 (E)	1535	1529	1534	1530 [G]
1_2	1,2,3	2624	2600	2562	2610 [G]
$1_1 + 2_1$	1,2,3,4,5	2836	2827		2807 [B]
3_1	6 (A_1)	2949	2948	2960	2980 [G]
4_1	7,8,9 (F_2)	3053	3051	3044	3036 [G]
2_2	4,5	3067	3051	3044	3036 [G]
MAE (Exact)			9.6	22.0	19.3
MAE (TA SCIVR)				14.3	13.4

[a]subspace from which the wavenumber is taken: G for the 7-D green one and B for 2-D blue one with reference to Figure 3.7

symmetry of the methyl rotors. The PES is permutationally invariant and was fitted to thousands of ab initio calculated energies and gradients at B3LYP/cc-pVDZ level of theory.[77] In this case we can compare our DC-SCIVR spectroscopic results with harmonic frequencies and gas phase IR and Raman experimental values.[81]

The two methyl rotational frequencies, i.e. the two lowest-frequency normal mode values, are not considered to be part of the vibrational space. Thus, the dimensionality of the vibrational space we consider is 28. We use a simulation time step of 5 a.u. The finite difference displacement of the i-th normal mode for Hessian and gradients is rescaled by $10^{-3}\sqrt{\max(\boldsymbol{\omega})/\omega_i}$,[48] to account for different PES curvatures along each one of the normal mode coordinates. We use the signal obtained from a single 30000 a.u. long trajectory with initial conditions $(\sqrt{\boldsymbol{\omega}}, \mathbf{q}_{eq})$. We do not observe any conformational change from the *trans* to the *cis* potential energy basin during our simulations.

We apply PG-EA, the 2-mode interaction method, and the Hessian method[50] to divide the vibrational space into subspaces. The results are quite different. For PG-EA, we run a thorough optimization using 10000 epochs, $m = 300$ chromosomes, $\mu = 0.01$ and $\eta = 0.4$, with the largest subspace constraint set to $\bar{D} = 15$ and we obtain the following three subspaces: **A** = (3,5,7,10,11,12,15,21,23,26,28,30), **B** = (4,9,14,16,17,18,19,22,24,25,27,29) and **C** = (6,8,13,20). The fitness score of the chromosome, i.e. the score of the Jacobian factorization, is 1.96. When applying the basic average Hessian criterion[50] with a corse-graining

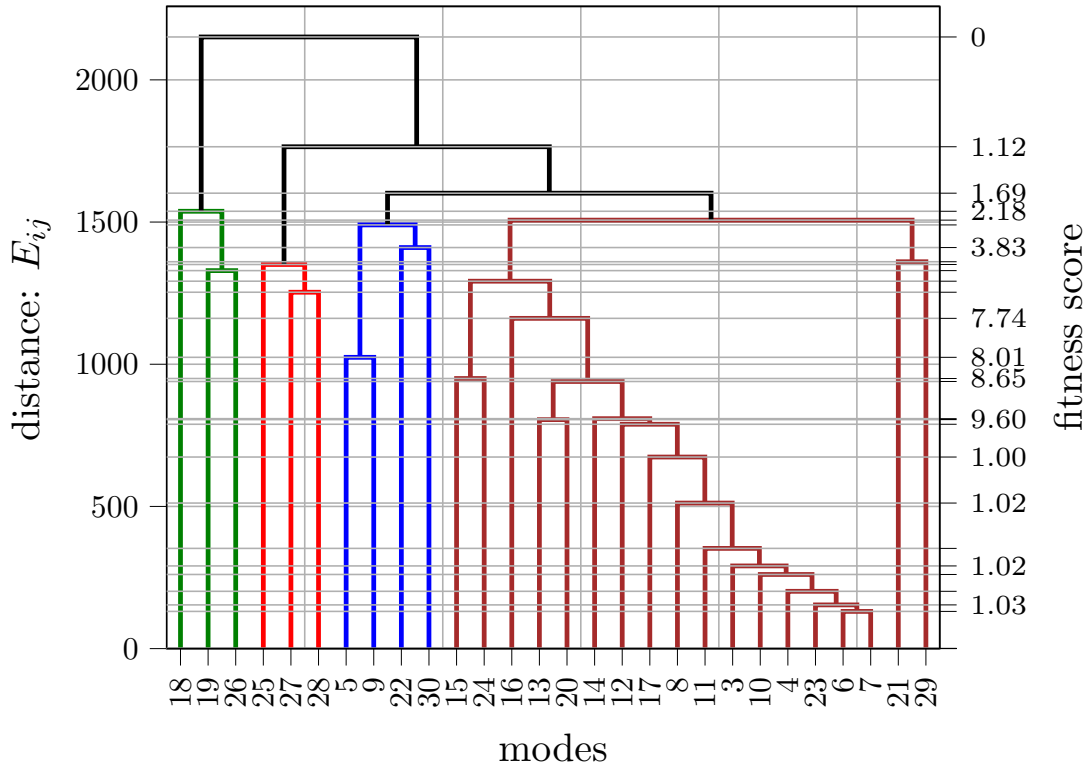


Figure 3.9: 2-mode interaction dendrogram of *trans*-NMA. Four subspaces are generated with a Jacobian factorization score equal to 2.18. As an alternative, one might prefer the five-subspace option with an error of 3.83. We do not show the likelihood vs epochs plot, since the convergence of the whole population is not achieved nor desired. Furthermore computing the likelihood in Eq. 3.7 for a 12 dimensional subspaces is not feasible as it requires the computation of 12^{10} products.

parameter equal to $8 \cdot 10^{-6}$, we obtain the following subspaces: a =(3,5,7,22), b =(10,11,13,16,17,18,20,24,25,26,27,28,29), c =(4), d =(6), e =(8), f =(9), g =(12), h =(14), i =(15), j =(19), k =(21), l =(23), m =(30), which can be associated to a fitness score equal to 4.49. Finally, we apply the 2-mode dendogram approach and obtain the following subspaces: α =(3,4,6,7,8,10,11,12,13,14,15,16,17,20,21,23,24,29), β =(5,9,22,30), γ =(18,19,26), δ =(25,27,28) with a fitness score of 2.18. The 2-mode interaction method produced the dendrogram reported in Figure 3.9, which has a slightly worse score than the PG-EA result. However, the presence of an 18-dimensional subspace makes this subdivision not convenient for Monte Carlo phase-space integration convergence. The Hessian method is instead clearly penalized by the many 1-D subspaces found. These considerations suggest that the PG-EA subdivision into three subspaces is indeed the best choice. Based on PG-EA, we calculate the spectra reported in Fig. 3.10, where the different spectral regions are highlighted by different colors according to the experimentalists' denomination. Overall, the DC-SCIVR spectra

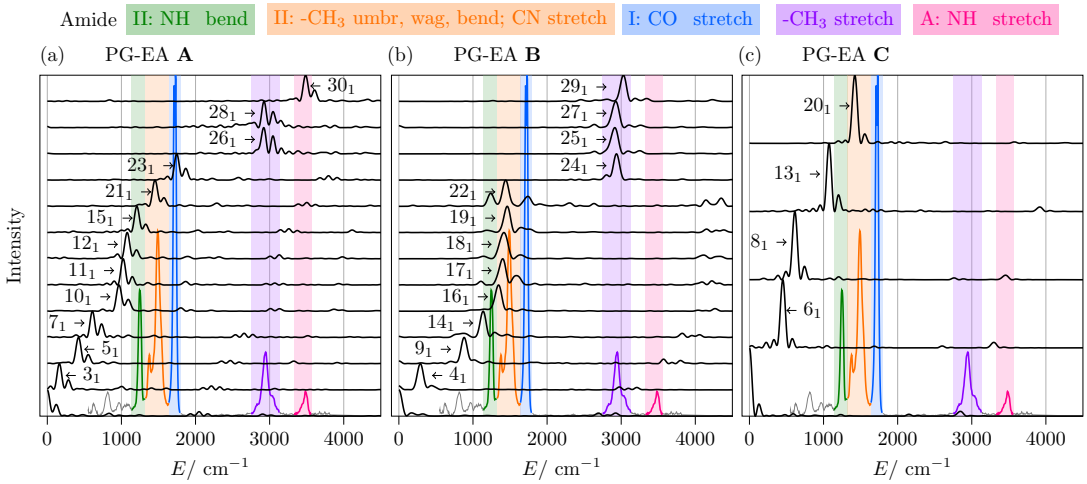


Figure 3.10: Panels (a), (b), and (c) are the three partial spectra for the three **A**, **B**, and **C** PG-EA subspaces. The colored spectrum is the experimental IR spectrum. [85] The black continuous spectra are DC-SCIVR spectra calculated using different ϵ_j value (see Eq.(2.16)) for each normal mode of the subspace. Colored windows are for the different spectroscopic regions as denominated by experimentalists. These windows are commonly labeled Amide II (green and orange), Amide I (blue) and Amide A (pink). The purple window corresponds to CH_3 stretching. Amide III, IV, and V bands are located below $\sim 1200\,\mathrm{cm}^{-1}$ and are very dependent on the side chains.

are reproducing well all the spectroscopic features of this molecule. Actually there are more spectral features in the DC-SCIVR simulation than there are in the experiment because DC SCIVR calculates the power spectrum, which is made of all vibrational levels (that we scale with respect to the zero-point energy), even those associated to transitions which are not IR active. An IR spectrum simulation with related intensities would require calculation of the vibrational eigenfunctions. This feature is not implemented yet in our divide-and-conquer approach, but we are planning to do it soon. In addition, according to our simulations and referring to the experimentalists' denomination, mode 23 is the only one responsible for the Amide I band, subspace A contributes the most to amide III and A bands, while amide II is mostly localized in subspace B.

All these subspace choices produce spectra with almost the same MAEs, as reported in Tab. 3.2. This suggests that the subspace choice is flexible, as well as the choice of the subdivision criterion. However, PG-EA is the method that minimizes the number of subspaces and prevents from having many 1-D subspaces, which could result into a very noisy and not resolved spectrum.[48]

A closer inspection of the vibrational frequency values in Tab 3.2 allows us to better understand the physical meaning of the MAE of the different methods, in particular of the Harmonic versus the DC-SCIVR one. In the case of the harmonic frequencies reported in the second column of Tab. (3.2), 22 out of 26 vibrational frequencies are

higher than the experimental values. Thus, the MAE value of 48 cm^{-1} is because of estimates by excess. In the case of the DC-SCIVR calculations it is the other way around. For example, 20 PG-EA values are underestimating the experimental frequencies and the MAE value of 30 cm^{-1} is given by estimates by defect. Thus, the amount of anharmonic contribution introduced by the DC-SCIVR calculations is on average per mode $\sim 78\,cm^{-1}$. This amount is comparable with the MAE under the column HO/MP2, where harmonic frequencies are calculated at a higher level of *ab initio* theory than the DFT-B3LYP/cc-pVDZ one. These considerations are suggesting that most probably, for this molecule, the discrepancies with respect to the experimental values are mainly due to the DFT level of *ab initio* theory. Conversely, only at a lower degree the inaccuracy can be related to the semiclassical approximation or the quality of the potential energy surface fitting, as previously shown on other systems.[87] Unfortunately, a DC-SCIVR simulation at MP2/aug-cc-pVTZ level of *ab initio* theory is out of reach at time of writing due to its computational burden.

3.4 Summary and Conclusions

We have presented a machine learning algorithm based on a Probability Graph representation and an Evolutionary Algorithm procedure. The algorithm is able to find the best subdivision of the full-dimensional vibrational space into subspaces for model systems in which the best subspace division is known. Our approach is able to preserve Liouville's theorem for each subspace as much as possible and for a given maximum dimensionality of the subspaces. We proved that the clustering provided by PE-GA is indeed one of the possible solutions that minimize the energy exchange between subspaces during the vibrational dynamics, and thus the most convenient for DC-SCIVR and spectroscopic calculations in general. As an alternative, we have proposed a 2-mode coupling scheme which is less computational intense but also less accurate. Application to the DC-SCIVR power spectrum calculation of *trans* N-Methylacetamide is made manageable under Liouville's criterion restrictions only by means of these algorithms. The calculation of the DC-SCIVR power spectrum of trans-NMA with subspace division selected with Liouville's criterion is manageable only employing these two algorithms.

The choice of the PG-EA parameters is arbitrary to some extent. As a matter of fact, the method is bound to look for new solutions at every epoch and hence it will get the global optimum, eventually. However, a sensible choice of the number of chromosomes and the mutation probability may significantly enhance the optimization. Assuming that the number of epochs is fixed, increasing the number of (elite) chromosomes means that the population evolves more slowly, enhancing the chances of eventually hitting the global optimum. However, as the evaluation of the fitness function is the most expensive step, a large population requires significantly more computational

time. Conversely, using a small population means a fast evolution and therefore it is likely to obtain a fast local minimum. We suggest that a sensible choice of the mutation probability is in the interval [0.001, 0.2]. This parameter makes sure that the algorithm does not get stuck in a local minimum, even if the whole population is homogeneous. It becomes less and less important as the pool of possible solutions and the number of elite chromosomes increase.

There are several quantum methods that can take advantage from partitioning the nuclear vibrational degrees of freedom. Clearly, dividing the vibrational space into putative independent subspaces is an approximation. However, if this subdivision is performed according to Jacobi's criterion, it may turn out not to be a rough approximation, especially for high dimensional and loosely connected systems. We believe that, at the affordable cost of a single adiabatic classical trajectory with Hessian calculation, the PG-EA algorithm can be useful to assist any method that has to deal with increasing computational costs with system dimensionality but that is able to perform accurate spectroscopic calculations for each subspace independently. This may be the case, for example, of the Local Mode variant of Multimode[88–90] or other semiclassical wavepacket propagation methods developed by other groups.[91–93]

The work we have presented provides a rigorous rationalization of the simplification of a larger dimensional problem into a set of lower dimensional ones. The examples illustrated in the paper demonstrate that reliable spectroscopic results are obtained if a rigorous strategy is employed to get to a reasonable subspace partition while a non-educated, unwise choice of subspaces may lead to inaccurate or unreliable results. Our algorithms might serve as a powerful tool for advancing the computational spectroscopy of large molecules.

Table 3.2: Vibrational fundamental frequencies for *trans*-N-Methylacetamide (NMA). The first column denominates the vibrational modes. In the second column the fundamental frequencies in harmonic approximation (HO) are reported. DC SCIVR fundamental frequency of vibrations are obtained on the basis of subspace partition by means of PG-EA (third column), Hessian method (Hess, fourth column), and 2-mode interaction method (2-mode, fifth column). The results are sorted by increasing value of the harmonic frequencies and assigned by comparing the associated vibrational motion to the corresponding experimental description. Superscripts refer to the subspace each mode belongs to. The sixth column reports the *ab-initio* harmonic frequencies at the MP2/aug-cc-pVTZ level of theory (HO/MP2) .[86] All data are compared by calculating the Mean Absolute Error (MAE) with respect to the experimental values (last column, Exp) by Ataka, Takeuchi and Tasumi.[81]

	Frequency $^{\text{subID}}$ / cm^{-1}					
modes #	HO	DC SCIVR PG-EA (3 subs)	DC SCIVR Hess (12 subs)	DC SCIVR 2-mode (4 subs)	HO/MP2[86]	Exp [81]
3	150	163 ^{A}	159 ^{a}	156 $^{\alpha}$	151	
4	290	289 ^{B}	285 ^{c}	283 $^{\alpha}$	259	279
5	393	421 ^{A}	417 ^{a}	416 $^{\beta}$	347	429
6	433	451 ^{C}	451 ^{d}	448 $^{\alpha}$	423	439
7	621	609 ^{A}	608 ^{a}	606 $^{\alpha}$	630	619
8	629	613 ^{C}	611 ^{e}	612 $^{\alpha}$	633	658
9	866	881 ^{B}	875 ^{f}	871 $^{\beta}$	883	857
10	995	967 ^{A}	970 ^{b}	972 $^{\alpha}$	1003	980
11	1038	1024 ^{A}	1028 ^{b}	1025 $^{\alpha}$	1058	1037
12	1112	1078 ^{A}	1080 ^{g}	1080 $^{\alpha}$	1119	1089
13	1132	1075 ^{C}	1069 ^{b}	1072 $^{\alpha}$	1169	
14	1166	1137 ^{B}	1138 ^{h}	1134 $^{\alpha}$	1195	1168
15	1260	1208 ^{A}	1210 ^{i}	1205 $^{\alpha}$	1290	1266
16	1391	1345 ^{B}	1344 ^{b}	1347 $^{\alpha}$	1402	1370
17	1415	1401 ^{B}	1388 ^{b}	1388 $^{\alpha}$	1460	1419
18	1434	1418 ^{B}	1409 ^{b}	1415 $^{\gamma}$	1487	1432
19	1474	1462 ^{B}	1463 ^{j}	1461 $^{\gamma}$	1494	1446
20	1485	1422 ^{C}	1426 ^{b}	1430 $^{\alpha}$	1499	1432
21	1491	1453 ^{A}	1453 ^{k}	1453 $^{\alpha}$	1529	1472
22	1550	1441 ^{B}	1444 ^{a}	1472 $^{\beta}$	1561	1511
23	1772	1774 ^{A}	1747 ^{l}	1745 $^{\alpha}$	1749	1707
24	3019	2936 ^{B}	2934 ^{b}	2920 $^{\alpha}$	3088	2915
25	3040	2914 ^{B}	2923 ^{b}	2919 $^{\delta}$	3091	2958
26	3072	2920 ^{A}	2903 ^{b}	2908 $^{\gamma}$	3165	2916
27	3125	2926 ^{B}	2933 ^{b}	2924 $^{\delta}$	3188	3008
28	3126	2924 ^{A}	2912 ^{b}	2912 $^{\delta}$	3188	3008
29	3137	3030 ^{B}	3044 ^{b}	3037 $^{\alpha}$	3197	2973
30	3630	3485 ^{A}	3484 ^{m}	3485 $^{\beta}$	3703	3498
MAE Exp	47.9	30.0	29.7	27.5	78.5	

Chapter 4

Numerical Approximations of the Hessian[1]

In this chapter we summarize some of the attempts to reduce the computational overhead of numerical techniques that require the potential Hessian matrix. We also present the use of the Neural Gas algorithm[94] for the task and its application to the TA-SCIVR method. [95] The developed method is then applied to the semiclassical vibrational spectroscopy of a 46 atoms peptide.

4.1 Introduction

In standard molecular dynamics (MD) simulations, the atomic positions, velocities and forces are evolved in time according to Hamilton's equations and calculated at each time-step. The physical interpretation of the atomistic details provided by dynamics simulations is very powerful and finds uncountable applications everyday. However, if one looks for a deeper physical insight that requires information about the potential curvature, it becomes necessary to evaluate the Hessian (second order derivatives of the potential energy) matrix at each time-step. Specifically, Hessians are employed for higher than second order MD time-integrators, [96–98] for geometry optimization calculations, [99, 100] for instantaneous normal mode analysis, [101, 102] for accurate force fields constructions, [103] for semiclassical dynamics, [104] and other applications, such as reaction rate constants with the instanton method. [105, 106] While integrating Hamilton's equations of motion is doable for any number of degrees of freedom assuming that the interacting potential is readily available as well as that there is suitable computational power, computing properties that depend on the second or even higher

[1] This chapter is a selection with minor modifications of the contents of the paper **Michele Gandolfi**, and **Michele Ceotto**, "An Unsupervised Machine Learning Neural Gas Algorithm for Accurate Evaluations of the Hessian matrix in Molecular Dynamics"

coordinate derivatives of the potential is a challenging task, since these calculations usually scale polynomially with the system size. The task may become prohibitive in *ab initio* molecular dynamics,[107] where the potential and its derivatives are evaluated on-the-fly, i.e. by solving the electronic structure problem and using the Hellman-Feynman theorem, or by finite difference formula using the forces or the potential. To address this issue, a number of approximate methods have been introduced. [108–111] Usually these are of the type of updating schemes, where the Hessian is approximated in a step-wise fashion using the latest information available.[112] These updating schemes were originally developed for optimization [112–116] (see also references therein), but have much evolved and improved since then. Later, they have been employed in various algorithms for direct dynamics simulations. [96–98] For example, the Broyden method is based on a first-order Taylor expansion, which is equivalent to the quasi-Newton methods employed in optimization processes. However, in *ab initio* MD, a higher accuracy is desirable, as it has been shown how a highly accurate Hessian approximation can attain high simulation quality.[108] More recently, Denzel and Kästner[117] followed another route to face the problem, which is to use the Gaussian Process Regression method [118–120] to generate a local fit of the potential surface (GPR-PES), possibly using Hessians as fitting variables. Then, the GPR-PES can be differentiated analytically as many times as required, providing accurate Hessian matrices. The method has been successfully employed in various applications ranging from accurate instanton calculations[121] to the modelling of molecular, amorphous materials and surfaces (see Ref.[120] and references therein), to mention some. However, the GPR-PES method (including Hessian estimation) was intended to give an accurate description of only a local region of the PES. Hence, it is unsuited for extensive molecular dynamics simulations. Furthermore, the GPR fitting time and memory usage scale unfavorably with system size and the method is not recommended for systems with more than 100 degrees of freedom, as the authors pointed out.[117]

In this work, we take a different strategy from those described above for the Hessian approximation. The idea is to assign the same Hessian matrix to a group of MD trajectory configurations that are characterized by similar geometric properties. Since the Hessian ultimately depends on the potential energy surface, we think that the collection of molecular coordinates is an appropriate set of variables to combine a group of configurations, given that the Hessian is uniquely defined for each set of atomic coordinates. Specifically, we employ the unsupervised machine learning algorithm "Neural Gas" to clusterize similar coordinates. The Neural Gas (NGas) algorithm was originally proposed as a Self-Organizing-Map (SOM) or Self-Organizing-Network by Martinetz *et al.*[94] in 1991, with the objective of learning the dimensions and topology of a generic manifold of simple geometrical shapes and complicated time series.[122] The

algorithm devised by Martinetz, Berkovich and Schulten features a number of landmark coordinates, called neurons that are initialized nearby the objective manifold either randomly or according to some rule. Then, the neurons gradually adapt and connect to best represent the manifold shape, thus arranging in the manifold as an approximate time-series. For the adaptation to be effective, the algorithm iteratively drags the neurons closer to the manifold in a way to minimize a given error function, which can be for example the sum of the Euclidean distances of each neuron from the collection of events in the time series. As a final result, the manifold is divided into optimal domains, the Voronoi cells, one for each reference neuron. An intermediate step of the algorithm is

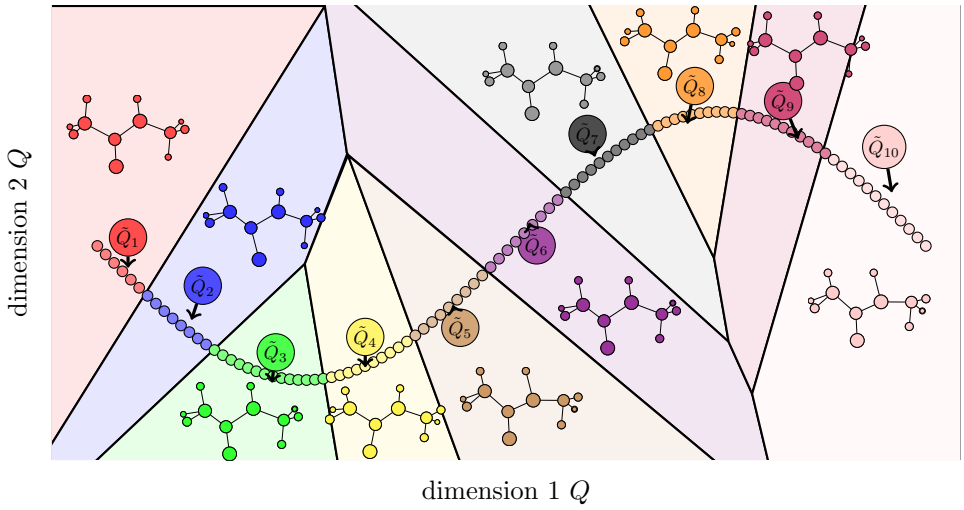


Figure 4.1: Pictorial representation of neurons adaptation for a MD trajectory manifold in a convenient 2-D plane. The trajectory configurations are represented by the collection of small circles and the neuron positions are the larger circles labeled by $\tilde{Q}_i$ for the i-th neuron. The arrows at the neuron circles represent the updating coordinate direction. The domains of each neuron are bordered by solid lines and identified by different colors. All geometries within the same domain share the same Hessian matrix, which is the one calculated at the neuron location.

pictorially represented in Figure 4.1, where the time series of configurations is reported as a line of small circles and the neuron locations as larger circles. The molecular geometry of each neuron is pictorially represented. The collection of the trajectory configurations that are related to each neuron are distinguished by a color code and the neuron domains are bordered by continuous black lines. In few words, molecular geometries of the same color share the same Hessian, which is the one calculated at the corresponding neuron geometry. The time series of configurations which form the manifold can be generated by many trajectories as well. The procedure avoids any redundancy that a multiple trajectory time-series may generate when trajectories have crossing paths. This algorithm will take advantage of the fact that bound system

trajectories are subject to visit the same phase space neighborhood several times during the dynamics, a point which is missed by the Hessian update schemes (vide infra). [111] As a matter of fact, neurons undergo a competitive behavior in getting closer to larger portions of manifold conformations and more probable phase space regions will exhibit higher densities of neurons. Also, we expect that, as shown in Figure 4.1, for a curved trajectory the optimal neuron location would be nearby the center of curvature, which is equally representative of the curvature geometries. Instead, when the trajectory lays on a straight line, the optimal neuron location would be on top of the trajectory. A modified version of the NGas algorithm was introduced in 1994 by Fritzke. [123] In this version it is not required to specify the number of input neurons, with new neurons being added as the optimization proceeds. Later on, many groups provided further advances and optimizations on top of the original version, to enforce topology preservation [124, 125] and allow for a better scaling and optimal growth with increasing amount of data. [125–127]

The use of supervised and unsupervised machine learning algorithms for molecular modelling has a long history in the field of Quantitative Structure-Activity Relationships (QSAR)[128]. Different kinds of molecular descriptors[129] are employed to predict a plethora of properties, especially in the field of medicinal chemistry[130] and drug discovery [131, 132]. In recent years, supervised algorithms were recognized as powerful tools in the formal field of theoretical physical chemistry (see Ref. [133] for an insightful perspective), also for molecular dynamics simulations [134–136]. In addition, unsupervised algorithms have found successful applications in (al)chemical space exploration and chemical design [137–139].

In this paper, we show that the unsupervised machine learning algorithm "neural gas" can optimally compress the information contained in simple molecular geometries along a molecular dynamics simulation and we use the compressed information to approximate the Hessian matrix. More specifically, in this work, we develop and test a NGas method for Hessian approximation with applications to the computation of vibrational spectra, using the Semiclassical Initial Value Representation (SCIVR) method [104, 140] with the Divide-and-Conquer technique (DC SCIVR) implementation developed by our group.[49] In fact, the bottleneck of SCIVR dynamics is the computation of the Hessian matrix along the trajectories.

4.2 Methods

4.2.1 Compact Finite Difference methods

In previous publications, [109, 110] Ceotto, Zhuang and Hase have presented and showed how to employ a Hessian updating scheme based on a Compact Finite Difference (CFD)

strategy for molecular dynamics simulations. [141–144] The CFD approach allows one to obtain a high-order finite difference approximation of function differentiations without incurring large stencil. This goal is achieved by including differentiated terms at more locations within a "compact" stencil. In this updating scheme the Hessian is estimated by extrapolation. For example, if the molecular dynamics geometry $X_i = (x_{i,1}, x_{i,2}, ..., x_{i,n})$ of n scalar entries is followed by $X_{i'}$ at a later time, the updating scheme $H(X_{i'}) = H(X_i) + \Delta H$ allows one to estimate the Hessian for the later geometry $H\left(X_{i'}\right)$, once ΔH is estimated. Bofill[116] proposed the following update recipe

$$\Delta H = (1-\lambda)\frac{R \otimes R^T}{R^T \cdot \Delta X} + \lambda \left(\frac{\Delta X \otimes R^T + R \otimes \Delta X^T}{||\Delta X||^2} - \frac{R^T \cdot \Delta X}{||\Delta X||^4} \Delta X \otimes \Delta X^T \right), \quad (4.1)$$

where λ is a parameter allowed to vary, $\Delta X = X_{i'} - X_i$,

$$R = 2\left[G(X_{i'}) - G(X_i) - H(X_i) \cdot (X_{i'} - X_i)\right], \quad (4.2)$$

$G\left(X\right)$ is the gradient and $\otimes$ and $\cdot$ are the symbols for outer and inner products of vectors. [108] When $\lambda = 0$, the CFD-symmetric rank-one (CFD-SR1) scheme [114] is derived, while the FD-Power Symmetric Broyden (CFD-PSB) scheme is obtained with $\lambda = 1$, and the CFD-Bofill family schemes is represented by the set of linear combinations between the two. Bofill [116] suggested the following practical value for λ

$$\lambda = 1 - \frac{(R^T \cdot \Delta X)^2}{\|R\|^2 \|\Delta X\|^2} \quad (4.3)$$

which avoids the singularity division by near-zero when R is almost orthonormal to ΔX in the first term of eq 4.1. This choice was reported to be a quite accurate Hessian approximation. [108] We provide both our implementation and the pseudocode in the Supplementary Material.

4.2.2 Dynamical Hessian Database methods

An alternative strategy proposed by our group is to create a dynamical database of Hessians (DBH) and related geometries. [111] The idea is to approximate $H\left(X_{i'}\right) \approx H\left(X_i\right)$ at the molecular dynamics configuration $X_{i'}$, whenever $X_{i'}$ is a geometry close enough to X_i, that is a geometry which has already been saved in a database. Two molecular configurations $X_i = (x_{i,1}, x_{i,2}, ..., x_{i,n})$ and $X_{i'}$ are considered close enough when

$$\sqrt{\frac{\sum_{k=1}^{n} \left(x_{i,k} - x_{i',k}\right)^2}{n}} < \rho, \quad (4.4)$$

or

$$\left|x_{i,k} - x_{i',k}\right| < \rho, \quad \forall k = 1, \ldots, n, \tag{4.5}$$

i.e. their distance is smaller than a given threshold ρ. Eq 4.4 is less strict than Eq. 4.5, thus we adopt the latter (Eq. 4.5) for the simulations presented below. To avoid database search latency time, $\left|x_{i,k} - x_{i',k}\right|$ can be evaluated mode after mode only for those geometries satisfying the threshold condition. If more than one geometry satisfies Eq. 4.5, then the Hessian is approximated by the one associated to the geometry with the smaller difference in Eq. 4.5. The database may be updated step by step during the molecular dynamics simulation or it may be created once from a given trajectory. In both cases, only the Hessians for those geometries which do not satisfy the requirement in Eq. 4.5 are computed and the corresponding entry is saved in the database. This method has been extensively tested. [111] The method has allowed for semiclassical simulation of systems where the computational time would have been otherwise too demanding. More details can be found in Ref. [111]. We provide both our implementation and the pseudocode in the Supplementary Material of this paper.

4.2.3 A Neural Gas Algorithm for Hessian approximation

We borrow from the DBH method the idea that close enough molecular configurations have similar Hessians, but we add the feature that the algorithm is allowed to look for optimal configurations even outside the trajectory pathway. As described above, the idea of the NGas method is to approximate a given set of elements with few representative ones, called neurons. In the case of the set of the Hessian matrices along a classical trajectory, a NGas algorithm would find few geometries whose Hessians can be employed to approximate the Hessian matrix at every configuration along the trajectory.

Notice that all the methods shown in this paper are agnostic with respect to the coordinate system and units. In our case, we usually perform molecular dynamics either in Cartesian or normal mode coordinates. However, we ultimately employ mass-scaled normal mode coordinates for our spectra calculations. To locate the neurons and proceed with the NGas optimization process, we first scale the whole trajectory set of coordinates to fit a cubic box with edge 1. In other words, we map each mass-scaled normal mode coordinates component $q_j\,(t)$ according to the equation:

$$Q_j(t) = \frac{q_j(t) - m_j}{M_j - m_j}, \tag{4.6}$$

where $m_j = \min q_j(t)$ is the minimum value in the time series of the j-th component, $M_j = \max q_j(t)$, and the new coordinates $\mathbf{Q}(t)$ are the scaled coordinates with values between 0 and 1. Once the number of neurons, i.e. the number of the most representative geometries, is chosen, we evenly sample the initial guess $\tilde{\mathbf{Q}}$ for the neuron

positions directly from the set $\mathbf{Q}(t)$ of the trajectory geometries, i.e. initially the set $\left\{\tilde{\mathbf{Q}}\right\} \subset \{\mathbf{Q}(t)\}$. In practice, a fixed number of neurons are initialized on top of the trajectory configurations in normal mode coordinates and distributed at fixed time intervals. In their first presentation of the NGas algorithm, Martinez and Schulten[94] suggested that, at each epoch τ, all the trajectory geometries $\mathbf{Q}_i$ are sampled in random order from the set of trajectory configurations $\{\mathbf{Q}(t)\}$ (without repetition). Every time a configuration $\mathbf{Q}_i$ is sampled, each j-th neuron $\tilde{\mathbf{Q}}_j$ is updated according to following rule:

$$\tilde{\mathbf{Q}}_j = \tilde{\mathbf{Q}}_j + \alpha(\tau)e^{-K_{ij}/\lambda(\tau)}(\tilde{\mathbf{Q}}_j - \mathbf{Q}_i) \tag{4.7}$$

where K_{ij} is an integer number that ranks the distance between the trajectory scaled coordinate geometry $\mathbf{Q}_i$ and the neuron $\tilde{\mathbf{Q}}_j$. Specifically, K_{ij} is equal to 0 for the nearest trajectory geometry and to $(n-1)$ for the furthest one. In eq 4.7, $\alpha(\tau)$ and $\lambda(\tau)$ are parameters which are modeled to decrease during the optimization process. These parameters change for each epochal iteration and they tune the neural gas adaptability, i.e. its ability to expand and how fast this expansion is performed. More specifically, λ tunes the number of neighbor coordinates that can significantly interact with each neuron, while α tunes the adaptability of the neural gas. In other words, the larger is λ, the greater is the number of trajectory geometries that significantly contribute to the updating scheme in eq 4.7, while α tunes how large is the response of $\tilde{\mathbf{Q}}$ and after how many iterations it is still responsive and learning. α and λ are updated at each epoch with the same rule[122]

$$g(\tau) = g_{init}\left(\frac{g_{final}}{g_{init}}\right)^{\tau/\tau_{max}} \tag{4.8}$$

with g_{init} and g_{final} being parameters. Reasonable choices for these parameters are $\alpha_{init} = 0.3$, $\alpha_{final} = 0.05$, $\lambda_{init} = 30$, $\lambda_{final} = 0.01$, independently of the simulated system.[122]

The updating formula in eq 4.7 can also be written using the $\mathcal{U}_{ij}$ operator formalism that we introduce here

$$\begin{aligned} \mathcal{U}_{ij}\tilde{\mathbf{Q}}_j &= \tilde{\mathbf{Q}}_j\left(1 + \alpha(\tau)\, e^{-K_{ij}/\lambda(\tau)}\right) - \alpha(\tau)\, e^{-K_{ij}/\lambda(\tau)}\mathbf{Q}_i \\ &= \tilde{\mathbf{Q}}_j\left(1 + A_{ij}(\tau)\right) - A_{ij}(\tau)\,\mathbf{Q}_i \end{aligned} \tag{4.9}$$

where $\alpha(\tau)$ and $\lambda(\tau)$ have been grouped into one parameter $A_{ij}(\tau) = \alpha(\tau)\, e^{-K_{ij}/\lambda(\tau)}$, which depends on α, λ, and K_{ij}. $A_{ij}(\tau)$ has the form of a Boltzmann factor with temperature $\lambda(\tau)$ and it is interpreted as a kind of neuron "influence probability". As the NGas training goes on, $\lambda(\tau)$ (the analogous of temperature) decreases and the gas freezes nearby the trajectory. Assuming that we know beforehand all the K_{ij} coefficients

for the motion of the neuron $\tilde{\mathbf{Q}}_j$ (in general we do not), by applying the $\mathcal{U}_{ij}$ operator of eq 4.9 for N_{steps} time-steps, i.e. for the whole set of trajectory points $\{\mathbf{Q}(t)\}$, in a random order and without repetition, one gets:

$$\begin{aligned}\mathcal{U}_{p(i,1),j}\mathcal{U}_{p(i,2),j}\ldots\mathcal{U}_{p(i,n),j}\tilde{\mathbf{Q}}_j &= \tilde{\mathbf{Q}}_j\prod_{u=1}^{N_{steps}}\left(1+A_{p(i,u)j}\right)-\sum_{u=1}^{N_{steps}}\mathbf{Q}_{p(i,u)}A_{p(i,u)j}\prod_{v=u+1}^{N_{steps}}\left(1+A_{p(i,v)j}\right)\\ &= \tilde{\mathbf{Q}}_j\prod_{u=1}^{N_{steps}}B_{iuj}(\tau)-\sum_{u=1}^{N_{steps}}\mathbf{Q}_{p(i,u)}\left(B_{iuj}(\tau)-1\right)\prod_{v=u+1}^{N_{steps}}B_{ivj}(\tau) \quad (4.10)\end{aligned}$$

where $p(i,\mathrm{u})$ is the u-th element of a random permutation of the index i over the list of the first N_{steps} natural numbers and we wrote the matrix with permuted indices as a three indices tensor: $B_{iuj}(\tau) = 1 + A_{p(i,u)j}(\tau)$. Eq 4.10 accounts for the core functionality of the neural gas algorithm. In eq 4.10 the first term on the right hand side does not depend on the trajectory position, but only on the trajectory distance ranking from the j-th neuron $\tilde{\mathbf{Q}}_j$, in the form of the K_{ij} coefficients. Instead, the second term in eq 4.10 depends on the trajectory position $\mathbf{Q}_{p(i,u)}$ both explicitly and implicitly through K_{ij}. Eventually, when λ tends to 0 as described above, also A_{ij} goes to 0 and the sums and products in eq 4.10 converges to a final neuron position with less than N_{steps} terms. Within the analogy of the neural gas, we would say that in the case when the gas is cold, it is influenced only by the local manifold (nearby trajectory points) rather than by the whole environment (the entire trajectory points), even if all the trajectory configurations are summed by the index u in eq 4.10. However, even if eq 4.10 has been introduced to better understand the physics of the neural gas iterations, eq 4.7 is employed in practice. According to these equations, the neural gas process is a competitive type of learning, since neurons compete to be nearest as possible to the trajectory geometries. This competitiveness is encoded in the parameters K_{ij}, which may change every time a neuron is moved and make impossible to use eq 4.10 straightforwardly.

Once the gas is frost, we perform a further optimization of each neuron position $\tilde{\mathbf{Q}}_j$. First, we consider that for each trajectory point $\mathbf{Q}_i$, there is only one nearest neuron position $\tilde{\mathbf{Q}}_j$. Then, we collect all these points into a set $\{\mathbf{Q}(t)\}_j$ which is the collection of trajectory segments nearest to the j-th neuron, and there will be as many sets of this type as the number of neurons. Eventually, we can estimate the error $E(\tilde{\mathbf{Q}}_j)$ to consider the neuron $\tilde{\mathbf{Q}}_j$ at the place of the trajectory segments $\mathbf{Q}(t)$ as the line integral

$$E(\tilde{\mathbf{Q}}_j) = \frac{1}{V}\int_{\{\mathbf{Q}(t)\}_j}\left(\tilde{\mathbf{Q}}_j-\mathbf{Q}(t)\right)^2 ds \quad (4.11)$$

where $V = \int_{\{Q(t)\}_j} ds$ is a normalization constant and ds is the integration line segment. Now, we can locate $\tilde{\mathbf{Q}}_j$ such that $E(\tilde{\mathbf{Q}}_j)$ is minimal. The first order derivative of $E(\tilde{\mathbf{Q}}_j)$

respect to each $\tilde{\mathbf{Q}}_j$ is

$$\begin{aligned}
\nabla_{\tilde{\mathbf{Q}}_j} E(\tilde{\mathbf{Q}}_j) &= \frac{2}{V} \int_{\{\mathbf{Q}(t)\}_j} \left(\tilde{\mathbf{Q}}_j - \mathbf{Q}(t) \right) ds \\
&= 2 \left(\tilde{\mathbf{Q}}_j - \frac{1}{V} \int_{\{\mathbf{Q}(t)\}_j} \mathbf{Q}(t) ds \right) \\
&= 2 \left(\tilde{\mathbf{Q}}_j - \langle \mathbf{Q}(t) \rangle_j \right). \qquad (4.12)
\end{aligned}$$

Eq. 4.12 is equal to 0 when $\tilde{\mathbf{Q}}_j$ is equal to the "center of mass" $\langle \mathbf{Q}(t) \rangle_j$ of the trajectory segments $\{\mathbf{Q}(t)\}_j$. Hence, we implemented into the algorithm this further optimization step, such that each neuron is eventually placed at the center of mass with respect to the trajectory points associated to that neuron.

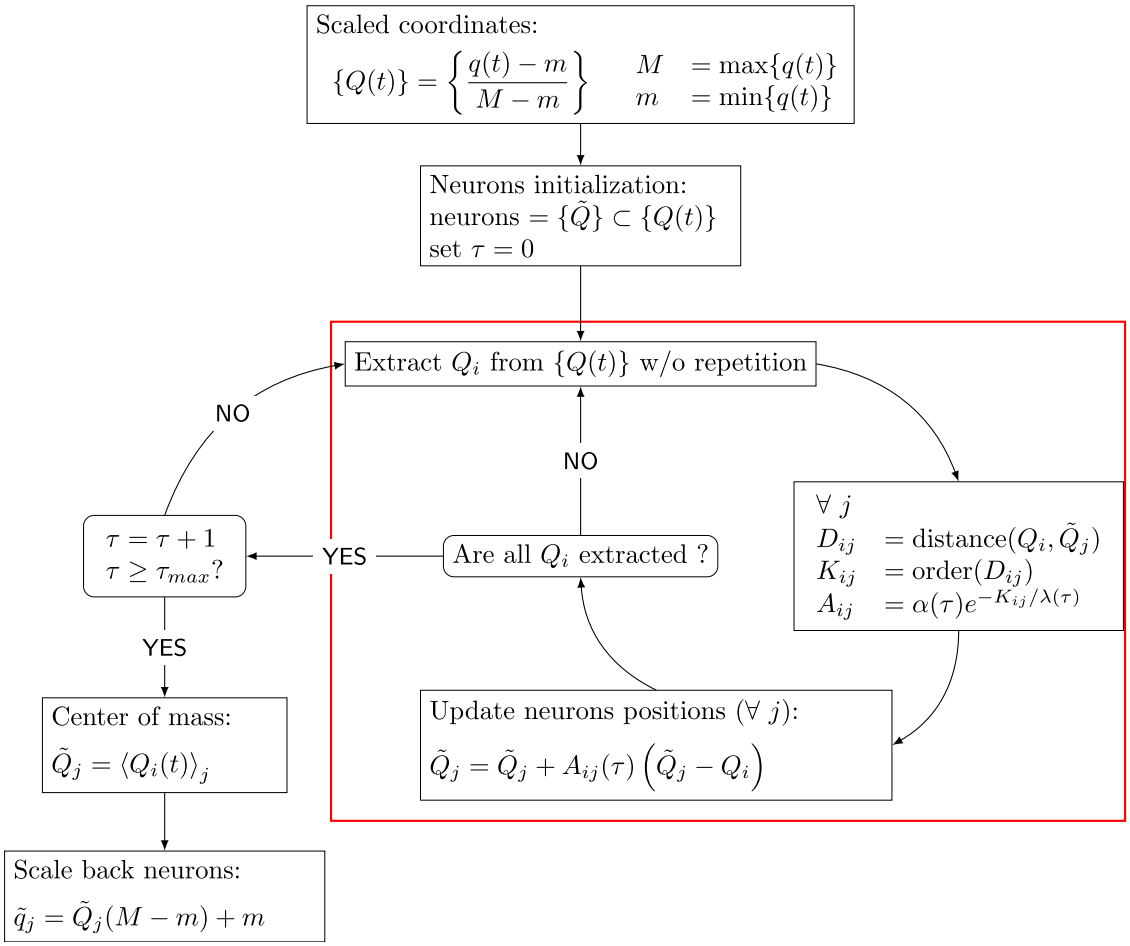


Figure 4.2: Flow diagram of our Neural Gas implementation. Neurons are sampled from the scaled coordinates and iteratively optimized according to the cyclic part of the diagram. Every time a coordinate $\mathbf{Q}_i$ is sampled, one needs to compute its distance from every neuron to determine the ordering (encoded in the $\mathbf{K}$ matrix). Once the training is done, the neurons are scaled back to their original normal mode or Cartesian form. A red rectangular frame delimits the core part of the NGas algorithm, where neurons are updated in competition with one another to get closer to the trajectory.

Figure 4.2 reports the flow diagram of the algorithm described above, with the core part of the algorithm enclosed by the red rectangular frame. Apart from the scaling and the final optimization steps, the algorithm can be traced back to the first version by Martinetz and Schulten.[94] At each epoch, all trajectory coordinate $\mathbf{Q}_i$ enter in a random order the neural gas optimization cycle, where the distance D_{ij} from its nearest neuron $\tilde{\mathbf{Q}}_j$ is evaluated together with the order coefficient K_{ij}. The epoch step is completed only after all trajectory points have been considered and the related neuron updated according to eq 4.7. For the following epoch, the $A_{ij}(\tau)$ is updated and so on. At the end of the epoch evolution, each j-th neuron coordinate is placed at the center of mass of the collection of trajectory points that are nearest to that neuron

$\{\mathbf{Q}(t)\}_j$. The new location $\tilde{\mathbf{Q}}_j$ is then transformed back into the original trajectory coordinate system of reference, $\tilde{\mathbf{q}}_j$, that can be either Cartesian or normal mode ones. Recently, an algorithm[145] that uses the idea of dividing the configuration space in Voronoi cells (as in the NGas method) has been proposed . The algorithm creates an *on-the-fly* updated mesh to approximate the potential energy from previous potential and potential gradient evaluations.

We evaluate the quality of the approximation as the mean absolute error of the Cartesian Hessian matrix elements

$$\sigma_{Hess} = \frac{1}{N_{steps} N_{cart}^2} \sum_{k}^{\mathrm{N_{steps}}} \sum_{i}^{N_{cart}} \sum_{j}^{N_{cart}} \left| H_{ij}(k) - H_{ij}^{\mathrm{approx}}(k) \right| \tag{4.13}$$

where N_{cart} is the number of Cartesian coordinates, N_{steps} the number of molecular dynamics time steps, $H_{ij}(k)$ is the entry of the exact Hessian matrix and $H_{ij}^{\mathrm{approx}}(k)$ the approximated one, both at step k. We provide both our implementation and the pseudocode in the Supplementary Material.

4.3 Results

In this Section we present simulations of growing complexity, starting from the small molecular systems H_2O, HCOH and CH_4, going to the smallest prototype of peptide bond (*trans* N-methylacetamide), up to a small synthetic peptipe (N-acetyl-L-phenylalaninyl-L-methionine amide), which is composed of 46 atoms and 132 vibrational degrees of freedom. All simulations consist of a single 3000 time-step constant energy (NVE) classical trajectory with a 10 a.u. constant time-step. The initial conditions are chosen according to the MC-DC-SCIVR recipe described above. The classical equations of motion are integrated using a four-order symplectic integrator.We employed pre-computed potential energy surfaces (PES) for H_2O, [146] HCOH, [54] CH_4, [73] and NMA, [84] while the N-acetyl-L-phenylalaninyl-L-methionine amide (Ac-Phe-Met-NH_2) molecule was simulated on-the-fly by direct *ab initio* molecular dynamics at the level of DFT-B3LYP-D/6-31G* theory. The PES derivatives are computed by finite differences with a fixed displacement of 10^{-3} a.u. In the case of the *trans* N-methylacetamide (NMA) calculation, we employ an analytical gradient PES [84, 147] and the Hessian matrix is computed by finite difference of the gradient. The NGas method has been optimized using 150 neurons for the simulation of H_2O, HCOH, CH_4, NMA and 300 neurons are employed in the case of the Ac-Phe-Met-NH_2. The number of learning epochs and the $\alpha_{init}, \alpha_{final}, \lambda_{init}, \lambda_{final}$ parameters are kept fixed, as described in Sec. 4.2. We performed all the computations reported here on a computer laptop using a single core (Intel(R) Core(TM) i7-4510U CPU @ 2.00GHz, with less than 16 GB of

available memory) with the exception of Ac-Phe-Met-NH_2, whose Hessians have been computed on the group computer cluster, using 10 cores (Intel(R) Xeon(R) CPU E5-2660 v3 @ 2.60GHz 125Gb) per Hessian matrix.

4.3.1 Hessian approximation accuracy

To test the accuracy of each method we performed calculations where 150 Hessians (300 in the case of Ac-Phe-Met-NH_2) out of 3000 time-steps (2500 in the case of Ac-Phe-Met-NH_2) are calculated explicitly, i.e. from the PES or the electronic structure, and the remaining ones are approximated. In others words, exact Hessians are estimated every 20 (about 8 in the case of Ac-Phe-Met-NH_2) molecular dynamics time-steps and all the others are approximated. The deviations of the approximated Hessians from the exact ones is estimated using eq 4.13. Notice that, although the Hessians in eq 4.13 are in Cartesian coordinates, we employ normal mode coordinates in the DBH and NGas methods to locate the optimal configurations.

Table 4.1: Accuracy and computational time for different Hessian approximation methods. First column is the molecule, second column the number of exact Hessian calculations, the third column the Hessian approximation method, the fourth column the error according to eq 4.13, the fifth column the relative error respect to the Neural Gas method, the sixth column the cpu-time for the each method, the seventh column the cpu-time for the exact Hessians evaluation and the last column the total computational time. All times are in seconds, except explicitly indicated. For each molecule the Neural Gas (NGas), the dynamical Hessian database (DBH) at threshold ρ, and the compact finite difference (Bofill) methods are tested. The "all Hessians" label is for Hessian calculations at each time-step, i.e. without any approximation.

molecule	# Hessians	method	$10^2\sigma_{Hess}$	relative σ_{Hess}*	method cpu-time	Hessians cpu-time	Total cpu-time
H_2O	150	NGas	0.539	1.00	19.314		19.51
	150	DBH(ρ=2.59)	0.728	1.35	0.355	0.197	0.55
	150	Bofill	2.336	4.33	0.148		0.346
	3000	all Hessians	0.000	NA	0.000	3.947	3.95
HCOH	150	NGas	0.612	1.00	19.395		19.75
	150	DBH(ρ=8.22)	0.824	1.34	0.329	0.356	0.69
	150	Bofill	1.570	2.56	0.160		0.52
	3000	all Hessians	0.000	NA	0.000	7.115	7.12
CH_4	150	NGas	0.732	1.00	18.923		19.42
	150	DBH(ρ=7.95)	1.000	1.37	0.345	0.492	0.84
	150	Bofill	2.231	3.05	0.192		0.68
	3000	all Hessians	0.000	NA	0.000	9.835	9.84
NMA	150	NGas	0.447	1.00	22.582		35.42
	150	DBH(ρ=21.15)	0.490	1.09	0.499	12.842	13.34
	150	Bofill	0.935	2.09	0.262		13.10
	3000	all Hessians	0.000	NA	0.000	256.834	256.83
Ac-Phe-Met-NH_2	298	NGas	0.059	1.00	27.667	7620.819**	7621.28**
	298	DBH(ρ=11.9)	0.059	1.00	2.697		7620.86**
	312	Bofill	0.153	2.58	2.030	7978.845**	7978.88**
	2500	all Hessians	0.000	NA	0.000	63933.049**	63933.05**

*defined as the error of the method divided by the error of the NGas method

**core hours (average of core hours necessary for the computation)

Table 4.1 reports the results of this test for each molecule and it shows that the computational time of the Hessian matrix calculation is the simulation bottleneck when evaluated by *ab initio* methods. Actually, when using a pre-computed PES, the time

required for the NGas algorithm iterations is roughly of the same order of magnitude of evaluating the Hessian for each trajectory configuration. To appreciate the advantage of the approximation schemes in terms of cpu-time, one has to reach the 30 degrees of freedom of the NMA molecule. However, even in this case, the use of analytical gradients provided by the pre-computed PES [84] accelerates the Hessian matrix estimation and keep the option to evaluate all the Hessians along the trajectory viable. We can see a clear advantage of the approximation methods only when dealing with Ac-Phe-Met-NH_2, which we simulated on-the-fly. In this case, the evaluation of a single Hessian matrix takes about 3 hours with NWChem package [148] on a 10 cores (Intel(R) Xeon(R) CPU E5-2660 v3 @ 2.60GHz 125Gb) node and the NGas and DBH methods become in this case the only viable option. Table 4.1 reports also in the fourth column the error σ_{Hess} for each method with respect to all-Hessian evaluation. We notice that the NGas method is as accurate as the DBH one in the case of Ac-Phe-Met-NH_2. In the fifth column of Table 4.1 the relative σ_{Hess} shows how each method compares with the NGas in terms of accuracy. We see that by using the NGas method we can decrease the error by about 26% for small molecular systems, while the NGas error is comparable with the DBH method in the cases of the NMA and Ac-Phe-Met-NH_2 molecules. We can understand this trend considering that the higher the number of degrees of freedom is, the less often the trajectory visits the same phase space region. In these cases, the NGas method provides a solution that is very similar to the DBH one since neurons are distributed along the trajectory and basically coincide with the molecular geometries at which Hessian matrices are calculated according to the DBH approach.

Overall, we can observe that the ratio of computational time versus the number of degrees of freedom is almost constant for all the methods and it increases moderately only in the case of the Ac-Phe-Met-NH_2 system. This is mainly due to the time required to store and copy the trajectory. At this stage, we can not assert what will happen for even higher dimensional systems. However, we still can test the robustness and stability of each method by decreasing the number of PES or *ab initio* Hessian entries. In this way, we can also better understand which are the minimum number of Hessian evaluations necessary for obtaining an accurate estimate. We focus on the Ac-Phe-Met-NH_2 system and on the NGas and DBH approximations. Table 4.2 reports the values of σ_{Hess} of eq 4.13 for the two methods for the different exact Hessian evaluation times reported in the second column. Clearly, the more ab initio Hessians are computed, the smaller the approximation error is, as reported in the third column. If the NGas and DBH error is comparable for about 200 exact Hessian evaluations, DBH is more and more accurate as the number is significantly reduced down to 25. We think that this poor performance of the NGas method is due to the fact that, given the extremely low numbers of Hessians provided, the neuron locations are not representatives of their

trajectory neighborhood. In other words, when the system conformation is averaged over many ones, the result may be very different from the actual conformations visited along the classical trajectory. To improve and going beyond this limitation, we use an extended set of variables for the neurons' space, which includes also the potential gradients in the NGas training process. While the original neurons are identified by a set of normal mode molecular coordinates of the type $\tilde{\mathbf{q}} = (\tilde{q}_1, \ldots, \tilde{q}_{N_{vib}})$, in the improved version the vector which identifies the neuron includes the energy gradient coordinates as well, $\tilde{\mathbf{q}} \cup \nabla\tilde{\mathbf{q}} = (\tilde{q}_1, \ldots, \tilde{q}_{N_{vib}}, \frac{\partial V(\tilde{q}_1)}{\partial \tilde{q}_1}, \ldots, \frac{\partial V(\tilde{q}_{N_{vib}})}{\partial \tilde{q}_{N_{vib}}})$. This extended neuron set of variables accounts for the PES slope, in addition to the molecular positions. In this way, unrealistic conformations with huge internuclear forces (and consequently large Hessian elements) are excluded in favor of more realistic conformations. The improved results are reported in the last column of Tab. 4.2. The extended NGas is always more accurate not only with respect to the original NGas method, but also to the DBH method, in particular for the cases when there are few exact Hessian estimates. We observe again that when the number of neurons is increased, these are allowed to have a neighboring trajectory segment that is a straight line, thus the NGas and DBH methods become alike.

Table 4.2: Hessian element mean absolute error from eq 4.13 (σ_{Hess}) for the Neural Gas (NGas) and the dynamical Hessian Database (DBH) methods by varying the number of exact Hessian evaluations indicated in the second column in the case of the Ac-Phe-Met-NH_2 molecule. The columns "$\tilde{q}$ " and "$\tilde{q} \cup \nabla\tilde{q}$" refer to different NGas training spaces, as described in the text, while the last column reports the relative error with respect to the best NGas estimate.

method	# Hessians	$10^2\sigma_{Hess}$		relative σ_{Hess}*
		$\tilde{q}$	$\tilde{q} \cup \nabla\tilde{q}$	$\tilde{q} \cup \nabla\tilde{q}$
NGas	25	0.372	0.260	1.00
DBH ($\rho = 55.0$)	25	0.319		1.23
NGas	50	0.357	0.219	1.00
DBH ($\rho = 40.0$)	50	0.267		1.21
NGas	100	0.157	0.153	1.00
DBH ($\rho = 27.3$)	100	0.167		1.09
NGas	200	0.090	0.089	1.00
DBH ($\rho = 17.3$)	200	0.090		1.01

*defined as the error of the method divided by the error of the NGas method

Although the NGas method seems to provide a small improvement respect to the DBH one for the larger systems, i.e. NMA and Ac-Phe-Met-NH_2, we can prove that it

can reach an accuracy comparable to that one observed for the smaller systems. In fact, both NGas and DBH methods approximate only the regions of configurational space that are close to the trajectory, since they are based on the distance from neighboring geometries. Hence, if we employ 150 neurons to approximate a 3000 steps trajectory, each Voronoi cell contains on average 20 geometries, and the related Hessians. When the system becomes larger, we expect these geometries to be visited within the same portion of the trajectory. This is the reason why DBH and NGas methods provide more and more similar results as the system size grows. However, things are different if we use an ensemble of MD trajectories, because in this case it is very likely that trajectories cross and overlap significantly, as in a tangle of strings. Table 4.3 reports the numerical results of two ensembles of trajectories.

Table 4.3: Hessian element mean absolute error of eq 4.13 (σ_{Hess}) for the NMA molecule, using the Neural Gas (NGas) and the dynamical Hessian Database (DBH) methods, obtained by varying either the total number of configurations (second column) and the number of *ab initio* Hessians (third column). The last column reports the relative error with respect to the NGas estimate.

method	Configurations (#trajectories x #steps)	# Hessians	$10^2\sigma_{Hess}$	relative σ_{Hess}*
NGas	100x1000	999	1.33	1.00
DBH (ρ =54.7)	100x1000	1008	1.64	1.23
NGas	500x1000	1000	1.52	1.00
DBH (ρ =67.6)	500x1000	999	1.94	1.28
NGas	100x1000	999	1.33	1.00
DBH (ρ =45.6)	100x1000	2049	1.34	1.01
NGas	500x1000	1000	1.52	1.00
DBH (ρ =47.5)	500x1000	6082	1.56	1.03

*defined as the error of the method divided by the error of the NGas method

The trajectory initial conditions were sampled from the Husimi distribution in phase space centered at the equilibrium values. We notice that the trajectories originated by this distribution are spread in energy values, on the contrary of previous simulations, and the errors in the Hessian matrix are inevitably higher. In the upper part of the Table 4.1, the NGas method provides a smaller value of σ_{Hess}, for the same number of *ab initio* Hessians employed in the DBH simulation. In the lower part of the table, the same average error in the Hessian matrix is reached only when the DBH employs more than six times *ab initio* Hessians than the NGas method. With about 500 thousands geometries and 1 thousands neurons our implementation of the NGas method takes its

toll and the training of the neural gas takes about 7 hours to be optimized, compared to the 50 minutes required by DBH. However, if one takes into account the ~10 minutes per core (Intel(R) Xeon(R) CPU E5-2660 v3 @ 2.60GHz 125Gb) that it takes to compute the *ab initio* Hessian matrix of NMA at DFT-B3LYP/6-31G* level of theory, it is still convenient to use the NGas method. Finally, the σ_{Hess} reported in Table 4.3 are rather large, as we enforced less than one ab initio Hessian matrix every 100 Hessians, which is quite a drastic setup.

4.3.2 Spectroscopic simulations

One may wonder which level of Hessian approximation accuracy is requested in MD applications and how important the choice of the approximation method is. To reply to this question, we decided to employ our approximate Hessians for the integration of eq 2.12 and the calculation of the power spectrum using eq 2.8. Specifically, we simulated the full dimensional vibrational power spectrum of the small Ac-Phe-Met-NH_2 peptide using a single on-the-fly *ab initio* trajectory with our MC-DC-SCIVR method. In the Divide-and-Conquer strategy we need to find a vibrational space subdivision, which is the result of a trade-off between spectroscopic accuracy and feasibility. Too high dimensional vibrational subspaces are not practical, but too low dimensional ones may turn out to be a drastic approximation. For these reasons, we performed a preliminary coarse-graining of the time-averaged Hessian matrix, by fixing to zero all the elements smaller than $8.0 \cdot 10^{-6}$a.u. [50] In this way, after conveniently permuting rows and columns, we obtained a block diagonal matrix whose 23-dimensional subspace contains all the stretching modes of the amine group we are interested in. These are denominated as sNH_2 (mode number 129), NH (II) (130), NH (I) (131), and aNH_2(132). We focus on these fundamentals because their experimental values are available for comparison.[149] This subspace is further decomposed into smaller subspaces using our PG-EA algorithm[51] . The stretches we are interested in are highlighted in bold in the normal mode subspaces {10 30 33 36 37 38 42 46 **130 131**} and {47 105 **129 132**}. The mode numbers are assigned according to the harmonic frequency values, where smaller numbers means lower harmonic frequency values. Both subspaces contain floppy modes. In particular, the first subspace contains several floppy modes and we expect that the partial spectra of the NH (II) (130) and NH (I) (131) modes will embody several combination features of these stretches with floppy modes.

Figure 4.3 shows the power spectra of the selected amide group stretching modes using different Hessian approximations. On each panel is reported the signal of each mode after a suitable combination of coherent states.[68] Continuous lines are for MC-DC-SCIVR simulations where Hessians have been calculated at each time-step and are labeled as "all Hessians". Dashed lines are for our NGas approximation presented

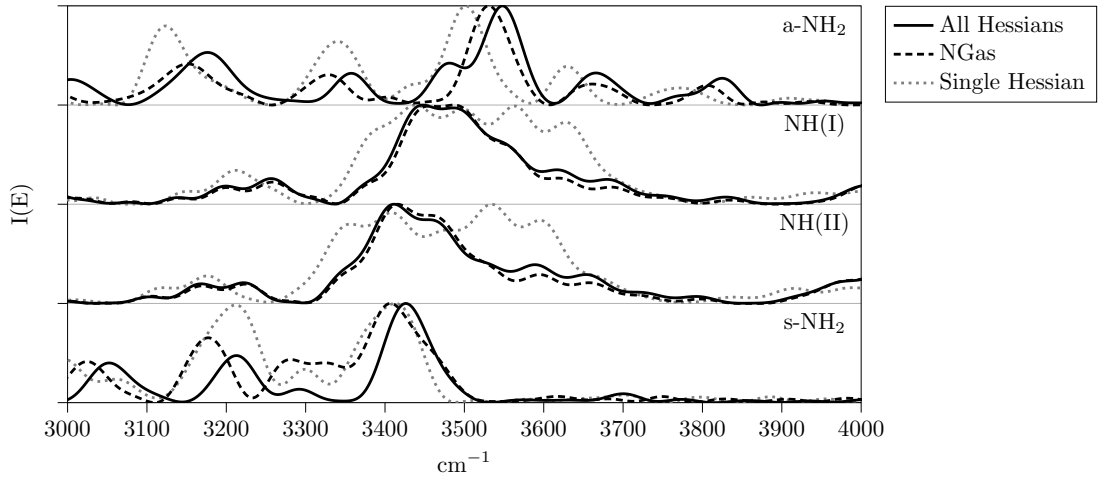


Figure 4.3: Spectroscopic Hessian accuracy test for the NH_2 stretches in the amide group of the Ac-Phe-Met-NH_2 peptide. The dotted line is for the Single Hessian approximation;[150] the dashed line for the Neural Gas approximation including the gradient information with 200 neurons and 200 *ab initio* Hessians calculation. The continuous line, which is labeled as "All Hessians", reports the simulation where all Hessians are obtained from *ab initio* calculations.

above, i.e. with the inclusion of the gradients in the set of neuron variables. The dotted line is the so-called "Single Hessian" (SH) approximation,[150] where the Hessian is constant and it is equal to the equilibrium geometry one. The NGas simulation is very similar to the exact, especially for the higher dimensional subspace. However, the SH approximation is quite good if one takes into account how drastic the approximation is. Nevertheless, the main problem of the SH approximation is that for the higher dimensional subspace containing the NH(I) and NH(II) stretches it does not allow for a definitive assignment, while in the case of the NGas spectra a main peak is present, despite the numerous overtone side peaks. We can confirm that these side peaks of smaller intensities are of the type of overtones or combination bands by comparing the NGas spectra with the classical ones in Figure 4.4. Quasi-classical spectra are obtained by Fourier transforming the velocity-velocity correlation function of a constant energy trajectory (NVE), which is the same employed for the MC-DC-SCIVR calculations, i.e. trajectories starting from the equilibrium geometry and with kinetic energy equals to the harmonic zero point energy. While these type of classical simulations provide frequency values with anharmonic corrections because the *ab initio* trajectory accounts for the shape of the PES, these values are restricted only to the fundamental transition frequencies and higher harmonics. Instead, semiclassical simulations, such as MC-DC SCIVR, provide the full collection of eigenvalues as in Eq. 2.8 and all type of transition frequencies can be obtained by difference between the eigenvalues. Thus, the semiclassical power spectrum includes not only the fundamental frequencies, but

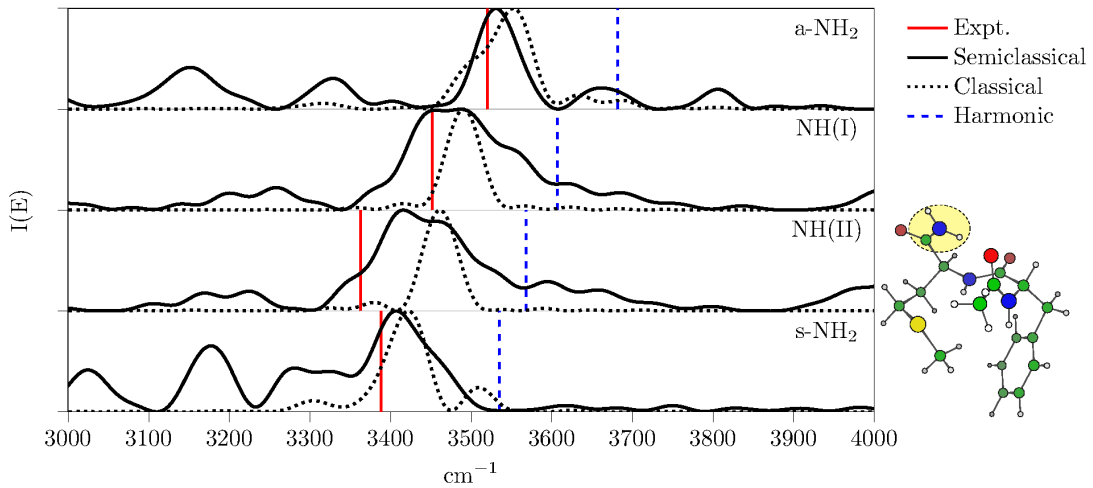


Figure 4.4: Ac-Phe-Met-NH_2 amide group related vibrational stretching power spectra. Vertical continuous sticks are the experimental values,[149] while vertical dashed sticks are the harmonic approximation frequencies. Continuous lines are for MC-DC-SCIVR simulation using the Hessian NGas approximation and dotted lines are for the quasi-classical simulation on the same *ab initio* potential (see main text).

also anharmonic overtones, combination bands and the ZPE value on an absolute scale. For these reasons the MC-DC-SCIVR spectra of Figure 4.4 (continuous lines) present several more spectroscopic features than the classical ones (dashed lines). However, it is still possible to compare the two of them with the experiments on the fundamental frequency values. The comparison is reported in Table 4.4 and Figure 4.4, where the experimental values[149] are reported as a red continuous stick spectrum, while the harmonic estimates are the dashed blue sticks. Overall, we can observe in Figure 4.4 that the semiclassical simulations present broader peaks than the classical ones. The classical peak width is what is expected from the Fourier transform of a $\sim$ 0.73 ps simulation. We do not pursue longer trajectories because the quantum accuracy of the semiclassical approximation would deteriorate for longer simulations. We also decided to not apply any artificial exponential constant decay (Gaussian filter) to avoid any sort of biasing. In Figure 4.4 the semiclassical signals are broader in the case of the NH(I) and NH(II) stretches as expected, because of the numerous strongly coupled floppy modes. Specifically, the more intense peaks in the NH(I) and NH(II) panels represent the convolution of a series of overtones coupled to the numerous floppy modes, while the other side peaks, which are absent in the classical spectrum, are combination or overtone bands of other modes. In fact, both the subspace subdivision and the filtering process provided by the combination of coherent states [68] can only partially filter the numerous eigenvalues which are present in a given energy window of a 132-dimensional power spectrum. Clearly, in Figure 4.4 these side peaks are less intense at higher

Table 4.4: Selected amide group vibrational stretching fundamentals of the Ac-Phe-Met-NH_2 peptide at different levels of approximation. The first column reports the type of stretch, the second the MC-DC-SCIVR frequencies without any Hessian approximation, the third and the fourth columns respectively the Neural Gas (NGas) and the Dynamic Hessian Database (DBH) approximated Hessians semiclassical frequency values, the fifth column the quasi-classical frequencies of vibration, the sixth column the harmonic results, and the last column the experimental values.[149] In the last row the Mean Absolute Error (MAE) with respect to the experimental values is reported for each method.

Modes	All Hessians	NGas	DBH [111]	Classical	Harmonic	Exp[149]
aNH_2	3548	3530	3490	3552	3682	3520
NH (I)	3448	3456	3480	3490	3607	3452
NH (II)	3412	3416	3300	3461	3568	3363
sNH_2	3426	3406	3360	3422	3535	3388
MAE	29	21	37	51	167	0.0

frequencies because the trajectory energy shell is at the level of the harmonic ZPE value, where the Fourier transformed coherent state is centered.

Table 4.4 summarizes the results in Figure 4.4 with the additional results of the semiclassical MC-DC-SCIVR simulation obtained using the Hessian Database approximation. [111] The comparison between different levels of calculation shows that classical and semiclassical results are systematically more accurate than the harmonic ones, while the semiclassical ones are further more accurate with respect to the classical ones. The semiclassical reference is reported in the second column of Tab. 4.4, where the calculations have been performed without any Hessian approximation but using directly the *ab initio* values. The third and the fourth column report respectively the NGas and the DBH approximated Hessians semiclassical values. For the NGas simulation 200 neurons and 200 *ab initio* Hessians have been employed, while the DBH results are obtained with 300 *ab initio* Hessians. [111] At this level of comparison we think it is not possible to assert which of the Hessian approximations, either the NGas or the DBH one, is more appropriate for spectroscopic analyses with the MC-DC-SCIVR method. Actually, the NGas MAE respect to the experimental values in Tab. 4.4 is slightly smaller than calculating all the *ab initio* Hessians. This is clearly due to a compensation of effects, which include the level of *ab initio* theory. Eventually, given the NGas and DBH MAE of Table 4.4, both of them are accurate enough for semiclassical calculations, considering that any semiclassical simulation strongly depends on the level of *ab initio* theory and that the Fourier transform broadening is about $\sim 20\ \mathrm{cm}^{-1}$ for a typical semiclassical trajectory simulation, where the total time is of the order of picoseconds.

4.4 Conclusions

Given the importance of an accurate method for approximating instead of calculating the Hessian matrix during molecular dynamics simulations, we have investigated the possibility to employ a slightly customized neural gas algorithm that allows us to compute the Hessian matrix of the potential energy along a molecular dynamics simulation. After presenting the method, we have tested its accuracy compared to other algorithms already present in the literature. [109–111] Then, we applied it to speeding up the calculation of semiclassical spectra, where the Hessian calculation is mandatory at each molecular dynamics time-step. We find that the NGas algorithm can be $\sim 20\%$ more accurate than other methods for simulations of molecular systems whose trajectories overlap and cross significantly. Furthermore, it appears that the NGas method may require far fewer *ab initio* Hessian calculations to provide the same accuracy as competitive methods. However, some caveats must be taken into account. First of all, if one aims to study a single short trajectory of a large molecular system (such as Ac-Phe-Met-NH_2), it appears that the NGas method is just as accurate as the Dynamical Hessian Database approach (DBH) that our group presented recently. [111] As a matter of fact, in such cases, the NGas method provides a solution that is similar to that proposed by DBH. Furthermore, if the user can afford to compute only very few Hessian calculations along a non overlapping trajectory, it is recommended to add gradients of the potential to the NGas training set. This set up is slightly more robust than the DBH method with respect to the number of ab initio Hessians. Secondly, while at high dimensions all methods scale favorably, the NGas method would suffer from longer simulations and higher number of neurons. However we expect that this feature should still compensate for the time spent for the *ab initio* calculation of all the Hessians. We didn't pursue the simulations of molecular systems significantly larger than Ac-Phe-Met-NH_2 because the Hessian calculations at each time-step would be out of reach for standard computational power. The third caveat is that the DBH method can be also performed on-the-fly while the NGas one is necessarily a post-processing method. This means that in the DBH method the number of *ab initio* Hessian calculations can be automatically determined during the dynamics if one applies the method to the available database at each time step and increment the database during the dynamics, while in the case of the NGas method it has to be fixed *a priori*. The last caveat is that the DBH parameter ρ is system dependent. Also ρ ensures that the approximated Hessians are close enough to the trajectory, but it does not allow to control the number of Hessians to compute. On the other hand, the NGas method requires as input the number of Hessians one is willing to compute, but it does not assure that the neurons locations would be close enough to the trajectory. Nevertheless, both these shortcomings can be mended by a preliminary trial and error calculation. Eventually, for semiclassical spectroscopic

calculations we conclude that both methods are accurate. We also tested the single Hessian approximation and confirm that this choice should be avoided or employed as a preliminary calculation together with a classical power spectrum calculation. Finally, in this work we have also shown that our DC-SCIVR technique implemented by reasonable approximations can allow for power spectrum calculations with the inclusion of quantum nuclear features of systems as large as small peptides. As a future perspective, our NGas method could be interfaced with methods that generate a local fit of the potential, such as the GPR-PES method. [117] In fact, the trajectory geometries within a Voronoi cell can be used to train a GPR model and better approximate the Hessian matrix within the same cell. This approach would allow for more reliable Hessian estimates, within the current limitations of applications of the GPR-PES methodology.

Chapter 5

Symplectic Integration

As discussed in section 2.2, the semiclassical Herman-Kluk propagator requires a numerical integration of the equations of motion that preserves the symplectic property. In this chapter we present in detail how to satisfy this property with numerical algorithms.

Consider a classical Hamiltonian function of the type:

$$H = K(p(t)) + V(q(t)) \tag{5.1}$$

where $K(p)$ is the kinetic energy function and $V(q)$ is the potential energy function. The canonical variables $p(t) = (p_1(t), \ldots, p_n(t))$, $q(t) = (q_1(t), \ldots, q_n(t))$ are vectors of the scalar coordinates $q_i(t)$ with conjugated momenta $p_i(t)$. We conveniently group them in the phase-space vector $z(t) = (q(t), p(t))$. The equations of motion for the system come from the time derivative of $z(t)$, which is given by Hamilton's equations

$$\dot{p_i}(t) = -\frac{\partial \mathcal{H}}{\partial q_i} \tag{5.2}$$

$$\dot{q_i}(t) = \frac{\partial \mathcal{H}}{\partial p_i}. \tag{5.3}$$

A more compact way to write the equations of motion

$$\dot{z}(t) = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix} \nabla \mathcal{H} \tag{5.4}$$

$$= \mathcal{J} \nabla \mathcal{H} \tag{5.5}$$

uses the Symplectic matrix $\mathcal{J} = \left(\begin{smallmatrix} 0 & I \\ -I & 0 \end{smallmatrix}\right)$ and the derivative operator $\nabla = \left(\frac{\partial}{\partial q}, \frac{\partial}{\partial p}\right)$. Eqs. 5.4 and 5.5 show that Hamilton's equations are skew-symmetric. It is possible to define a property called the linear symplectic structure, as $[z_A, z_B] = \mathcal{J} \cdot z_A \cdot z_B$. For a one dimensional system the symplectic structure evaluates to $[z_A, z_B] = p_A q_B - p_B q_A$, and

has the simple geometrical interpretation of the area of the parallelogram composed by z_A and z_B, as represented graphically in Figure 5.1.

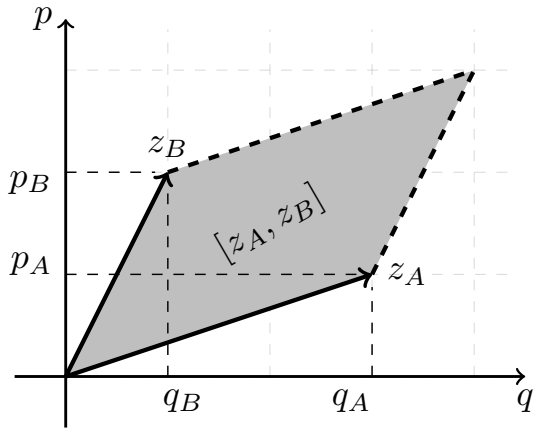


Figure 5.1: Geometrical representation of the symplectic structure $[z_A, z_B]$ in one dimension, as the area of the parallelogram delimited by the vectors z_A and z_B.

When the system is higher dimensional $[z_A, z_B]$ becomes a sum of terms $p_{Ai}q_{Bj} - p_{Bj}q_{Ai}$, each of which has the same geometrical interpretation. Thus the symplectic structure $[z_A, z_B]$ is the sum of the areas given by the projection of the parallelograms defined by the z_A and z_B over the coordinate plane. Since classical mechanics is deterministic the phase space at some time t is completely determined by the phase space vector at a previous time. Given an infinitesimal change of two phase space vectors $\partial z_A(t)$ and $\partial z_B(t)$, the symplectic structure is conserved upon multiplication with the Jacobian matrix from the left. That is

$$\left[\frac{\partial z_A(t)}{\partial z_A(t')} \cdot \partial z_A(t'), \frac{\partial z_B(t)}{\partial z_B(t')} \cdot \partial z_B(t')\right] = \left[\partial z_A(t), \partial z_B(t)\right]. \tag{5.6}$$

Equation 5.6 guarantees that time evolution preserves the symplectic structure, and that the sum of the projection areas remains constant in time. [151] Any phase space function $f(z)$ preserves the symplectic structure if and only if

$$\frac{\partial f(z)}{\partial z}^T \mathcal{J} \frac{\partial f(z)}{\partial z} = \mathcal{J}. \tag{5.7}$$

A numerical technique, or algorithm, which evolves the equations of motion and guarantees Eq. 5.7 for the integrals of motion can be called a symplectic integrator. Also notice that the Poisson bracket corresponds to the symplectic structure of the

gradients

$$\{\mathcal{H}, z\} = \frac{\partial \mathcal{H}}{\partial q}\frac{\partial z}{\partial p} - \frac{\partial \mathcal{H}}{\partial p}\frac{\partial z}{\partial q} \tag{5.8}$$

$$= -\begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix} \cdot \left(\frac{\partial \mathcal{H}}{\partial q}, \frac{\partial \mathcal{H}}{\partial p}\right) \cdot \left(\frac{\partial z}{\partial q}, \frac{\partial z}{\partial p}\right) \tag{5.9}$$

$$= -\left[\nabla \mathcal{H}, \nabla z\right]. \tag{5.10}$$

5.1 Splitting Methods and Symplectic Maps

A practical way to develop symplectic algorithms is to start from the solution of the equations of motion in Liouville's formalism for a time independent Hamiltonian, that is

$$z(\tau) = e^{-i\tau\hat{\mathcal{H}}} z(0) \tag{5.11}$$

$$= e^{-i\tau\left(\hat{\mathcal{K}}+\hat{\mathcal{V}}\right)} z(0), \tag{5.12}$$

where the symbols $\hat{\mathcal{H}}$, $\hat{\mathcal{K}}$, and $\hat{\mathcal{V}}$ are all operators that transform the state function $z(t)$ to its time derivative under the assumption that the system is subject to a Hamiltonian operator embedded in each symbol. For instance, in the most general case in which one wants to evolve the density matrix, Eq. 5.12 is the Liouville-von Neumann equation, with $z(t)$ the complex valued density matrix, and $\hat{\mathcal{H}} = \left[\hat{H}, \cdot\right]$. If we use quantum mechanics to evolve a pure state, then $z(t)$ is the complex-valued wave function and $\hat{\mathcal{H}}$ is the Hamiltonian operator. Finally, if we use classical mechanics, then $z(t)$ is a real-valued vector of the canonical coordinates, and $\hat{\mathcal{H}} = -i\{H, \cdot\}$ is the classical Liouville operator, which work as an imaginary Poisson bracket of the function it is evaluated to.

Independently of the mechanics (and state function) one adopts, the symbols in Eq. 5.11 and 5.12 obey similar mathematical rules. In particular, because the operators $\hat{\mathcal{K}}$ and $\hat{\mathcal{V}}$ do not commute, the exponential map in equation 5.12 cannot be written as a product of two exponential maps,

$$e^{-i\tau\left(\hat{\mathcal{K}}+\hat{\mathcal{V}}\right)} \neq e^{-i\tau\hat{\mathcal{K}}} e^{-i\tau\hat{\mathcal{V}}}. \tag{5.13}$$

In fact, in the general case,[1] the propagator cannot be written as a finite product of propagators of $\hat{\mathcal{K}}$ and $\hat{\mathcal{V}}$. [152, 153] However, it is possible to approximate the propagator

[1] The "general case" consists of a system with generic potential that cannot be approximated to a finite power series. We shall see in the next sections that a symplectic map of order n corresponds to approximating the potential to a power series of order at most equal to n.

with a single finite product of partial propagators of the form

$$e^{-i\tau\hat{\mathcal{H}}} \approx e^{-i\tau\hat{\mathcal{H}}_1} e^{-i\tau\hat{\mathcal{H}}_2} \dots e^{-i\tau\hat{\mathcal{H}}_n}, \tag{5.14}$$

with the condition $\sum_i \hat{\mathcal{H}}_i = \hat{\mathcal{H}}$. The practical choice $\hat{\mathcal{H}}_i = a_i\hat{\mathcal{K}}$ and $\hat{\mathcal{H}}_{i+1} = b_i\hat{\mathcal{V}}$, with constraints $\sum_i a_i = \sum_i b_i = 1$ is convenient because each partial propagator can be solved analytically. Although Eq. 5.14 is approximate, one can always interpret the right-hand-side as an exact solution of a locally approximate system. As we shall see, when the potential is a finite sum of powers, the propagator can be approximated as a single finite product of the partial propagators. The expression

$$e^{-i\tau\hat{\mathcal{H}}} \approx \mathcal{M}_n = e^{-i\tau a_1\hat{\mathcal{K}}} e^{-i\tau b_1\hat{\mathcal{V}}} \dots e^{-i\tau a_n\hat{\mathcal{K}}} e^{-i\tau b_n\hat{\mathcal{V}}} \tag{5.15}$$

is called a symplectic map, and corresponds to a symplectic integration algorithm that gives the exact time evolution of the approximated system. Eq. 5.15 is a symplectic algorithm independently of the value of the map order and of the coefficients a_k, and b_k. However, for the symplectic integration to be accurate, one needs to choose the map's coefficients a_k and b_k wisely.

To find the optimal form of the map in Eq. 5.15 one needs to expand all the exponential maps in powers of τ and choose the coefficients so that the terms with low orders of τ are zero. In the next to sections we show two approaches for that purpose.

5.2 Algebraic Solutions of Symplectic Maps

The expression $\log\left(e^{-a_1\tau\hat{\mathcal{K}}} e^{-b_1\tau\hat{\mathcal{V}}}\right)$. can be expanded into a more revealing form using the Baker-Campbell-Hausdorff (BCH) formula (see appendix B for more details about BCH)

$$\begin{aligned} -\tau\bar{\mathcal{H}}_1(a_1, b_1) &= \log\left(e^{-a_1\tau\hat{\mathcal{K}}} e^{-b_1\tau\hat{\mathcal{V}}}\right) && (5.16)\\ &= -a_1\tau\hat{\mathcal{K}} - b_1\tau\hat{\mathcal{V}} - a_1 b_1 \frac{\tau^2}{2}\left[\hat{\mathcal{K}}, \hat{\mathcal{V}}\right] + O(\tau^3) && (5.17) \end{aligned}$$

In order to make Eq. 5.17 exact one would need to set the a_1 and b_1 coefficients to 1, and set to zeros the coefficients of the commutators. In fact, the exact symplectic map would be $-\bar{\mathcal{H}}_\infty = -\hat{\mathcal{K}} - \hat{\mathcal{V}}$. For a first order solution one needs to set $a_1 = b_1 = 1$, which gives the Symplectic Euler (or Euler-Cromer[154]) integrator.

For the second order map one needs to apply the BCH formula three times,

$$\begin{aligned}\tau\bar{\mathcal{H}}_2(\mathbf{a},\mathbf{b}) &= \log\left(e^{a_1\tau\hat{\mathcal{K}}}e^{b_1\tau\hat{\mathcal{V}}}e^{a_2\tau\hat{\mathcal{K}}}e^{b_2\tau\hat{\mathcal{V}}}\right)\\ &= \tau(a_1+a_2)\hat{\mathcal{K}}+\tau(b_1+b_2)\hat{\mathcal{V}}\\ &\quad+\frac{\tau^2}{2}(a_1b_1-b_1a_2+a_1b_2+b_1b_2)\left[\hat{\mathcal{K}},\hat{\mathcal{V}}\right]\\ &\quad+\frac{\tau^3}{12}\left(a_1^2b_1+a_1^2b_2+b_1a_2^2+a_2^2b_2+2a_1a_2b_2-4a_1b_1a_2\right)\left[\hat{\mathcal{K}},\left[\hat{\mathcal{K}},\hat{\mathcal{V}}\right]\right]\\ &\quad+\frac{\tau^3}{12}\left(a_1b_1^2+a_1b_2^2+b_1^2a_2+a_2b_2^2+2a_1b_1b_2-4b_1a_2b_2\right)\left[\hat{\mathcal{V}},\left[\hat{\mathcal{V}},\hat{\mathcal{K}}\right]\right]\\ &\quad+O(\tau^4)\end{aligned} \tag{5.18}$$

Clearly in this case the solution can be sought after in the system of equations

$$\begin{cases}a_1+a_2=1\\ b_1+b_2=1\\ a_1b_1-b_1a_2+a_1b_2+b_1b_2=0.\end{cases} \tag{5.19}$$

This system is undetermined. It requires another independent equation to have a single solution. The second order map $\mathcal{M}_2(\tau)=\exp\left(-\tau\bar{\mathcal{H}}_2\left(\frac{1}{2},\frac{1}{2},1,0\right)\right)=e^{-\frac{i\tau}{2}\hat{\mathcal{K}}}e^{-\hat{\mathcal{V}}}e^{-\frac{i\tau}{2}\hat{\mathcal{K}}}$ is a very famous solution of Eq. 5.19. It is known by various names,[155] depending on the context (Strang splitting [156] by mathematicians, Trotter-Suzuki splitting [157] in the quantum mechanics community, Symplectic Leapfrog or explicit Verlet in the classical mechanics community). The reason for this choice of coefficients is that $\mathcal{M}_2$ is exactly time-reversible. In fact the time-reversibility condition is enforced by applying the inverse symplectic map after the symplectic map, that is

$$\mathcal{M}_n(\tau)\mathcal{M}_n(-\tau) = \left(e^{-a_1\hat{\mathcal{K}}}e^{-b_1\hat{\mathcal{V}}}\dots e^{-a_n\hat{\mathcal{K}}}e^{-b_n\hat{\mathcal{V}}}\right)\left(e^{a_1\hat{\mathcal{K}}}e^{b_1\hat{\mathcal{V}}}\dots e^{a_n\hat{\mathcal{K}}}e^{b_n\hat{\mathcal{V}}}\right). \tag{5.20}$$

The first step to make Eq. 5.20 equal to the identity one needs to set either $b_n=0$ or $a_1=0$, so that the last factor of $\mathcal{M}_n(\tau)$ would commute with the first factor of $\mathcal{M}_n(-\tau)$. Which gives

$$\mathcal{M}_n(\tau)\mathcal{M}_n(-\tau) = e^{-a_1\hat{\mathcal{K}}}e^{-b_1\hat{\mathcal{V}}}\dots e^{-b_{n-1}\hat{\mathcal{V}}}e^{-(a_n-a_1)\hat{\mathcal{K}}}e^{b_1\hat{\mathcal{V}}}\dots e^{a_n\hat{\mathcal{K}}} \tag{5.21}$$

Thus, time-reversibility is enforced when

$$b_n=0, a_n=a_1, b_{n-1}=b_1, a_{n-1}=a_2, b_{n-2}=b_2,\dots \tag{5.22}$$

Or when

$$a_1=0, b_n=b_1, a_n=a_2, b_{n-1}=b_2, a_{n-1}=a_3,\dots \tag{5.23}$$

Thus, in the case of the second order map $\mathcal{M}_2$ the time-reversibility condition is perfectly enforced. Also notice that, since $b_2 = 0$, the second order map can be obtained by applying the BCH formula only twice. Yoshida proved that a symplectic map which satisfies $\mathcal{M}_n(\tau)\mathcal{M}_n(-\tau) = \mathcal{M}_n(-\tau)\mathcal{M}_n(\tau) = 1$ must be of an even order.[158] The proof is simple. Assuming that $\mathcal{M}_n(\tau)$ is time-reversible and be of the type

$$\mathcal{M}_n(\tau) \quad = \quad e^{\tau\gamma_1+\tau^2\gamma_2+\tau^3\gamma_3+\tau^4\gamma_4+\cdots}, \tag{5.24}$$

where γ_m is a sum of commutators of order m. Then the product $\mathcal{M}_n(\tau)\mathcal{M}_n(-\tau)$ gives

$$\mathcal{M}_n(\tau)\mathcal{M}_n(-\tau) \quad = \quad e^{\tau\gamma_1+\tau^2\gamma_2+\tau^3\gamma_3+\tau^4\gamma_4+\cdots}e^{-\tau\gamma_1+\tau^2\gamma_2-\tau^3\gamma_3+\tau^4\gamma_4+\cdots} \tag{5.25}$$

which is identical to 1 only if $\gamma_2 = \gamma_4 = \cdots = \gamma_{even} = 0$. This can be demonstrated by using the BCH formula twice and comparing with the exact propagator to lower orders of τ. [158, 159]

One may continue to apply the BCH formula to look for higher order maps, and to check the existence of time-reversible maps that violate Eqs. 5.22 and 5.23. However, a simpler approach to get higher order maps is to search for higher order algorithms by composition of lower order ones. [157, 158, 160, 161] Consider the following analogy. $\mathcal{M}_2(\tau) = e^{-\frac{\tau}{2}\hat{\mathcal{K}}}e^{-\tau\hat{\mathcal{V}}}e^{-\frac{\tau}{2}\hat{\mathcal{K}}}$ applies three propagators, of which the first and last are identical. Thus one can imagine to construct a map composed of three $\mathcal{M}_2(\tau)$ propagators as

$$\begin{aligned}\mathcal{M}_2(x\tau)\mathcal{M}_2(y\tau)\mathcal{M}_2(x\tau) &= \left(e^{-\frac{x\tau}{2}\hat{\mathcal{K}}}e^{-x\tau\hat{\mathcal{V}}}e^{-\frac{x\tau}{2}\hat{\mathcal{K}}}\right)\left(e^{-\frac{y\tau}{2}\hat{\mathcal{K}}}e^{-x\tau\hat{\mathcal{V}}}e^{-\frac{y\tau}{2}\hat{\mathcal{K}}}\right)\times \\ &\quad \left(e^{-\frac{x\tau}{2}\hat{\mathcal{K}}}e^{-x\tau\hat{\mathcal{V}}}e^{-\frac{x\tau}{2}\hat{\mathcal{K}}}\right) \qquad (5.26)\\ &= e^{-\frac{x\tau}{2}\hat{\mathcal{K}}}e^{-x\tau\hat{\mathcal{V}}}e^{-\frac{(x+y)\tau}{2}\hat{\mathcal{K}}}e^{-x\tau\hat{\mathcal{V}}}e^{-\frac{(x+y)\tau}{2}\hat{\mathcal{K}}}e^{-x\tau\hat{\mathcal{V}}}e^{-\frac{x\tau}{2}\hat{\mathcal{K}}}. \qquad (5.27)\end{aligned}$$

If we apply BCH formula to Eq. 5.27 twice we get a map that will turn out to be of fourth order

$$\mathcal{M}_4(x, y, \tau) = e^{(y+2x)\tau\alpha_1+(y^3+2x^3)\tau^3\alpha_3+O(\tau^5)}. \tag{5.28}$$

The conditions on Eq. 5.28 are simply

$$\begin{cases} y + 2x = 1 \\ y^3 + 2x^3 = 0, \end{cases} \tag{5.29}$$

which has the unique real solution

$$\begin{cases} y = -\dfrac{2^{1/3}}{2 - 2^{1/3}} \\ x = \dfrac{1}{2 - 2^{1/3}}. \end{cases} \tag{5.30}$$

Substitution of Eq. 5.30 into Eq. 5.28 gives the set of a_k and b_k coefficients already discovered independently by many authors [161–164].

$$\begin{cases} a_1 = a_4 = (2^{1/3} + 2^{-1/3} + 2)/6 \\ a_2 = a_3 = -(2^{1/3} + 2^{-1/3} - 1)/6 \\ b_1 = 0 \\ b_2 = b_4 = (2^{4/3} + 2^{2/3} + 4)/6 \\ b_3 = -(2^{7/3} + 2^{5/3} + 2)/6 \end{cases} \tag{5.31}$$

Following the same strategy one can obtain even higher order integration algorithms. For instance a sixth order map can be obtained by the symmetric product of three fourth order ones $\mathcal{M}_6(x, y, \tau) = \mathcal{M}_4(x\tau)\mathcal{M}_4(y\tau)\mathcal{M}_4(x\tau)$. In general the symmetric product rule of lower order maps is obtained as

$$\mathcal{M}_{n+2} = \mathcal{M}_n(x_n\tau)\mathcal{M}_n((1 - 2x_n)\tau)\mathcal{M}_n(x_n\tau), \tag{5.32}$$

with $x_n = 1/(2 - 2^{1/(n+1)})$.

A major inconvenience of the symmetric product approach is that some of the coefficients are greater than one, and others may be negative. This detail is inconvenient, because coefficients larger than one imply propagation for a time longer than the time-step, and negative coefficients mean propagation back in time (to compensate oversized steps). Furthermore, if one aims to propagate a diffusive (stochastic) process, having negative coefficients make no sense. In order to approach and overcome these difficulties Suzuki proposed to include higher order commutators in the symplectic map. In particular, Chin showed that plugging the operator $e^{-\tau d_k[\hat{\mathcal{V}},[\hat{\mathcal{K}},\hat{\mathcal{V}}]]}$ in the symplectic map allows to build an extremely accurate fourth order map, whose accuracy is comparable to traditional sixth or eighth order map. Chin proposed to initially construct a fourth order map of the type

$$\mathcal{M}_4(x, y, z, \tau) = e^{x\tau\hat{\mathcal{V}}}e^{y\tau\hat{\mathcal{K}}}e^{z\tau\hat{\mathcal{V}}}e^{y\tau\hat{\mathcal{K}}}e^{x\tau\hat{\mathcal{V}}} \tag{5.33}$$

that has BCH expansion

$$\begin{aligned}\log \mathcal{M}_4(x,y,z) &= 2y\tau\hat{\mathcal{K}} + (z+2x)\tau\hat{\mathcal{V}} + \\ &-\frac{1}{6}y\left(z^2-2zx-2x^2\right)\tau^3\left[\hat{\mathcal{V}},\left[\hat{\mathcal{K}},\hat{\mathcal{V}}\right]\right] + \\ &+\frac{1}{6}y^2\left(x-4z\right)\tau^3\left[\hat{\mathcal{K}},\left[\hat{\mathcal{V}},\hat{\mathcal{K}}\right]\right] + O(\tau^5). \end{aligned} \tag{5.34}$$

If we set the conditions

$$\begin{cases} y = \dfrac{1}{2} \\ z+2x = 1 \\ z = 4x \end{cases} \tag{5.35}$$

we obtain the map

$$\log \mathcal{M}_4 = \tau\left(\hat{\mathcal{K}}+\hat{\mathcal{V}}\right) - \frac{1}{72}\tau^3\left[\hat{\mathcal{V}},\left[\hat{\mathcal{K}},\hat{\mathcal{V}}\right]\right] + O(\tau^5) \tag{5.36}$$

with coefficients $x = 1/6, y = 1/2, z = 2/3$. The nested commutator in Eq. 5.36 can be removed by the transformation $e^{\tau^3/144[\hat{\mathcal{V}},[\hat{\mathcal{K}},\hat{\mathcal{V}}]]}\mathcal{M}_4 e^{\tau^3/144[\hat{\mathcal{V}},[\hat{\mathcal{K}},\hat{\mathcal{V}}]]}$. Chin proposed also to insert the corrective term in the center of the map, that is

$$\begin{aligned}\mathcal{M}_4(\tau) &= e^{\frac{\tau}{6}\hat{\mathcal{V}}}e^{\frac{\tau}{2}\hat{\mathcal{K}}}e^{\frac{\tau^3}{144}[\hat{\mathcal{V}},[\hat{\mathcal{K}},\hat{\mathcal{V}}]]}e^{\frac{2\tau}{3}\hat{\mathcal{V}}}e^{\frac{\tau^3}{144}[\hat{\mathcal{V}},[\hat{\mathcal{K}},\hat{\mathcal{V}}]]}e^{\frac{\tau}{2}\hat{\mathcal{K}}}e^{\frac{\tau}{6}\hat{\mathcal{V}}} && (5.37)\\ &= e^{\frac{\tau}{6}\hat{\mathcal{V}}}e^{\frac{\tau}{2}\hat{\mathcal{K}}}e^{\frac{2\tau}{3}\hat{\mathcal{V}}}e^{\frac{\tau}{2}\hat{\mathcal{K}}}e^{\frac{\tau}{6}\hat{\mathcal{V}}} && (5.38)\end{aligned}$$

where $\hat{\hat{\mathcal{V}}} = \hat{\mathcal{V}} + \frac{\tau^2}{48}\left[\hat{\mathcal{V}},\left[\hat{\mathcal{K}},\hat{\mathcal{V}}\right]\right]$. Eq. 5.38 is the most computationally convenient version of this type of map, simply because $\left[\hat{\mathcal{V}},\left[\hat{\mathcal{K}},\hat{\mathcal{V}}\right]\right]$ commutes with $\hat{\mathcal{V}}$, but not with $\hat{\mathcal{K}}$. Using the classical mechanics definitions for $\hat{\mathcal{K}}$ and $\hat{\mathcal{V}}$, the nested commutator corresponds to a product of the Hessian matrix and the force,

$$\left[\hat{\mathcal{V}},\left[\hat{\mathcal{K}},\hat{\mathcal{V}}\right]\right](q(t),p(t)) = -2\frac{\partial^2 V(q(t))}{\partial q^2}\cdot\frac{\partial V(q(t))}{\partial q}\cdot\frac{\partial p(t')}{\partial p}. \tag{5.39}$$

In conclusion the propagator in Eq. 5.38 has the desired property of positive only (and small) coefficients and, as a consequence, is significantly more accurate than traditional symplectic integrators. The drawback is the need to estimate the hessian matrix at every time-step, as well as three estimation of the potential gradient per time-step.

5.3 Derivation of the Coefficients by Taylor Expansions

A less refined approach to find the coefficients of symplectic map is a simple Taylor expansion of the propagators, followed by a comparison with the time-Taylor expanded variables. We present also this approach, because it was used in chapter 6.

The time evolution of a system described by the classical Hamiltonian $\mathcal{H}(q,p) = K(p) + V(q)$, of phase space variables $z = (q,p)$ corresponds to a solution of Hamilton's equations. In Liouville's formalism

$$\dot{z} = \{H, z\}\,. \tag{5.40}$$

The Poisson bracket is defined as $\{H,z\} = \dfrac{\partial H}{\partial q}\dfrac{\partial z}{\partial p} - \dfrac{\partial H}{\partial p}\dfrac{\partial z}{\partial q}$ and the derivatives of z define the base vectors in phase space, that is $\dfrac{\partial z}{\partial p} = (0,1)$ and $\dfrac{\partial z}{\partial q} = (1,0)$. Following Rangarajan, Dragt and Neri [161], we define the Lie operator $\{H,\cdot\}$, whose effect is the substitution of $\cdot$ with the function that it is applied to and evaluated accordingly. For instance, $\{H,\cdot\}\, f(a) = \{H,f\}\,(a)$. Thus, we can solve Eq. 5.40 in terms of z and t, for $t = \tau$ with initial condition $t = 0$

$$z(\tau) = e^{-\tau\{H,\cdot\}} z(0). \tag{5.41}$$

By definition $e^{-\tau\{H,\cdot\}} = 1 + \tau\,\{H,\cdot\} + \dfrac{\tau^2}{2!}\,\{H,\cdot\}^2 + \ldots$ and it is a sum of sequences of Poisson brackets.

The application of the two rightmost operators in the symplectic map gives a complex expression, in which the potential (blue) and kinetic (green) parts of the Lie operator are mixed

$$\begin{aligned} z(\tau) &= \mathcal{M} z(0) && (5.42)\\ &= \ldots e^{-\tau b_1\{V,\cdot\}} e^{-\tau a_1\{K,\cdot\}} z(0) && (5.43)\\ &= \ldots (1 - a_1\tau\,\{K,\cdot\} - b_1\tau\,\{V, 1 - a_1\tau\,\{K,\cdot\}\}) z(0). && (5.44) \end{aligned}$$

The $z(\tau)$ vector can then be written in two components as

$$\begin{aligned} p(\tau) &= p(0) - b_1\tau\frac{\partial V}{\partial q}(0)\\ q(\tau) &= q(0) + a_1\tau\frac{\partial K}{\partial p}(0) - b_1 a_1\tau^2\frac{\partial V}{\partial q}(0).\\ &= q(0) + a_1\tau p(\tau) \end{aligned}$$

For the two equations to be accurate we compare them with the Taylor series in the time domain, and find out that the optimal coefficients are $a_1 = 1$ and $b_1 = 1$. The

resulting integrator corresponds to the semi-implicit Euler method (or Symplectic Euler, or Euler-Cromer). A similar analysis can be carried out for the second order map $\mathcal{M}_{k=2}$, which provides the following equations:

$$\begin{aligned} p(\tau) &= p(0) - b_1\tau\frac{\partial V}{\partial q}(0) - b_2\tau\frac{\partial V}{\partial q}\left(q(0) + a_1\tau p(0) - a_1 b_1\tau^2\frac{\partial V}{\partial q}(0)\right) \\ q(\tau) &= q(0) + (a_1+a_2)\,p(0)\tau - (a_1+a_2)\,b_1\tau^2\frac{\partial V}{\partial q}(0) \\ &\quad -a_2 b_2\tau^2\frac{\partial V}{\partial q}\left(q(0) + a_1\tau p(0) - a_1 b_1\tau^2\frac{\partial V}{\partial q}(0)\right). \end{aligned} \tag{5.45}$$

Here, however, the forces are computed at displaced positions (or at different times), hence we expand the forces in powers of q around $q(0)$. Comparing the expanded version of Eq. 5.45 to the Taylor series, up to the second order, one gets the following system of equations for the coefficients:

$$\begin{cases} a_1 + a_2 = 1 \\ b_1 + b_2 = 1 \\ a_1 b_2 = \dfrac{1}{2} \\ a_1 b_1 + a_2 b_1 + a_2 b_2 = \dfrac{1}{2}. \end{cases} \tag{5.46}$$

This system is undetermined. However, the simplest and most famous solution is $a_1 = 1, a_2 = 0, b_1 = 1/2, b_2 = 1/2$. Another solution is $a_1 = 1/2, a_2 = 1/2, b_1 = 0, b_2 = 1$ and this is the symplectic leapfrog method. Other, more accurate solutions are obtained by adding constraints to the third order terms. By constraining $a_1 b_1 b_2 = 1/6$, one obtains an integrator that is exact to 3rd order in the momentum and it has an error equal to $(1/24)p_0\tau^3\partial^2 V(0)/\partial q^2$ on the position. Such integrator has the coefficients $a_1 = 3/4, a_2 = 1/4, b_1 = 1/3, b_2 = 2/3$ and it is not time-reversible. Higher order integrators can be obtained with the same procedure[161, 165]. For the fourth order map $\mathcal{M}_{n=4}$, we can derive and solve the system of equations with the help of the SageMath[166] computer algebra system:

$$
\begin{cases}
a_1 + a_2 + a_3 + a_4 = 1 \\
b_1 + b_2 + b_3 + b_4 = 1 \\
(a_1 + a_2 + a_3 + a_4)b_1 + (a_2 + a_3 + a_4)b_2 + (a_3 + a_4)b_3 + a_4 b_4 = \frac{1}{2} \\
a_1 b_2 + (a_1 + a_2)b_3 + (a_1 + a_2 + a_3)b_4 = \frac{1}{2} \\
(a_1 + a_2 + a_3)a_4 b_4 + (a_1 a_2 + a_1 a_3 + a_1 a_4)b_2 + ((a_1 + a_2)a_3 + (a_1 + a_2)a_4)b_3 = \frac{1}{6} \\
a_1 b_1 b_2 + ((a_1 + a_2)b_1 + a_2 b_2)b_3 + ((a_1 + a_2 + a_3)b_1 + (a_2 + a_3)b_2 + a_3 b_3)b_4 = \frac{1}{6} \\
\frac{1}{2}a_1^2 b_2 + \frac{1}{2}(a_1^2 + 2a_1 a_2 + a_2^2)b_3 + \frac{1}{2}(a_1^2 + 2a_1 a_2 + a_2^2 + 2(a_1 + a_2)a_3 + a_3^2)b_4 = \frac{1}{6} \\
(a_1 a_2 + a_1 a_3 + a_1 a_4)b_1 b_2 + (((a_1 + a_2)a_3 + (a_1 + a_2)a_4)b_1 + (a_2 a_3 + a_2 a_4)b_2)b_3 + \\
\quad +((a_1 + a_2 + a_3)a_4 b_1 + (a_2 + a_3)a_4 b_2 + a_3 a_4 b_3)b_4 = \frac{1}{24} \\
\frac{1}{2}(a_1^2 + 2a_1 a_2 + a_2^2 + 2(a_1 + a_2)a_3 + a_3^2)a_4 b_4 + \frac{1}{2}(a_1^2 a_2 + a_1^2 a_3 + a_1^2 a_4)b_2 + \\
\quad +\frac{1}{2}((a_1^2 + 2a_1 a_2 + a_2^2)a_3 + (a_1^2 + 2a_1 a_2 + a_2^2)a_4)b_3 = \frac{1}{24} \\
a_1^2 b_1 b_2 + ((a_1^2 + 2a_1 a_2 + a_2^2)b_1 + (a_1 a_2 + a_2^2)b_2)b_3 + ((a_1^2 + 2a_1 a_2 + a_2^2 + 2(a_1 + a_2)a_3 + a_3^2)b_1 + \\
\quad +(a_1 a_2 + a_2^2 + (a_1 + 2a_2)a_3 + a_3^2)b_2 + ((a_1 + a_2)a_3 + a_3^2)b_3)b_4 = \frac{1}{8} \\
\frac{1}{6}a_1^3 b_2 + \frac{1}{6}(a_1^3 + 3a_1^2 a_2 + 3a_1 a_2^2 + a_2^3)b_3 + \frac{1}{6}(a_1^3 + 3a_1^2 a_2 + 3a_1 a_2^2 + a_2^3 + 3(a_1 + a_2)a_3^2 + a_3^3 + \\
\quad +3(a_1^2 + 2a_1 a_2 + a_2^2)a_3)b_4 = \frac{1}{24} \\
a_1 a_2 b_2 b_3 + ((a_1 + a_2)a_3 b_3 + (a_1 a_2 + a_1 a_3)b_2)b_4 = \frac{1}{24}
\end{cases}
\tag{5.47}
$$

It is important to notice that this system is undetermined. We need to choose another constraint to effectively get numerical values for a_k and b_k coefficients. A reasonable choice to saturate the system is to set $b_1 = 0$, which leads to three possible sets of real coefficients. The first set was first published by Forest and Ruth[162]

$$
\begin{cases}
a_1 &= \frac{2^{1/3}+2^{-1/3}+2}{6} \\
a_2 &= -\frac{2^{1/3}+2^{-1/3}-1}{6} \\
a_3 &= -\frac{2^{1/3}+2^{-1/3}-1}{6} \\
a_4 &= \frac{2^{1/3}+2^{-1/3}+2}{6} \\
b_1 &= 0 \\
b_2 &= \frac{2^{4/3}+2^{2/3}+4}{6} \\
b_3 &= -\frac{2^{7/3}+2^{5/3}+2}{6} \\
b_4 &= \frac{2^{4/3}+2^{2/3}+4}{6}
\end{cases}
, \tag{5.48}
$$

while the second was reported by Brewer *et al.* [167],

$$\begin{cases} a_1 &= \sqrt{3}/6 + 1/2 \\ a_2 &= -\sqrt{3}/3 \\ a_3 &= \sqrt{3}/3 \\ a_4 &= -\sqrt{3}/6 + 1/2 \\ b_1 &= 0 \\ b_2 &= -\sqrt{3}/6 + 1/4 \\ b_3 &= 1/2 \\ b_4 &= \sqrt{3}/6 + 1/4 \end{cases} \tag{5.49}$$

and a third set that is equal to the set in Eq. 5.49 but with b_2 swapped with b_4, a_2 swapped with a_3, and a_1 swapped with a_4. These three are the only real solutions that we have found to the system in Eq. 5.47 with the added constraint $b_1 = 0$. There are other complex solutions that we disregarded. Another famous choice for the saturating conditions is $a_4 = 0$ and $b_1 = b_4$, which leads to the 4th order integrator allegedly discovered by Neri and re-derived by Yoshida[158] and by Forest and Ruth[162].

Chapter 6

Dynamics of Artificially Decoupled Systems[1]

In this chapter we introduce the concept of pair-decoupling and we give a recipe to accurately simulate a system whose nuclear degrees of freedom are artificially decoupled. [95] The technique we use to integrate the equations of motion preserves symplectic symmetry and can be generalized to any order of accuracy using the composition method of Eq. 5.32. The pair-decoupling integration method is applied to study the vibrational spectroscopy of the Salicylic Acid with classical mechanics.

6.1 Introduction

The coupling among atoms in molecular systems is a key concept in chemistry and material science, especially in organic chemistry where it allows for an intuitive and qualitative description of the rich reactivity of organic compounds. Direct evidence of the vibrational couplings can be observed in the features of the vibrational (IR and Raman) spectrum, and it can be measured [168–170] in the off-diagonal features of 2D vibrational spectra. In theoretical chemistry, 2D vibrational spectra can be calculated either by using a model coupling Hamiltonian [168, 169], semiclassical approaches [170], or other trajectory based methods [171]. This field of research is important even outside the realm of spectroscopy, because rationalization of the vibrational spectra allows the prediction, for instance, of selectivity[172, 173] and reaction yields[174].

The aim of the study described in this chapter is to investigate the effects of the couplings from a *dynamical* perspective, accounting for the real time vibrations of the molecules, and employing the intuitive Cartesian coordinate system. To that

[1] This chapter is a selection with minor modifications of the content of the paper **Michele Gandolfi**, and **Michele Ceotto**, "Molecular Dynamics of Artificially Pair-Decoupled Systems: An Accurate Tool for Investigating the Importance of Intramolecular Couplings"

end we provide an entirely new approach to the study of couplings: We rely on a practical description of the coupling between pairs of degrees of freedom, that, in its simplicity, allows us to define numerical experiments of artificially *de*coupled atoms in molecules. Specifically, we introduce a method that allows a real time, full dimensional, and numerically accurate simulation of an artificially decoupled system.

We define the atom-atom coupling as the phenomenon in which the force perceived by atom A depends from the position of atom B. In such cases we would say that the motion of A and B is correlated, or that A and B are coupled. First of all, we imagine a molecule represented by a collection of atoms, in which *each* pair of atoms are linked by the endpoints of a connector. If we put the molecule in its geometrical equilibrium and then abruptly displace a single atom, we would have a force acting on every atom that tries to move the whole system towards the nearest potential minimum. The force acts by either elongating or compressing (compared to their state at equilibrium) each connector and trying to adjust them in response to the deformation. This means that atoms A and B, connected by the connector c_{AB}, would feel either an attractive force that pulls them together or a repulsive one that pushes them apart. However, if we cheat the physics and artificially set this force to its value before the deformation, A and B would perceive each other just as if the system were still in equilibrium, and so, the connector c_{AB} would not respond to the deformation, while all the other atoms and connectors would perceive the force and respond accordingly. For instance, another atom C would perceive the deformation, and c_{AB} might either elongate or compress (because of the compression/elongation of c_{AC} and/or c_{BC}). As a result of this artificial intervention, the motion of decoupled atoms (A and B) in response to the deformation, is directly uncorrelated (although it could be indirectly correlated via a third atom). A cartoon of the pair-decoupling idea is shown in Figure 6.1.

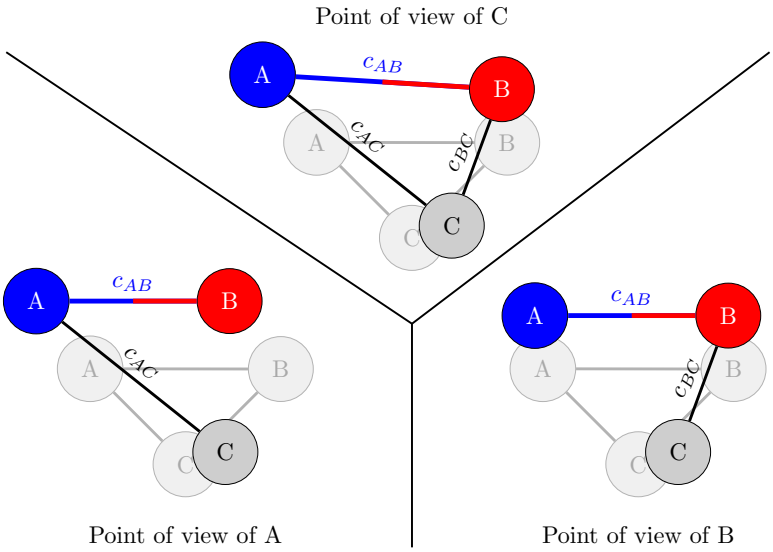


Figure 6.1: Cartoon of the pair-decoupling idea, representing the points of view of the three atoms compared to their initial geometry (the shaded molecules in the background). In the top panel is represented a snapshot of the simulation, which corresponds to the (objective) point of view of atom C. In the left and right panels are represented the points of view of the decoupled ($\alpha = 0$) atoms A and B respectively, in which either atom B *perceives* atom A as if it never displaced (from its point of view) or vice versa. From the points of view of A and B, the connector c_{AB} is set at the initial geometry value.

In practice, if $\dot{f}_{AB}$ is the rate of change of the force perceived by the connector c_{AB} in normal condition, we could artificially scale it as $\dot{f}_{AB} \cdot \alpha$, where α is a real number between 0 and 1. Of course, the closer α is to 1, the more coupled A and B are, and the more realistic the simulation is. On the contrary, the closer α is to 0, the more artificial it is. The practice of artificially modifying the potential is commonly used in accelerated molecular dynamics methods,[175–178] to explore the configuration space faster.

We propose to perform the artificial decoupling in a molecular dynamics simulation, in which the atoms are moved according to their initial velocities, and in which we use α to shield the atoms from seeing each other's displacements. To reach this goal, we developed a very simple numerical technique, called the Symplectic Explicit with Force (SEF) integration algorithm that allows an accurate time evolution of pair-decoupled systems. Moreover, we show that the SEF integration of the equations of motion preserves the symplectic symmetry and preserves in a significant amount also time-reversibility. We also show how the time-reversibility and energy conservation properties are exact for harmonic potentials and still accurate when the potential is anharmonic. The reader should notice that the potential between the atoms A and B is not modified when they are in their equilibrium position: the attractive/repulsive contribution of

the force on the pair is artificially modified only when they are displaced during time evolution. In fact, when $\alpha = 0$, the pair-decoupling implies that pairs of atoms may perceive each other as if they remain in their initial position. This detail also implicates that, while the equilibrium properties of the system may be unaltered by the decoupling (as it can be the case of normal mode coordinate decoupling), the dynamical properties may change very significantly, as we shall see in the section dedicated to the Salicylic Acid decouplings.

Hereafter, the Theory and Methods Section formalizes the pair-decoupling idea and presents the molecular dynamics modified integrator for artificially decoupled pairs of atoms. The Results Section presents the bench-marking of the method and its application to the Salicylic Acid. The Discussion and Conclusions Section concludes the work.

6.2 Theory and Methods

6.2.1 Decoupling Hamiltonian

As anticipated above, we want to describe molecular systems as composed of some arbitrarily pair-decoupled atoms. We call $H = K + V$ the Hamiltonian of the fully-coupled (normal) system and $\tilde{H} = K + \tilde{V}$ the Hamiltonian of the corresponding pair-decoupled system. For the rest of the work we will assume that V (and $\tilde{V}$) contains the electron-electron, electron-nucleus and nucleus-nucleus Coulomb interactions, as well as the electrons exchange potential, and the electronic kinetic energy. Thus V (and $\tilde{V}$) is a function of the nuclei positions in the Born-Oppenheimer approximation and K is the corresponding nuclear kinetic energy. We make the further approximation that the nuclei behave as classical particles, so that we can express the classical pair-decoupled Hamiltonian as $\tilde{H} = \sum_\alpha \left(\tilde{p}_\alpha^2/2m_\alpha\right) + \tilde{V}(\tilde{q})$, where we assign the tilde ($\sim$) symbol to the canonical coordinates $\tilde{q}$ and $\tilde{p}$ to specify that they are phase-space coordinates of the pair-decoupled Hamiltonian. While we do not have an explicit expression for $\tilde{V}$ in terms of V, we express the relationship in terms of the potential derivatives, because we employ these for integrating the equation of motion. Specifically, we assume that the main coupling between pairs is given by the second order derivative terms w.r.t. coordinates q_i and q_j (i.e. the Hessian matrix elements h_{ij}) and that it can be artificially scaled:

$$\frac{\partial^2 \tilde{V}(\tilde{q})}{\partial \tilde{q}_i \partial \tilde{q}_j} = \alpha \frac{\partial^2 V(\tilde{q})}{\partial \tilde{q}_i \partial \tilde{q}_j} = \alpha h_{ij}\left(\tilde{q}\right). \tag{6.1}$$

In Eq.(6.1), we have assumed that the decorrelation is for pairs of degrees of freedom and α is the amount of decoupling, ranging from 1 (no decoupling) to 0 (fully decoupled).

However, one can decouple multiple degrees of freedom at the same time. For instance, the pair-decoupled Hessian matrix for a three degrees of freedom system where two of them are fully coupled and the third one is partially decoupled from them, is

$$\tilde{h}(\tilde{q}_1, \tilde{q}_2, \tilde{q}_3) = \begin{pmatrix} h_{11} & h_{12} & \alpha h_{13} \\ h_{12} & h_{22} & \alpha h_{23} \\ \alpha h_{13} & \alpha h_{23} & h_{33} \end{pmatrix}. \tag{6.2}$$

Notice that when $\alpha = 0$, Eq. 6.2 corresponds to the Hessian matrix of two independent systems, one of which is two dimensional and the second is mono-dimensional. When the time-evolution algorithm described in the next sections is applied to such a system, the evolution of the system is artificially separable and the potential is up to the second order of the type $V(\tilde{q}_1, \tilde{q}_2, \tilde{q}_3) = V_{1,2}(\tilde{q}_1, \tilde{q}_2) + V_3(\tilde{q}_3)$. Notice that in case of a truly separable potential, we could write $V_3(\tilde{q}_3) - V_3(\tilde{q}_3^{eq}) = V(\tilde{q}_1, \tilde{q}_2, \tilde{q}_3) - V(\tilde{q}_1, \tilde{q}_2, \tilde{q}_3^{eq})$ and $V_{1,2}(\tilde{q}_1, \tilde{q}_2) - V_{1,2}(\tilde{q}_1^{eq}, \tilde{q}_2^{eq}) = V(\tilde{q}_1, \tilde{q}_2, \tilde{q}_3) - V(\tilde{q}_1^{eq}, \tilde{q}_2^{eq}, \tilde{q}_3)$. These last expressions correspond to the "projected potentials" used to compute the vibrational spectroscopic features of molecules as large as G-quadruplex in solution[179] with the Divide-and-Conquer SemiClassical Initial Value Representation (DC-SCIVR) method.[49–51]. While the DC-SCIVR method simulates the dynamics of a system under a full dimensional potential and then approximates the classical action with a potential projected into subspaces[49], the algorithm we are presenting here evolves the dynamics entirely under the subspace-projected potential (or partially projected, when $\alpha \neq 0$). Furthermore, the decoupling could be applied to all the degrees of freedom pertaining to two atoms, that is, for instance, to the Cartesian product $(x_1, y_1, z_1) \times (x_2, y_2, z_2)$, to decouple atoms 1 and 2. In this work we focus on this Cartesian atom-decoupling scheme, as it appeals to chemical intuition, and we believe it would result as the most interesting for the chemical community. Nonetheless the decoupling idea could be applied to any coordinate system.

In practice, we propose a time-propagation rule that allows to enforce the pair decoupling idea by making explicit the contributions to the force given by the Hessian matrix. In other words, we obtain the molecular dynamics force by numerically integrating the Hessian matrix over time as $\tilde{F}(t) = \int_0^t \tilde{h}(t')\dot{\tilde{q}}(t')dt'$. However, this procedure is tricky because the variables $\tilde{q}(t)$, $\tilde{p}(t)$ and $\tilde{F}(t)$ must be evaluated at the same time for the propagation to take place:

$$\begin{aligned} \tilde{q}(\tau) &= \tilde{q}(0) + \int_0^\tau \tilde{p}(t)\ dt \\ \tilde{p}(\tau) &= \tilde{p}(0) + \int_0^\tau \tilde{F}(t)\ dt. \\ \tilde{F}(\tau) &= \tilde{F}(0) - \int_0^\tau \tilde{h}(t) \cdot \tilde{p}(t)/m\ dt, \end{aligned} \tag{6.3}$$

where $\tilde{h}(t) \coloneqq \alpha\partial^2 V(\tilde{q}(t))/\partial\tilde{q}_i\partial\tilde{q}_j$ is the pair-decoupled Hessian matrix at time t. The procedure we have developed employs a standard symplectic integration for the coordinates $\tilde{q}$ and $\tilde{p}$, while the force is updated from a time-integration of the Hessian matrix embedded with the symplectic map, consistently with the canonical variables. Specifically, for an integration that is accurate to order n, the practical update is a cycle over the integer k, up until $k = n = 2$ or 4 of the following four simple steps:

$$\begin{aligned} \tilde{p}_k &= \tilde{p}_{k-1} + b_k\tau\tilde{F}_{k-1} \\ \tilde{q}_k &= \tilde{q}_{k-1} + a_k\tau\frac{\tilde{p}_k}{m} \\ q_{aux} &= \tilde{q}_k + \sum_{j=1}^{k}(a_j - b_j)\frac{\tilde{p}_k}{m}\tau \\ \tilde{F}_k &= \tilde{F}_{k-1} - c_k\tau\left(\tilde{h}(q_{aux})\cdot\frac{\tilde{p}_k}{m}\right), \end{aligned} \tag{6.4}$$

where m is the mass and $\tilde{p}_k = \tilde{p}\left(t = \sum_{j=1}^{k} b_j\tau\right)$ and $\tilde{q}_k = \tilde{q}\left(t = \sum_{j=1}^{k} a_j\tau\right)$ are the momentum and position (vector) variables at step k of the symplectic map starting from the initial conditions $\tilde{p}_0, \tilde{q}_0$, and $\tilde{F}_0 = \partial V(\tilde{q}_0)/\partial\tilde{q}$. The numerical coefficients a_k, b_k and c_k are universal real numbers that depend only on the order of approximation. Elegant derivations of the a_k and b_k coefficients for high order integrators can be found in the literature, as many authors have worked in the field of symplectic integration. [155, 158, 159] Thus, in the next sections we give only a brief overview for the derivation of the a_k and b_k coefficients, which are solutions of the system of equations given in the Appendix (we leave a more in depth explanation in the Supporting Information material). In the next sections we discuss in some more details how the c_k coefficients can be easily obtained for a second order integrator, and how the fourth and higher order integrators can be obtained by composition of second order ones. For a second order integration we found the unique solution $b_1 = 0, b_2 = 1$; $a_1 = c_1 = 1/2, a_2 = c_2 = 1/2$, which corresponds to the symplectic leapfrog algorithm with $c_k = a_k$. We then propose a 4th order version of the pair-decoupled algorithm as a symmetric product of three leapfrog algorithms, with coefficients $a_1 = a_4 = (2^{1/3} + 2^{-1/3} + 2)/6$; $a_2 = a_3 = -(2^{1/3} + 2^{-1/3} - 1)/6$; $b_1 = 0$; $b_2 = b_4 = (2^{4/3} + 2^{2/3} + 4)/6$; $b_3 = -(2^{7/3} + 2^{5/3} + 2)/6$, $c_k = a_k$. This choice is the most accurate according to our numerical tests. However, several other choices arise when the higher order algorithms are not derived as symmetric products of lower order algorithms, as discussed in the next sections and more in detail in the Supporting Information of this work.

6.2.2 Integration of the pair-decoupled system

We construct our algorithm to be of the type of a nth order symplectic map

$$\mathcal{M}_n = e^{-\tau b_n \{\tilde{V},\cdot\}} e^{-\tau a_n \{K,\cdot\}} \dots e^{-\tau b_1 \{\tilde{V},\cdot\}} e^{-\tau a_1 \{K,\cdot\}}, \tag{6.5}$$

which consists of a time evolution of a free system, followed by a time evolution of the pair-decoupled system with zero velocity, followed by evolution of the free system and so on. Notice that, in comparison with the standard symplectic map in Eq. 5.15, we modify only the form of the potential energy operator $\hat{\mathcal{V}}$, without changing the structure of the map, which remain symplectic, independently of how we modify $\hat{\mathcal{V}}$. In fact, as long as Eq. 6.5 can be written as a single product of time evolution operators, we are sure that symplectic structure is preserved, contrary, for instance, to standard Runge-Kutta-Nystrom algorithms, which cannot be written as a single product, as explained by Chin[159]. We use the definition given in Eq. 6.1 to integrate the Hessian and get the locally harmonic approximated expression for the (pair-decoupled) force

$$\tilde{F}(\tau) \;=\; \tilde{F}(0) - \sum_{k=1}^{n} \tilde{h}\left(\tilde{q}(b_k\tau)\right) \cdot \dot{\tilde{q}}(b_k\tau)\; c_k\tau. \tag{6.6}$$

The positions and momenta are those resulting from the application of the operators in the symplectic map of Eq. 6.5. Notice that in Eq. 6.6, one needs $\tilde{q}$ and $\dot{\tilde{q}}$ to be evaluated at the same time $t = b_k\tau$. However, after the application of the two rightmost operators of Eq. 6.5, one obtains the position and the velocity at different time values, i.e. $\tilde{q}(a_1\tau)$ and $\tilde{p}/m = \dot{\tilde{q}}(b_1\tau)$, since, in general, $a_1 \neq b_1$. Hence, we introduce an auxiliary position variable $q_{1,aux} := \tilde{q}(a_1\tau) - (a_1 - b_1)\tau\dot{\tilde{q}}(b_1\tau) \approx q(b_1\tau)$, which is the position estimate at the same instant of time of the conjugated momentum variable. In general $q_{k,aux} := \tilde{q}(a_k\tau) - \sum_{j=1}^{k}(a_j - b_j)\tau\dot{\tilde{q}}(b_k) \approx q(b_k\tau)$. In this way the integration of the force in Eq. 6.6 is consistent with the rest of the algorithm. Finally, given Eq. 6.6, we can compare the explicit form of the evolution operator for the pair-decoupled potential with the evolution operator of the original potential. To do that, we first evolve $z(0)$ until we get $(p(t_p), q(t_q)) = e^{-\tau a_k \{K,\cdot\}} \prod_i^{k-1} \left(e^{-\tau b_i \{\tilde{V},\cdot\}} e^{-\tau a_i \{K,\cdot\}}\right) z(0)$, where we call $t_p = \sum_i^k a_i\tau$ and $t_q = \sum_i^{k-1} b_i\tau$ the time arguments of the momentum and position variables. Then, the additional application of the evolution operator $e^{-\tau b_k \{\tilde{V},\cdot\}}$ gives:

$$\begin{aligned}
\text{normal}: \; e^{-\tau b_k \{V,\cdot\}} z(t_p, t_q) &= \left[1 + F\left(q(b_k\tau)\right)\right] \frac{\partial z}{\partial p}(t_p, t_q) \\
\text{decoupled}: \; e^{-\tau b_k \{\tilde{V},\cdot\}} \tilde{z}(t_p, t_q) &= \left[1 + \left(\tilde{F}(0) - \sum_{i=1}^{k} \tilde{h}\left(q_{i,aux}\right) \cdot \dot{\tilde{q}}(a_i\tau) c_i\tau\right)\right] \frac{\partial \tilde{z}}{\partial \tilde{p}}(t_p, t_q)
\end{aligned} \tag{6.7}$$

We call $\tilde{F}_k := \left(\tilde{F}(0) - \sum_{i=1}^{k} \tilde{h}\left(q_{i,aux}\right) \cdot \dot{\tilde{q}}(a_i\tau)\; c_i\tau\right)$, $\tilde{p}_k := \tilde{p}_{k-1} + b_k\tau\tilde{F}_{k-1}$, and $\tilde{q}_k := \tilde{q}_k + a_k\tau\tilde{p}_k$ the variables we need to store to implement the algorithm. We determine the

coefficients c_k by considering that in the case of no decoupling ($\alpha = 1$) the pair decoupled algorithm must provide a good estimate of the original force, i.e. $\tilde{F}_k \approx F_k$, and of the pair-decoupled position and momentum. Hence, a route to find the c_k coefficients is to assume that a_k and b_k coefficients are equal to those derived for the fully coupled system, and then to choose the c_k coefficients so that the errors on $\tilde{q}_n$ and $\tilde{p}_n$ (and $\tilde{F}_n$) are of a given order of τ when $\alpha = 1$. Notice that, in the particular case of a quadratic potential, $\tilde{h}$ in Eq. 6.7 is a constant (and $\tilde{F}(0) = 0$, assuming that at $t = 0$ the system is in equilibrium). Thus, for a quadratic potential we have $F(q(b_k\tau)) \approx -\tilde{h} \cdot \sum_{i=1}^{k} \dot{\tilde{q}}(a_i\tau)c_i\tau$. This is a good estimate of the force when the sum $\sum_{i=1}^{k} \dot{\tilde{q}}(a_i\tau)c_i\tau$ is a good estimate of the position, that is when $c_i = a_i$. In fact, the appealing choice $c_k = a_k$ is appropriate also for non quadratic potentials when using a second order integrator (that is, either $a_1 = a_2 = 1/2, b_1 = 0, b_2 = 1$, or $b_1 = b_2 = 1/2, a_1 = 0, a_2 = 1$). This is not surprising, because the expanded expression for $e^{-\frac{\tau}{2}\hat{\mathcal{K}}}e^{-\tau\hat{\mathcal{V}}}e^{-\frac{\tau}{2}\hat{\mathcal{K}}}z(t)$ contains the derivatives of $V(q)$ and $K(p)$ up to the second order, meaning that it is, in fact, a propagator of the system under a local quadratic approximation of the exact functions $K(\tilde{p})$, which is actually quadratic, and $\tilde{V}(\tilde{q})$, which is not. We call this choice the "SEF2" method (Symplectic Explicit with Force integration of the 2nd order). The full set of coefficients are reported in the corresponding lines of Table 6.1.

The obvious way to derive an integrator of order four and higher is by composing lower order integrators, as described in section 5.1. The fourth order SEF4 integrator can be obtained as a symmetric product of the second order SEF2, with a_k and b_k coefficients equal to those derived by Forest and Ruth (also, independently derived by Campostrini and Rossi, and Candy and Rozmus), [163, 164] and with $c_k = a_k$. All the coefficients are reported in the SEF4 row of Table 6.1. Integrators of order 6 and higher can be easily obtained in the same way, resulting in the coefficients reported by Yoshida[158], with $c_k = a_k$. However, the method of composing lower order integrators does not generate all the solutions to the 4th order symplectic map. All such solutions can be found by solving the system in Eq. 5.49, where, however, time reversible symmetry is not enforced. In particular, the coefficients reported in the Appendix of Ref. [167] are solutions to Eq. 5.47, but do not enforce time reversibility, despite providing an integrator that is more accurate than Forest and Ruth's (in terms of energy conservation). While it is possible to build a pair-decoupled integrator using the a_k and b_k coefficients by Brewer *et al.*, it is not possible to reach 4th order accuracy, and there are multiple possible choices of the c_k coefficients. We derived versions of the SEF4 integrator using the Taylor expansion approach described in section 5.3, and accounting for the updated force in Eq. 6.6. All such coefficients are reported in the rows SEF4-I to SEF4-IV of Table 6.1, as possible variants of the SEF4 integrator.

In the Numerical tests Section we show numerically that our modification of

Table 6.1: Summary of the a_k, b_k, and c_k coefficients for various versions of the SEF algorithm.

SEF version	coeff.	$k=1$	$k=2$	$k=3$	$k=4$
SEF2[a)]	a_k	$1/2$	$1/2$	0	0
	b_k	0	1	0	0
	c_k	$1/2$	$1/2$	0	0
SEF4[b)]	a_k	$\frac{(2^{1/3}+2^{-1/3}+2)}{6}$	$\frac{-(2^{1/3}+2^{-1/3}-1)}{6}$	$\frac{-(2^{1/3}+2^{-1/3}-1)}{6}$	$\frac{(2^{1/3}+2^{-1/3}+2)}{6}$
	b_k	0	$\frac{(2^{4/3}+2^{2/3}+4)}{6}$	$\frac{-(2^{7/3}+2^{5/3}+2)}{6}$	$\frac{(2^{4/3}+2^{2/3}+4)}{6}$
	c_k	$\frac{(2^{1/3}+2^{-1/3}+2)}{6}$	$\frac{-(2^{1/3}+2^{-1/3}-1)}{6}$	$\frac{-(2^{1/3}+2^{-1/3}-1)}{6}$	$\frac{(2^{1/3}+2^{-1/3}+2)}{6}$
SEF4-I[c)]	a_k	$\sqrt{3}/6+1/2$	$-\sqrt{3}/3$	$\sqrt{3}/3$	$-\sqrt{3}/6+1/2$
	b_k	0	$-\sqrt{3}/6+1/4$	$1/2$	$\sqrt{3}/6+1/4$
	c_k	$5/26$	$8\sqrt{3}/39+4/13$	$-8\sqrt{3}/39+4/13$	$5/26$
SEF4-II[c)]	a_k	$\sqrt{3}/6+1/2$	$-\sqrt{3}/3$	$\sqrt{3}/3$	$-\sqrt{3}/6+1/2$
	b_k	0	$-\sqrt{3}/6+1/4$	$1/2$	$\sqrt{3}/6+1/4$
	c_k	$5\sqrt{3}/48$	$\sqrt{3}/8+1/2$	$-\sqrt{3}/8+1/2$	$-5\sqrt{3}/48$
SEF4-III[c)]	a_k	$\sqrt{3}/6+1/2$	$-\sqrt{3}/3$	$\sqrt{3}/3$	$-\sqrt{3}/6+1/2$
	b_k	0	$-\sqrt{3}/6+1/4$	$1/2$	$\sqrt{3}/6+1/4$
	c_k	$1/4$	$\sqrt{3}/6+1/4$	$-\sqrt{3}/6+1/4$	$1/4$
SEF4-IV[c)]	a_k	$\sqrt{3}/6+1/2$	$-\sqrt{3}/3$	$\sqrt{3}/3$	$-\sqrt{3}/6+1/2$
	b_k	0	$-\sqrt{3}/6+1/4$	$1/2$	$\sqrt{3}/6+1/4$
	c_k	$\sqrt{3}/6$	$1/2$	$1/2$	$-\sqrt{3}/6$

a) using Symplectic Leapfrog a_k and $b_k$coefficients

b) using Forest and Ruth a_k and b_k coefficients[162]

c) using Brewer, Hulme, and Manolopoulos a_k and b_k coefficients[167]

the symplectic algorithm, that accounts for the pair-decoupling concept, preserve the properties of symplectic integration. Considering the Jacobian matrix

$$J_{ij}(t,t') = \frac{\partial(\tilde{p}_i(t), \tilde{q}_j(t))}{\partial(\tilde{p}_i(t'), \tilde{q}_j(t'))}, \tag{6.8}$$

and the canonical symplectic matrix

$$\mathcal{J} = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}, \tag{6.9}$$

we measure how much the relation $J^T(t,t') \cdot \mathcal{J} \cdot J(t,t') = \mathcal{J}$ holds true for the special case $t' = 0$. In the case of a quadratic potential, the time evolution is exactly time reversible. Instead, in the case of a generic potential, the time-reversibility property is only approximate.

6.2.3 Computational details

In order to test our algorithm of Eq. 6.4, we employ accurate quartic force fields[54, 180] for the simulations of Water and Formaldehyde molecules. Instead, for Salicylic Acid calculations, we use the fitted potential energy DFT surface[181, 182] provided alongside

the sGDML software.[181–187] The sGDML PES is given already trained[181, 182] on 1000 training points and it showed, with the inclusion of the Tkatchenko-Scheffler correction to account for the van der Waals interactions,[188] a Mean Absolute Error (MAE) which is less than 0.12 kcal/mol with respect to the pVDZ/DFT-PBE values.

To test the accuracy of the integration technique, we run 4 types of tests with $\alpha = 1$. First we check energy conservation along the simulation. Then, we check the symplectic property of the Jacobian matrix. Then, we check the time-reversibility of the integrator and eventually we compute also the classical power spectrum. The check of energy conservation may appear redundant, because a symplectic integration, by definition, implies that all the constants of motion are preserved. However, our approximation of the force implies that the system is evolved under an approximation of the potential. As a matter of fact, even if $\alpha = 1$, if the force estimate in Eq. 6.6 is not accurate, the energy (of the fully coupled system) might not be accurately conserved, while the integration remain symplectic on the approximated potential.

To prove that the Jacobian matrix $J(t, t')$ is symplectic we use the relation $J^T(t, t') \cdot \mathcal{J} \cdot J(t, t') = \mathcal{J}$ for the special case $t' = 0$, as anticipated above. The Jacobian with $t' = 0$ is called the monodromy matrix $M(t)$, which can be computed numerically with the extended version of the algorithm described in the Supporting Information. Hence, we asses the stringent condition[42]

$$\Upsilon(t) = \sqrt{\sum_i \sum_j \left| (M(t)^T \cdot \mathcal{J} \cdot M(t))_{ij} - \mathcal{J}_{ij} \right|^2} \approx 0. \tag{6.10}$$

Although Eq. 6.10 proves the symplectic property of the Jacobian matrix only for the special case $t' = 0$, this is the most stringent test from the numerical point of view.

To measure the degree of time reversibility of the integrator, we run a simulation until time T, with a 10 a.u. time step. After that, we invert the sign of the momentum variable and we continue the propagation for another time lapse equal to T backward, until a total simulation time of $2T$ is reached. Finally, we measure the quantity

$$\tau(t) = \frac{1}{F} \sum_{j=1}^{F} |x_j(2T - t) - x_j(t)| \approx 0, \tag{6.11}$$

where $F = 3N_{at}$ (N_{at} is the number of atoms) and x_j is the j^{th} element of the F-dimensional Cartesian geometry vector. In all our tests $T = 6000$ a.u.

Finally, we apply our integration technique for the calculation of the vibrational spectra, ranging from small molecules up to the Salicylic Acid molecule in gas phase. We use a numerically convenient formula[52] to evaluate the power spectrum of the j^{th}

mass-scaled normal mode,

$$I_j(\omega_j) = \frac{1}{T}\left|\int_0^T p_j(t)e^{i\omega_j t}dt\right|^2. \tag{6.12}$$

This formula provides a resolved power spectrum with short ($\approx$ 0.6ps) simulations. In fact, Eq. 6.12 is the classical analogue of the Time-Averaging method employed in semiclassical spectroscopy [36, 58]. Since Eq. 6.12 computes a power spectrum from the velocity correlation function, all vibrational frequencies are reproduced and they can be compared with either Infra-red or Raman experimental frequencies. Instead, the intensities $I(\omega_j)$ are not comparable with IR or Raman experiments, because they depend only on the number of times the vibrational mode with frequency ω_j has occurred during the simulation time, which depends ultimately on the trajectory initial conditions. On the other hand, the experimental Infra-red and Raman intensities depend, for example, on the transition dipole moments and on the polarizabilities. To represent the power spectrum intensity of a multi-dimensional system we simply compute the sum of the power spectra of Eq. 6.12, that is

$$I(\omega) = \sum_j I_j(\omega_j). \tag{6.13}$$

The definite integral of Eq. 6.13 over a frequency domain can be interpreted as the average kinetic energy of the modes of vibration within that frequency domain[15]. Thus, when the intensity $I(\omega)$ of the pair-decoupled simulation is different from the non-decoupled one, there must be a shift in the vibrational frequency, or an intramolecular vibrational energy redistribution caused by the decoupling. The two effects may occur at the same time.

Although all the integrators described in this work allow evolving the system in any full-dimensional coordinate system, we always employ mass-scaled normal modes, which have the advantage of discarding translational and rotational motion. To decouple the Cartesian degrees of freedom, we just rotate the normal mode Hessian matrix to Cartesian coordinates, we apply the decoupling, and rotate the decoupled matrix back to normal modes. We use this procedure, instead of evolving in Cartesian coordinates, because the SEF algorithm cannot accurately describe free translations and rotations, or other types of motion that have a very flat (in general very anharmonic) potential landscape.

All the simulations described in this work are full dimensional and start from the equilibrium geometry of the fully coupled system. The normal mode coordinates are constructed using the fully coupled Hessian matrix. Also, in case the SEF algorithm is used, the initial force is assumed to be 0, just as if the initial geometry were an energy

minimum for the pair-decoupled system as well. The initial momentum in normal mode coordinates is set equal to the square root of the corresponding harmonic frequency, so that the initial kinetic energy is equal to the harmonic zero point energy. In the simulations with $\alpha \neq 1$ we follow the same recipe, but, when the effect of the decoupling is weak, we run the simulations for longer time (5000 time steps), and discard the initial 2000 steps (which is about 0.5 ps), to allow the decoupled fragments to actually decorrelate. In some cases, such as when decoupling all the functional groups of Salicylic Acid, the decoupling effect is very strong and the decorrelation effects can be seen already from the very beginning of the simulation. In such cases, we run the simulation only for 3000 steps and discard none. Anyway, if the pairs of atoms are naturally independent, the normal spectrum and the pair-decoupled spectrum would be exactly the same.

6.3 Results

6.3.1 Numerical tests

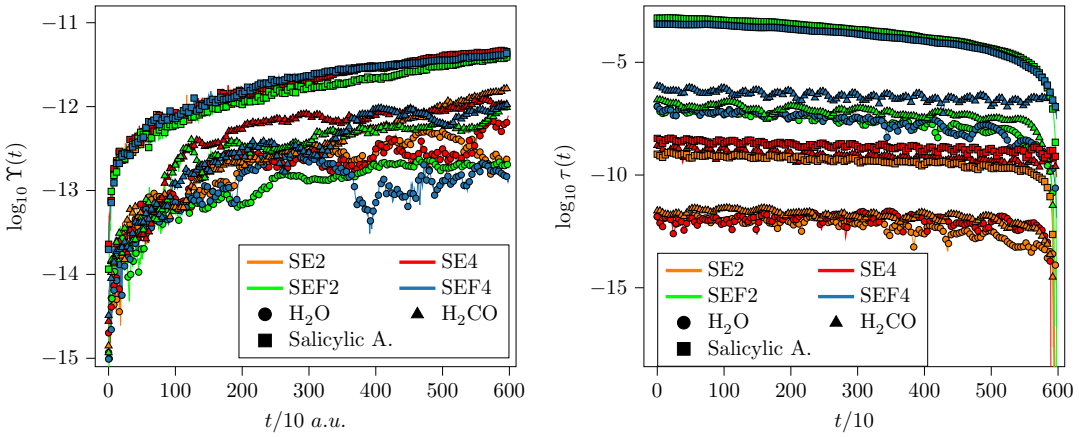


Figure 6.2: $\Upsilon(t)$ and $\tau(t)$ for a 3000 time-step simulation of 10 a.u. each. SE2 (orange), SE4 (red), SEF2 (green), and SEF4 (blue) integration methods for H_2O(circles), H_2CO(triangles), and Salicylic Acid (squares). SEF2 and SEF4 are tested without decoupling.

We start by testing the accuracy of our algorithms. In Fig. 6.2 we show how much the SEF2 and SEF4 integrators with $\alpha = 1$ preserve the symplectic symmetry of the monodromy matrix $M(t)$ and the time reversibility property. These are compared with the well established Symplectic Leapfrog (SE2) and 4th order SE4 method, that is the Symplectic Explicit integration method with the coefficients of Forest and Ruth[162]. Independently of the integrator, the larger the system the quicker $\Upsilon(t)$ and $\tau(t)$ deteriorates, and this is mainly due to the fact that more operations are carried out in a

finite precision arithmetic. However, when switching from the SE to the SEF algorithms no significant further errors in $\Upsilon(t)$ are introduced, while the time reversibility accuracy is decreased by orders of magnitude. This is expected for two main reasons. First, the calculation of the force in the SEF algorithms is performed by time integration and it requires four sums of matrix multiplications. The second, and the most important one, is that the calculation of the force is based on a local harmonic approximation of the potential landscape. Our approximate evolution of the force within the local harmonic approximation is not a time-reversible process, except for quadratic potentials. These limitations are clearly amplified with the dimensionality.

The symplectic properties of the SEF integration are preserved also in case of the decoupled pairs of degrees of freedom, i.e. $\alpha = 0$. We show in the Supporting Information (Fig. S1) that $\Upsilon(t)$ and $\tau(t)$ have the same shape even for the pair-decoupled system.

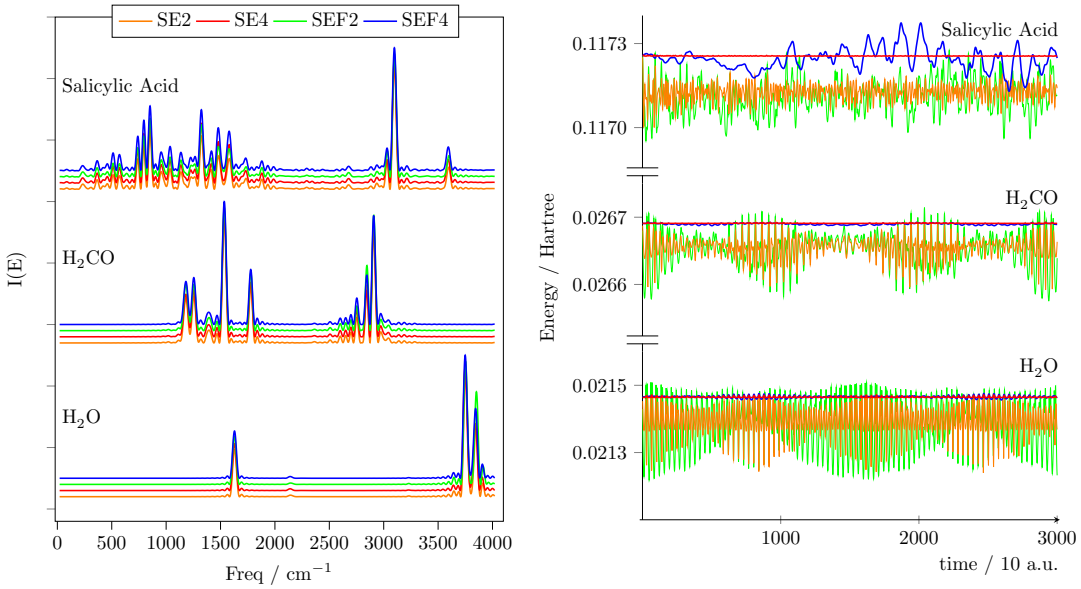


Figure 6.3: Power spectra and energy conservation. The left panel shows the power spectra of H_2O, H_2CO, and Salicylic Acid molecules computed with the SE2 (orange), SE4 (red), SEF2 (green), and SEF4 (blue) integrators (always with $\alpha = 1$). The right panel shows the energy profiles of the corresponding simulations.

In Figure 6.3 we show that, for all systems, the spectroscopic features are perfectly captured by all the integration method. SEF4, SE2 and SE4 provide spectra that are almost quantitatively equivalent when applied to all systems. The total energy of the H_2O and H_2CO systems is well conserved by the SEF algorithms, with SEF2 having an oscillation that is about $\sim$10% larger than SE2, and SEF4 an oscillation that is $\sim$20% to $\sim$30% larger than SE4. When the system includes floppy modes, however, such as

the Salicylic Acid, these modes induce a slow oscillatory pattern in the energy profile that is not well captured by the SEF algorithm. In fact, SEF can not predict very accurately the strongly anharmonic contributions to the force. Nevertheless, this is not an issue, because the SEF energy does not display a systematic drift, but only a slow oscillatory pattern that follows the oscillation of the low energy modes.

6.3.2 Decoupling the Salicylic Acid fragments

Since the pair decoupling is, by definition, an artificial procedure, we rationalize the following results in a *reductio ad absurdum* style, where first we enforce that some fragments of the molecule are independent and simulate the corresponding system, and then we see how much the vibrational features are affected by the decoupling. In this way we can observe that decoupling some fragments of the Salicylic Acid does not lead to significant conformational changes within a short simulation time, while decoupling other fragments quickly leads to unrealistic phenomena. However, given a long enough simulation time, decoupling any pairs of molecular fragments will lead, eventually, to unphysical behaviors.

Previous infrared spectroscopic studies of Salicylic Acid (SA) focused primarily on the intramolecular H-bond between Hydrogen 11 and Oxygen 9[189–193] and Hydrogen 11 and Oxygen 10[189] (see the atom numbering in Figure 6.4) in the ground and 1st excited electronic states. These studies are mainly about the proton transfer process and deactivation of the excited electronic state via a radiationless mechanism, which is possible only if O9 and H11 are close enough, as shown in panel A) of Figure 6.4, which is at the equilibrium geometry of the ground state. Below we show that such a configurational arrangement is stable over time only if the motion of O9 and H11, as well as of O9 and H16, is correlated. Furthermore, we investigate how much the carboxyl O−H stretching motion changes after artificially decoupling the different functional groups of the molecule.

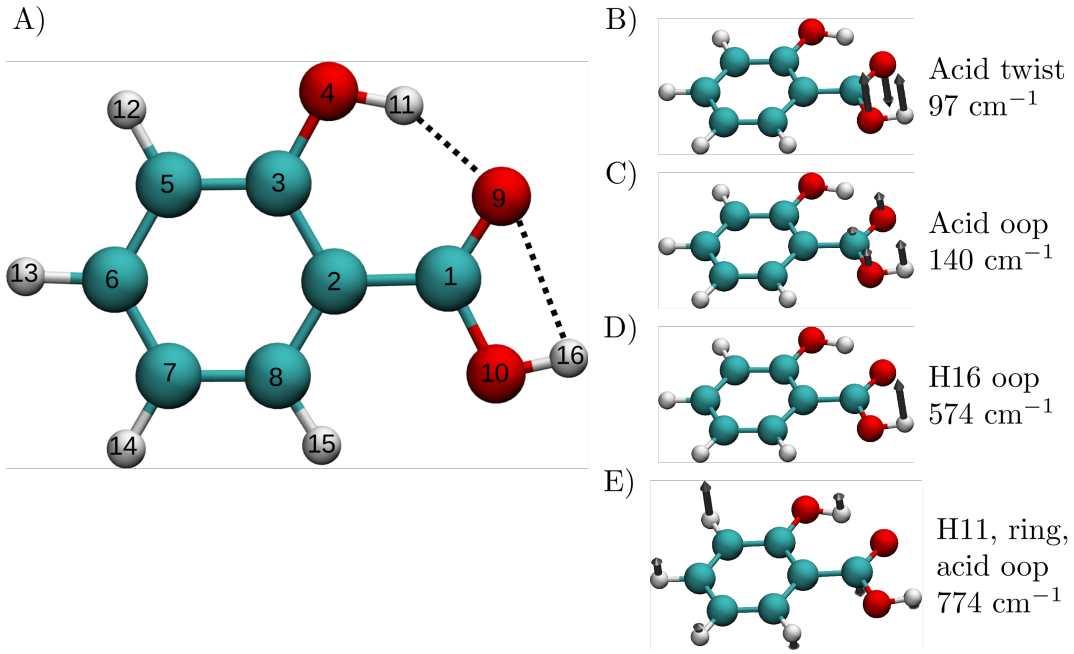


Figure 6.4: A) Picture of Salicylic Acid with labeled atoms and B), C), D), E) some relevant types of motion involving the Acid group, H11, and H16 with their relative harmonic frequencies.

Modes of vibration of the Salicylic Acid

The Salicylic Acid molecule in its minimum energy geometry and within the harmonic approximation has 42 normal modes of vibration. Only some of them show a significant displacement of H11 or H16. However, beyond this approximation, the vibration of the hydroxyl and carboxylic acid fragments imply the significant displacement of H11 and H16 from their equilibrium condition. More specifically, approximating the O4 – H11 and O10 – H16 stretching modes with the harmonic approximation implies to neglect the coupling between these stretching modes and the twist, wag, and other complex motion that involve the whole OH and Acid fragments. Four low frequency normal modes which are crucial for the Salicylic Acid vibrational motion are reported in panels B), C), and D) of Figure 6.4. These modes are the Acid group twist and the Acid group out-of-plane (oop) modes, which involve, respectively, a twist and oop wagging of the carboxyl group with respect to the ring, and the H16 oop motion, which is an out of plane wag of the H16 hydrogen. In addition, in panel E) of Fig. 6.4, is represented an out-of-plane mode that is delocalized over the three functional groups, involving H11, the ring, and the carboxylic acid group. We find that these four types of motion are those that are most significantly influenced by the pair decoupling of the hydrogen bonded fragments, and of the functional groups of SA. There are two main reasons for this. One is that they involve flexible regions of the molecule that easily couple with

many other types of motion and the other is that they break the directionality of the intramolecular hydrogen bonds.

An estimate of the vibrational frequencies of the pair decoupled system cannot be straightforwardly done within the harmonic approximation. In fact, while scaling the off-diagonal entries of the Hessian does not change the trace, which is conserved in the diagonalization, it might change the values and order of the eigenvalues. These changes imply that the pair-decoupled Hessian does not correspond to a stationary point configuration anymore. Consider, for instance, the case when the carboxyl group is decoupled from the hydroxyl groups, that is, all the atoms in the carboxyl group are fully decoupled from the O and H atoms in the hydroxyl group. The normal mode analysis of such a system at the original equilibrium geometry shows that the now unhindered Acid twist mode has frequency of about 3650 cm^{-1}. This is clearly not realistic. As mentioned above, the reason why a normal mode analysis can not be employed for an artificially decoupling analysis is that the equilibrium geometry of the system, at which the Hessian matrix is computed, is not a stationary point for the pair-decoupled system. Instead, computing the vibrational spectrum from the velocity correlation function does not suffer from this problem, and it can account for both anharmonicities, non-equilibrium, and dynamical couplings, which are lacking in the harmonic approximation.

Decoupling the O10 – H16 and C1 = O9 stretching modes

In this section we apply the pair-decoupling idea in normal mode coordinates. In particular, we decorrelate the O–H stretching mode from the C=O stretching mode of the Salicylic Acid. Both the O–H and C=O stretchings are localized, meaning that we can interpret O–H and C=O as two oscillators, where the O and H, and C and O atoms are each connected by a spring. Even though the two oscillators are defined as independent when the molecule is at equilibrium, outside of equilibrium the two oscillators are coupled, and each one depends on the other one's displacement. Furthermore, both oscillators also couple with all the other oscillators that compose the Salicylic Acid vibrations. All the observations that we can make about the spectra in Figure 6.5 originate from the in-plane oscillations localized on the Carboxylic Acid group.

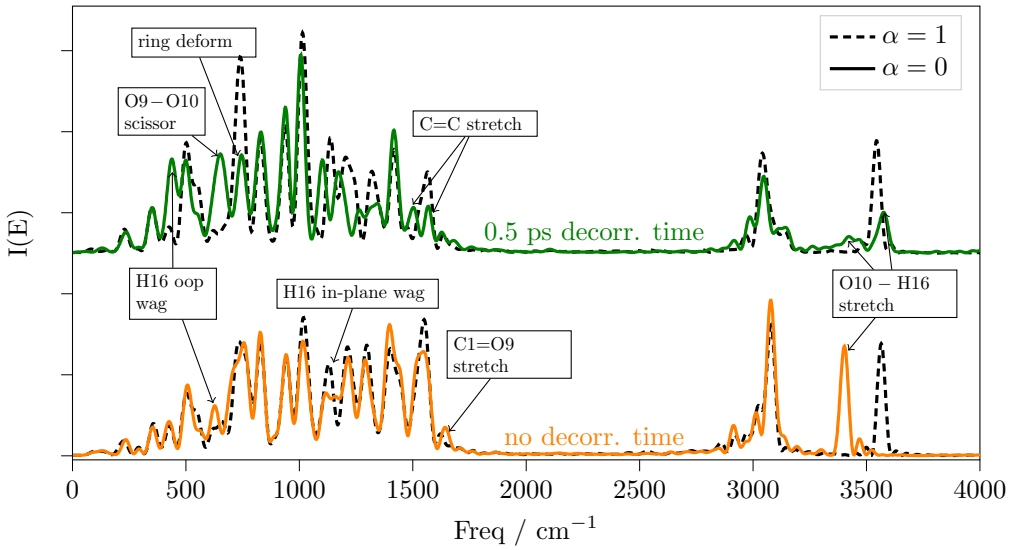


Figure 6.5: Power spectra of the Salicylic Acid (black dashed lines), and of the decoupled ($\alpha = 0$) C1 – O10 and O10 – H16 stretching modes, without waiting any decorrelation time (solid orange line) and after 0.5 ps of decorrelation time (solid green line). The $\alpha = 1$ spectra are taken over the same time intervals of the corresponding $\alpha = 0$ ones, accounting for the decorrelation time.

In Figure 6.5 we see that if we do not wait for any decorrelation time, the anharmonic vibrational frequency of the O–H stretch is redshifted by nearly 160 cm^{-1}. This effect is evidently localized on the O–H, because the rest of the spectrum is only slightly changed by the decoupling. However, after 0.5 ps of decorrelation time, the decoupling also affects the O – C – O scissoring mode, as well as modes that involve deformations and C=C stretchings. Moreover, after 0.5 ps of decorrelation time, the O–H stretching frequency becomes again similar to that of the non-decoupled system. Let us first focus on the bottom part of Figure 6.5. O9 and H16 are connected by a non-directional hydrogen bond, that is weakened when the C=O and O–H bonds are stretched, because the two stretchings move the two atoms further apart. As the two oscillators stretch, they do not retain their reciprocal phase, because of the difference in mass between O and H. However, when we apply the decoupling, atom H16, would keep feeling the effect of a non-stretched C1 – O9 oscillator, while oscillating back and forth. Thus we can see that the O–H stretching is hampered by the stable O9 – H16 hydrogen bond. And we can quantify the importance of this effect by measuring the redshift, which amounts to about 160 cm^{-1}.

Let us now focus on the upper part of Figure 6.5. It shows that after 0.5 ps of decorrelation time, also the normal modes localized on the ring are affected by the decoupling. The involvement of the other modes implies a structural deformation of the entire molecule, compared to the non-decoupled dynamics. This brings the Carboxylic

Acid O – H stretching frequency to about 3570 cm^{-1}, which is slightly blueshifted from the non-decoupled spectrum (dashed line). This effect cannot be explained with simple arguments, because it evidently involves the whole molecule. As a matter of fact, the only portions of the spectrum that appear almost unaffected by the decoupling are C – H and hydroxyl O – H stretching modes, as well as the 700 and 1050 cm^{-1} region, which involves some ring deformation and breathing modes.

Decoupling the carboxyl O9 – H16 Hydrogen bond

The first physical insight provided by the artificially decoupled O9 – H16 Hydrogen bond is that the O9 and H16 atoms do not oscillate synchronously anymore. This asynchronous motion induces an angular momentum that enhances the Acid twist mode to the point that, after less than 300 fs, the carboxyl group attempts a 180 degrees rotation around the C1 – C2 axis. This 180 degrees rotation has a potential barrier (computed as energy of the transition state minus energy of the minimum) in the original fully coupled system of 6366 cm^{-1} at pVDZ/DFT-PBE level of theory, and it should be an extremely rare event for the fully coupled system, considering that the Acid twist motion is initialized with less than 100 cm^{-1} of kinetic energy. To avoid this artificial twist, which is not in a fitted region of the given sGDML potential energy surface, we run simulations where the Acid twist normal mode is kept at equilibrium and the O9 – H16 Hydrogen bond is still decoupled. The power spectrum of this simulation after a 0.5 ps decorrelation time is shown in Figure 6.6. In this case we can observe a rather weak decorrelation effect in terms of the enhanced H16 rock and in-plane wag. The effects of such enhanced motion are given by the more intense bands in the 400 to 500 cm^{-1} and 1100 to 1300 cm^{-1} regions, as well as by the blueshift of the O10 – H16 stretching mode, indicating a slightly weaker bond between O10 and H16.

From a fixed nuclei picture of the Salicylic Acid in the minimum energy configuration, one assumes that the Hydrogen bond between O9 and H16 ensures that the carboxyl group remains confined in a plane, and that H16 is oriented towards O9. In a dynamical picture instead, H16 oscillates out of the O9 – C1 – O10 plane and, given enough energy, it might overcome the potential barrier and get oriented towards C8 in a 180° rotation around the C1 – O10 axis. In the original fully coupled system, this barrier height is about 4355 cm^{-1} at pVDZ/DFT-PBE level of theory, and thus, the 180° rotation around the C1 – O10 axis is a rare event, considering that the H16 oop wagging is initialized with 574 cm^{-1} of kinetic energy. The lack of synchronization in the out-of-plane motion of O9 and H16 redistributes some of the stretching vibrational energy to the out-of-plane modes, to the point that the rotations become allowed. In fact, if one keeps the carboxyl twist mode at equilibrium, the H16 oop wag begins to oscillate significantly after about 1.2 ps of simulation and we observe an attempt of 180

degree rotation of O10 – H16 around the C1 – O10 axis.

To sum up, the simulations of the O9 – H16 decoupled SA provide two main physical insights. First of all, the artificial decoupling allows to appreciate the importance of the synchronous oop vibration without which the carboxyl group would rotate, leading to a less stable minimum configuration. Secondly, the asynchronous motion of O9 and H16 leads to a fast vibrational energy redistribution in favor of the out of plane modes. As a secondary result, we see that the pair decoupling allows to quickly explore otherwise almost forbidden configurational regions of the potential surface.

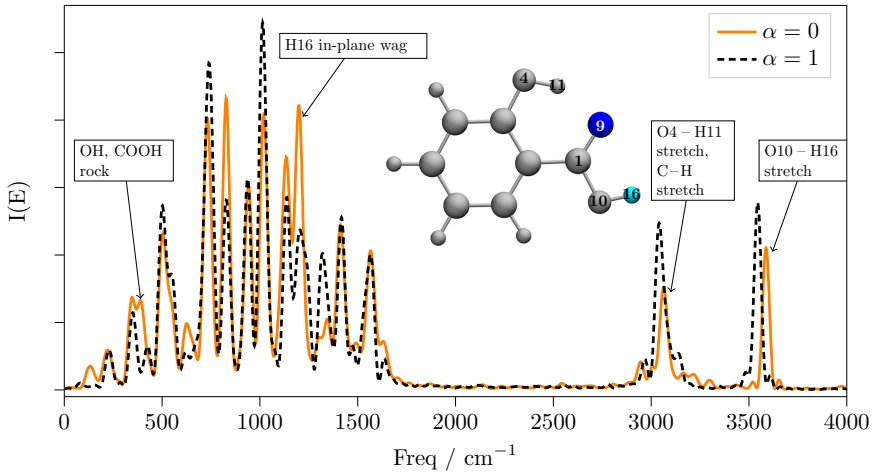


Figure 6.6: Power spectra of the Salicylic Acid (dashed black line), and of the O9 – H16 decoupled ($\alpha = 0$) Salicylic Acid (solid orange line). Both spectra are recorded after discarding the first 0.5 ps of simulation.

Decoupling the hydroxyl and carboxylic acid groups

Here we show with our simulations why the O9 – H11 Hydrogen bond is fundamental for the planar shape of the SA. As a consequence of the O9 – H11 decoupling, the carboxyl group quickly initiates a large amplitude twist around the C1 – C2 axis. This effect is similar to that one we have described in the previous section for the O9 – H11 decoupling, where the decoupling artificially augments the kinetic energy of the acid twist motion represented in panel B) of Figure 6.4. However, in this case, the role of the H-bond is very different. When O9 and H11 are decoupled, we argue that, even if the out-of-plane motions are not synchronized anymore, the oop wag of H11 remains coupled to O10 (and H16) and this coupling stimulates the oop motion of O10 (given the planarity of the carboxyl group). Eventually, the artificially enhanced oop wagging of H11 induces an attempted 180° rotation of the carboxyl group around the C1 – C2 axis. We deduce that there must be a strong synchronized interaction of each atom composing the *whole* hydroxyl group with each atom composing the *whole* carboxyl

group. This interpretation of the importance of the H11 interaction with each singular atom composing the carboxyl group is validated by the fact that the acid twist motion is not enhanced when the whole carboxyl group is decoupled from the whole hydroxyl group. In Figure 6.7 we show both the spectrum when the O9 – H11 interaction is decoupled and the acid twist mode is kept at equilibrium, and also the spectrum when it is the carboxyl – hydroxyl entire groups to be decoupled. Both spectra are recorded after 0.5ps of decorrelation time and both spectra show that the decoupling effect is quite significant in terms of vibrational energy redistribution. In fact, the ring breathing and ring deformation modes donate vibrational energy to the out-of-plane and C=O stretching modes at about 100 to 400, and 1600 cm^{-1} respectively. More specifically, both the O9 – H11 (orange line) and the carboxyl – hydroxyl decoupled spectra (green line) show that the decouplings induce significant vibrational energy redistributions in both the low frequency and fingerprint regions of the spectra, especially from the ring breathing and ring deformation modes, while the ring C–H and O–H stretching signals retain their kinetic energy on average. In both the O9 – H11 and carboxyl – hydroxyl decouplings, the O–H stretching signals are mildly blueshifted, indicating slightly weaker hydrogen bonds. In conclusion, these results clearly show that the intuitive picture of the independent functional groups in ortho position on the aromatic ring is partial to describe the appropriate vibrational dynamics of the SA and that the single atom-atom instantaneous couplings are essential for an accurate description of the interactions between the two functional groups. Our results also show that, surprisingly, the hydroxyl – carboxyl decoupled system provides a more realistic simulation of the O9 – H11 decoupled one, because it does not induce the acid twist rotation.

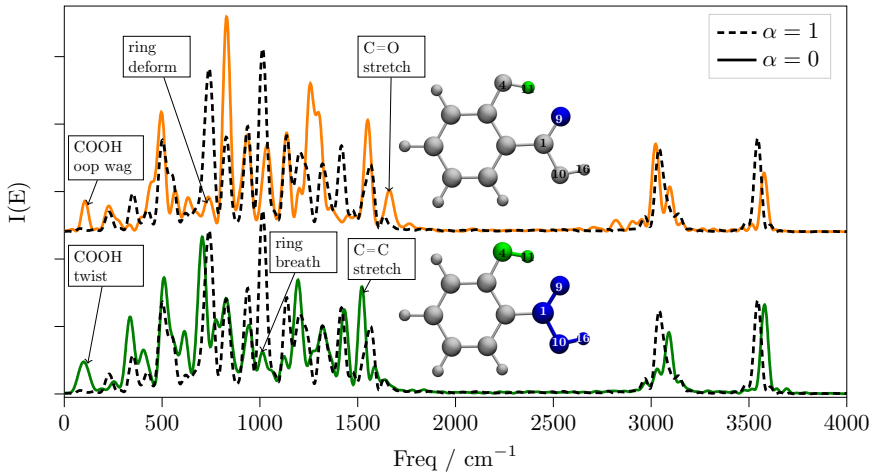


Figure 6.7: Power spectrum of the O9 – H11 (solid orange line), and carboxyl – hydroxyl (solid green line) decoupled SA. The black dashed line is the power spectrum of the fully coupled simulation for comparison. The O9 – H11 simulation (orange line) was performed without evolving the acid twist mode and both spectra are recorded after a 0.5ps decorrelation time.

Decoupling the carboxyl and ring fragments

We find that the strongest decorrelation effects that impacts the carboxyl O–H stretching mode occurs when we decouple the motion of the ring from that of the carboxylic acid group. Specifically, the stretching O–H mode of the carboxylic acid is blueshifted by 103 cm^{-1} (from 3573 cm^{-1} of the fully coupled system to 3675 cm^{-1} of the $\alpha = 0$ decoupled system), as shown by the solid orange line in Fig. 6.8. This effect is accompanied by an increased amplitude H16 oop wagging motion, which, shortly after about 0.7ps induces a 180° rotation of the O–H around the C1 – O10 axis. If we keep the H16 oop wag at its equilibrium geometry to avoid this artificial rotation and record the power spectrum after 0.5 ps of decorrelation time, the O–H stretching of the acid is still blueshifted, although only by about 50 cm^{-1}, as shown by the solid green line of Fig. 6.8. We conclude that the ring and carboxyl group decoupling has a strong effect on the carboxyl O–H motion. In fact, the C–H and hydroxyl O–H stretching signals around 3100 cm^{-1} are split, although no one of those hydrogens is decoupled.

The ring-carboxyl decoupling is a good example of how the two fragments can be interpreted as independently vibrating fragments just after the decoupling effect is turned on, but not after 0.5ps of decorrelation time. As a matter of fact, after the decorrelation time, the carboxyl O–H stretching signal is smeared over nearly 500 cm^{-1}, mainly because of the enhanced oop wagging motion of the carboxyl group at 150 cm^{-1}. Nonetheless, most of the signals that involve ring stretchings and other motions delocalized over the ring are still well recognizable in the fingerprint region of

the spectrum, even after the decorrelation time. This observation shows that the effects of substituents on the vibrational features of the ring is mostly static, i.e. decorrelating the ring and carboxyl group vibrations do not induce very significant changes in the ring vibration frequencies. In fact, the fingerprint region of the spectra, between 700 and 1500 cm^{-1}, displays only some mild vibrational energy redistribution, in favours of the COOH twist and oop wag.

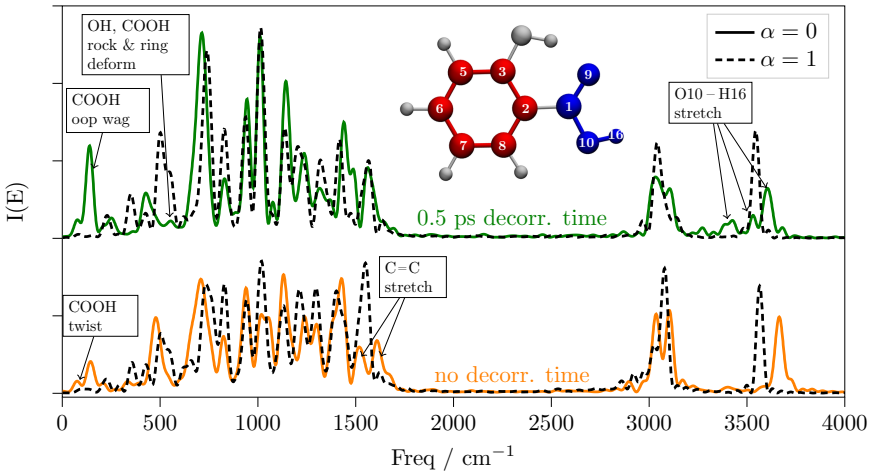


Figure 6.8: Spectrum of the SA with the Carbon ring decoupled from the carboxyl group with $\alpha = 0$, without waiting any decorrelation time (orange line) and after 0.5ps of decorrelation time (green line). The dashed black lines are the spectra of the fully coupled system after discarding the corresponding amount of simulation time to match the decorrelation time.

6.3.3 Decoupling the entire SA into a ring part and its substituents

We conclude the results section by describing the scenario where the SA molecule is decomposed into a ring and its substituents.

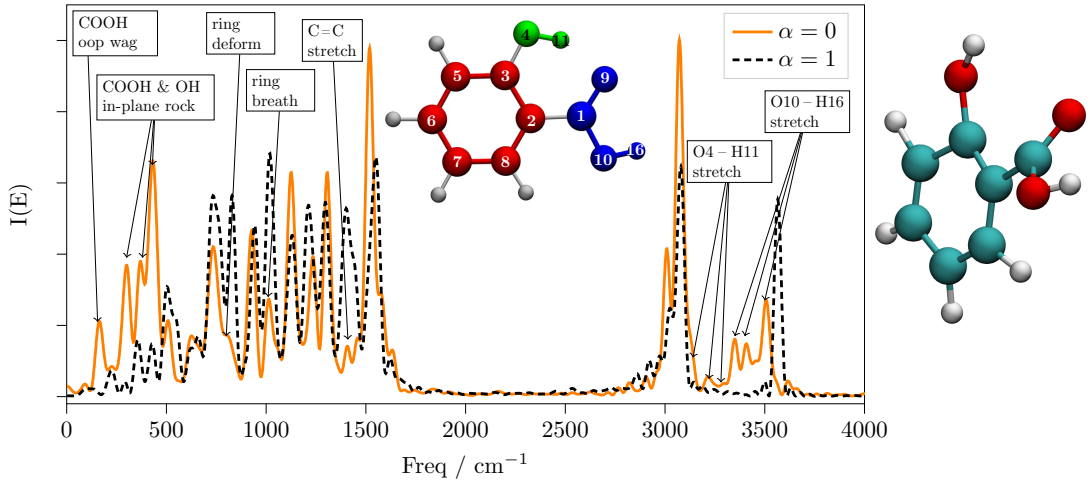


Figure 6.9: Spectrum of the SA with the Carbon ring, carboxyl, and hydroxyl fragments all decoupled from each other, with $\alpha = 0$ (orange line). The dashed black line is the spectrum of the fully coupled system for comparison. The inset on the right shows a representative frame of the simulation, as a result of the asynchronous motion of the three decoupled fragments.

In Figure 6.9 is shown in orange the power spectrum of the SA where its functional groups are artificially decoupled. These groups are the aromatic Carbon ring, the hydroxyl group, and the carboxyl group. After only 180 fs the decoupled system attempts a rotation of the O4 – H11 group around the C3 – O4 axis. Therefore, in this case, we decided to keep at the equilibrium position the mode which involves simultaneously the H11, ring, acid oop displacements, which is indicated in panel E) of Figure 6.4. Then, we record the spectrum without waiting for any decorrelation time. The strongest decorrelation effects observed in this dynamics consists in the enhanced acid twist, acid oop wag, and O–H oop wags, represented in panels B), C), and D) of Figure 6.4. All these effects quickly bring the molecule into very energetic regions of the PES, such as shown in the inset of Figure 6.9. From Figure 6.9 we can also see the same smearing of the carboxyl O–H stretching signal observed in the green spectrum of Figure 6.8, which occurs without any decorrelation time. This smearing effect is mainly due to the fact that the acid twists to a staggered position and simultaneously it also bends towards the ring (see the inset of Figure 6.9). In such a distorted configuration, the carboxyl O–H stretching motion strongly depends on the orientation of the O–H, as well as on the amount of twisting. In addition, the hydroxyl group is anomalously stretched apart from the ring and the C1 – O10 – H16 angle is highly increased, and similarly the signals of the hydroxyl O–H stretchings in the 3100 to 3300 cm^{-1} frequency interval are smeared. On the contrary the C–H stretching signals at about 3000, and 3100 cm^{-1} are only slightly shifted, but not smeared. The redshift of the carboxyl

stretching can also be interpreted from the distorted geometry of the SA. When the carboxyl group is twisted, its electron withdrawing effect on the ring is weakened and it becomes a weaker acid.

Analogously to what we observed in the ring – carboxyl decoupling of Figure 6.8, the fingerprint region of the spectrum remains reasonably similar (in terms of frequencies) to that of the fully coupled system. However, some of the ring modes, in particular the C=C stretchings, ring deformation, and breathing modes, donate vibrational energy to the hydroxyl and carboxyl groups in the 100 to 500 cm^{-1} region of the spectrum.

6.4 Discussion and Conclusions

In this work we introduce a pair-decoupling idea that offers a novel perspective for the study of relationships among groups of atoms or, more generally, of degrees of freedom in a molecule. The pair-decoupling idea is based on a simple, yet always applicable, mathematical definition: the Hessian matrix of a pair decoupled system is equal to the Hessian matrix of a normal system where some of the off-diagonal elements are scaled by an arbitrary coefficient α. The SEF algorithm that we introduce enforces the pair-decoupling idea for the molecular dynamics simulations of small and middle sized organic molecules. The simulations preserve faithfully the properties of symplectic symmetry of classical dynamics, in particular the phase space conservation, in agreement with Liouville's theorem. The SEF method is effectively a symplectic integration technique of a system under a locally harmonic, "pair-decoupled" potential. The main disadvantage of the method is the requirement of 2 or 4 Hessian matrix calculations per time-step. However, this limitation could be alleviated by suitable numerical techniques.[109, 110] This unavoidable feature limits the employment to middle-size molecules and imposes the use of a computationally affordable potential for the electronic structure.

The application of our technique to the Salicylic Acid has shown both intuitive behaviors of the pair-decoupled system, such as the rotation of the carbonyl in response to a decoupling of the hydroxyl-carboxyl H-bond, and less intuitive and surprising effects, such as the blueshift of the carboxyl O−H stretching frequency when the acid Hydrogen is decoupled from the aromatic ring. We also showed that the synchronous vibrations of the atoms in the carboxyl and hydroxyl fragments is essential for the equilibrium configuration to be stable over time. As a consequence, in absence of such couplings, the proton transfer photochemistry of the Salicylic Acid would be impossible. Ultimately, our simulations of the pair-decoupled Salicylic Acid shows that the picture of the molecule as composed of independent vibrating fragments is partial, and often unreliable. Since this intuitive picture is at the origin of the functional groups definition, we think that these results show how there are important exceptions to the functional group picture. In fact, an artificial decoupling of apparently unrelated groups of atoms

may induce evident changes to the vibrational spectroscopy of the whole molecule. We think that these considerations are applicable to many other chemical systems, and that our results open the path to further investigations thanks to the computational tool that we have presented. Furthermore, the pair-decoupled simulation technique can be used to validate applications that assume that a portion of the system is partially uncoupled from another, such as in MCTDH and QM/MM calculations. The most practical way to do that in the case of QM/MM, for example, is to simulate the chosen pair-decoupled system at MM level, and verify whether it is an appropriate partition for the QM/MM calculation.

We hope that the pair-decoupling idea can inspire other less computationally expensive methods that can asses the importance of couplings in molecules. Finally, we believe that the SEF algorithm can help increase the sensibility of chemists towards the (unexpected) effects of approximations that involve artificially decoupled systems.

Chapter 7

DC-SCIVR Exact Decomposition

As described in section 2.2.1, the Divide and Conquer method for semiclassical spectroscopy is not an exact decomposition of the full dimensional spectrum. In this chapter we show how to use the same numerical trick employed for the pair-decoupling integration algorithm to obtain an exact decomposition of the full dimensional potential over an arbitrary collection of vibrational degrees of freedom.

The integration rule for the SEF4 algorithm (see Eq. 6.4) is not limited to integrate second derivatives. It can be used to integrate any derivative of the potential. Especially, we are interested into integrating the force to obtain the potential energy as a sum of contributions coming from each degree of freedom. In the kth step of the symplectic map one can compute

$$\begin{aligned} V(t_k) &= V(t_{k-1}) - c_k\tau\left(F(q_{aux})\cdot\frac{p(t_k)}{m}\right) \\ &= V(t_{k-1}) - c_k\tau\sum_j F_j(q_{aux})\frac{p_j(t_k)}{m_j}, \end{aligned} \tag{7.1}$$

where the index j runs over the nuclear degrees of freedom. Eq. 7.1 shows that one can separate the potential contributions from each degree of freedom, provided that the initial potential of the nuclei is set to 0. Thus, one can collect the partial potentials for an arbitrary partition of degrees of freedom, such that the partial potentials sum up to the total potential. For instance, the potential contribution from the j^{th} degree of freedom is computed as

$$V_j(\tau) = V_j(0) - \sum_k c_k\tau F_j(q_{k,aux})\frac{p_j(t_k)}{m_j} \tag{7.2}$$

with

$$V(\tau) = \sum_j V_j(\tau). \tag{7.3}$$

The sum property in Eq. 7.3 makes the partial potentials $V_j(\tau)$ (or any partial sum of them) a numerically exact candidate for the projected potentials in the DC-SCIVR method. From a practical point of view, the TA-SCIVR formula of Eq. 2.8 is the phase-space average of the squared modulus Fourier transform of the time-dependent function $g(t) = e^{\frac{i}{\hbar}[S(t)+\phi(t)]} \langle \chi | p(t), q(t) \rangle$. For the Time Averaged-SCIVR spectrum to be factorizable, we need to write $g(t)$ as a product of functions that depend only on the variables of the subsystems, as

$$g(t) = \prod_{j=1}^{M} g_j(t). \tag{7.4}$$

Whenever this factorization is possible, $I(E)$ can be computed as the convolution of the $g_i(t)$, as per the convolution theorem.

$$I(E) = \frac{1}{T} \left(g_1(t) \otimes g_2(t) \otimes g_3(t) \dots \right)^\star \cdot \left(g_1(t) \otimes g_2(t) \otimes g_3(t) \dots \right), \tag{7.5}$$

where $\otimes$ represents the convolution product and $\star$ the complex conjugate. The factorization of Eq. 7.4 means that the power spectrum is of the type

$$I(E) \quad \propto \quad \frac{1}{T} \left| \int_0^T e^{i\frac{E}{\hbar}t} e^{\frac{i}{\hbar}[\sum_j S_j(t) + \sum_j \phi_j(t)]} \prod_j \langle \chi_j | p_j(t), q_j(t) \rangle \, dt \right|^2 \tag{7.6}$$

where Γ_j is a diagonal block of the Γ matrix, p_j, q_j are the j components of the corresponding vectors and S_j and ϕ_j are the contributions to S and ϕ coming from the j^{th} degree of freedom. We must point out that the phase of the prefactor cannot be decomposed as a sum of contributions, that is $\phi(t) \neq \sum_j \phi_j(t)$. The action can be expressed as a sum of the degrees of freedom components, as long as the potential is integrated along the simulation, with the symplectic propagation algorithm as

$$V_j(q(t)) = \int_0^t \frac{\partial V(q(t'))}{\partial q_j} \dot{q}_j(t') dt'. \tag{7.7}$$

Eq. 7.7 is numerically very precise when used within the symplectic map of order 4, implemented as in Eq. 7.2, and using the c_k coefficients in Table 6.1.

7.1 Harmonic Oscillators

I consider the model Hamiltonian

$$H = \frac{\mathbf{p}^2}{2} + \frac{1}{2} \mathbf{q}^T \cdot \Omega \cdot \mathbf{q}. \tag{7.8}$$

with

$$\Omega = \begin{pmatrix} 2500^2 & 1000^2 & 0 & 0 \\ 1000^2 & 3000^2 & 0 & 0 \\ 0 & 0 & 3500^2 & 1000^2 \\ 0 & 0 & 1000^2 & 4000^2 \end{pmatrix}, \tag{7.9}$$

where Ω is in cm^{-1}. The matrix Γ is the diagonal matrix of ordered square root eigenvalues of Ω

$$\Gamma \approx \begin{pmatrix} 2434 & 0 & 0 & 0 \\ 0 & 3054 & 0 & 0 \\ 0 & 0 & 3464 & 0 \\ 0 & 0 & 0 & 4031 \end{pmatrix} \tag{7.10}$$

Clearly Eq. 7.8 is separable into two subsystems, each of which is a 2D coupled system. The spectra for all the possible decompositions of Eq. 7.8 into lower dimensional Hamiltonians are given in Fig. 7.1.

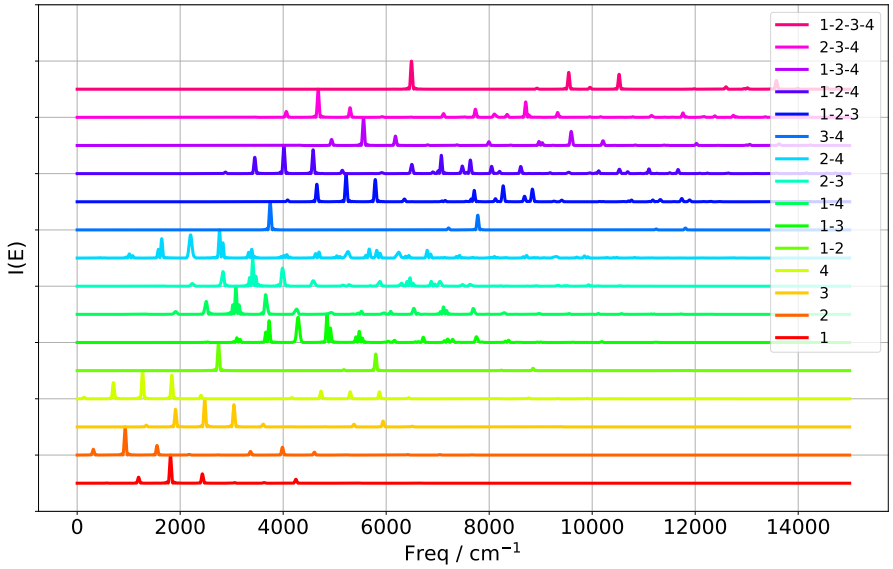


Figure 7.1: All possible DC-SCIVR spectra of 4 harmonic oscillators as in Eq. 7.8. The oscillators contributing to each spectrum are shown in the legend

In Fig. 7.1 the $1-2$, $3-4$ and $1-2-3-4$ spectra are exact, because the system is exactly separable in these groups. All the other spectra are split into three or more signals, which depend on the differences in the square root eigenvalues of Ω. For instance the separation from the two side peaks of the spectra 1 and 2 (in red and in orange)

correspond to $\Gamma_{22} - \Gamma_{11} \approx 620$ cm^{-1}. While the separation from the two side peaks of the spectra 3 and 4 (in ochre and yellow) are $\Gamma_{44} - \Gamma_{33} \approx 567$ cm^{-1}. There are two separation from the side peaks (from the central one) in spectrum $1-3$, with values 561 and 628 cm^{-1}, which are close to be, respectively $\Gamma_{44} - \Gamma_{33}$ and $\Gamma_{22} - \Gamma_{11}$. The second cluster of separation are three signals, distant from the central one approximately by $2(\Gamma_{44} - \Gamma_{33})$, $2(\Gamma_{22} - \Gamma_{11})$ and $(2(\Gamma_{44} - \Gamma_{33}) + 2(\Gamma_{22} - \Gamma_{11}))/2$. Similar behaviors appear in all the other spectra with same interpretations.

7.2 Water molecule

The isolated water molecule[180] has one bending mode of vibration and two stretching modes. The two stretching modes are strongly coupled, while the bending is relatively uncorrelated from them.

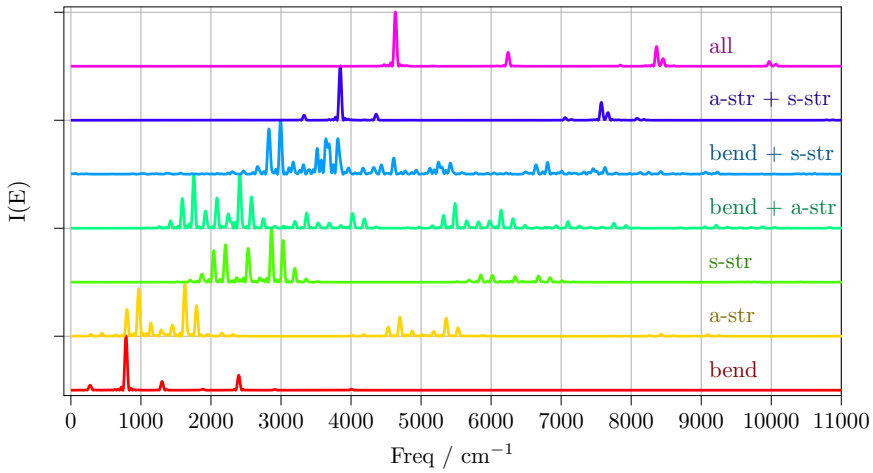


Figure 7.2: All the possible DC-SCIVR spectra of the water molecule

Clearly, from Figure 7.2 we can see that the partial spectra of the bending mode (red in the figure) and that of the two stretching modes (purple in the figure) are the least coupled. This result is consistent with both experience and what we find with the DC-SCIVR techniques for the identification of the modes subspaces. [49, 51] The main advantage of this new definition of the projected potentials is that the frequencies of the signals of the partial spectra sum up to the frequencies of the full dimensional spectra. For instance the first central signal in the red spectrum (787 cm^{-1}) and the first central signal in the purple spectrum (3846 cm^{-1}) sum up exactly to the ZPE signal of the pink spectrum (4633 cm^{-1}). The same argument applies also to other partitions of the degrees of freedom and to the other energy levels, which means that energy of full dimensional eigenstate is exactly separated into the energies of the lower dimensional subspaces. The main disadvantage of this approach is that the subspaces

are now all coupled, and it may become difficult to correctly interpret the spectra of the strongly coupled modes, due to the abundance of coupling signals.

7.3 Derivatives of the partial potentials

The reason why the definition of the projected potentials given in Eq. 7.7 is exact, compared to their original definition can be sought after in the potential derivatives. The partial potential V_j in in Eq. 7.7 depends on all the coordinate variables q_k with $k \neq j$. In fact, if you take the derivative of the partial potentials w.r.t. q_k, you get

$$\frac{\partial V_j}{\partial q_k} = \int_0^t \frac{\partial^2 V(q(t'))}{\partial q_j \partial q_k} \dot{q}_j(t') dt'. \tag{7.11}$$

In the particular case with $k = j$ you get a second derivative depending only on the j^{th} degree of freedom. The force acting on the k degree of freedom is

$$\begin{aligned} -F_k &= \int_0^t \sum_j \frac{\partial^2 V(q(t'))}{\partial q_k \partial q_j} \dot{q}_j(t') dt' && (7.12) \\ &= \sum_j \int_0^t \frac{\partial^2 V(q(t'))}{\partial q_k \partial q_j} \dot{q}_j(t') dt', && (7.13) \end{aligned}$$

to be compared with Eq. 7.11. Clearly, all the contributions to the force acting on the degree of freedom k, that is F_k, must account for all the second derivatives of V_j w.r.t. the degrees of freedom. Also, note that the vector form of Eq. 7.13 is the recipe to get the gradients from the hessians given for the pair-decoupling integration method.

On the contrary, the projected potentials defined in the DC paper [49] distinguish between modes within the subspace, $\tilde{q}$, and modes outside the subspace $\hat{q}$,

$$V_S(\tilde{q}) = V(\tilde{q}; \hat{q}) - V(\tilde{q}^{eq}; \hat{q}), \tag{7.14}$$

If you take the derivatives of this projected potential you get, of course, that the derivative depends on the mode.

$$\begin{cases} \dfrac{\partial V_S}{\partial \tilde{q}} = \dfrac{\partial V}{\partial \tilde{q}} (\tilde{q}; \hat{q}) \\ \\ \dfrac{\partial V_S}{\partial \hat{q}} = \dfrac{\partial V}{\partial \hat{q}} (\tilde{q}; \hat{q}) - \dfrac{\partial V}{\partial \hat{q}} (\tilde{q}^{eq}; \hat{q}) \,, \end{cases} \tag{7.15}$$

where you can see that the derivatives are different, depending if the coordinate to be derived is inside or outside the subspace. The top one (inside) is exactly the potential derivative w.r.t. $\tilde{q}$. The bottom one, on the other hand becomes zero when the $\hat{q}$ and

$\tilde{q}$ degrees of freedom are independent. When $\hat{q}$ and $\tilde{q}$ are not independent, the force acting on $\tilde{q}$ that comes from $\hat{q}$ is changed because of the second term in the bottom equation.

Chapter 8

Dynamical Coupling of Organic Molecules

As described in chapter 6, we define a decoupled system as a system whose second partial derivatives of the potential with respect to the system coordinates are scaled by a factor α, that is

$$\frac{\partial^2 \tilde{V}}{\partial \tilde{q}_i \partial \tilde{q}_j} = \alpha \frac{\partial^2 V}{\partial q_i \partial q_j}. \tag{8.1}$$

Because this definition imply a modification of the potential energy function, the decoupled system, if evolved in time, would follow a different path from the normal system. Thus, the variables $\tilde{q}$ and $\tilde{p}$, representing the position and momentum of the pair-decoupled system, form their own phase space, which depends on the decoupling parameter α. In the special case $\alpha = 1$ the phase space of the decoupled system is superimposed to that of the normal system. If we apply a small pair-decoupling perturbation, then the system would move slowly into the pair-decoupled phase space, and we expect a change in total energy in response to that. This creates the opportunity to learn what are the pairs of degrees of freedom that are most influenced by the decoupling. In particular, we can measure how much energy is lost (or gained) by the system when a small pair-decoupling perturbation is applied for a short time. The basic idea behind this analysis is to measure the difference in energy of the molecule when it undergoes the normal and the decoupled propagation. When the selected pair of atoms are strongly coupled, we may expect a substantial change in energy because of the decoupling, while for uncoupled pairs we expect no change at all. In particular, we are interested into the signless energy difference, as we would like to quantify the importance of the coupling, rather than understand whether the energy is gained or lost. Therefore, we measure the absolute value of the energy difference averaged over a statistical ensemble, and we call this quantity the Dynamical Coupling Index (DCI). The idea behind the DCI measurement is represented graphically in Figure 8.1.

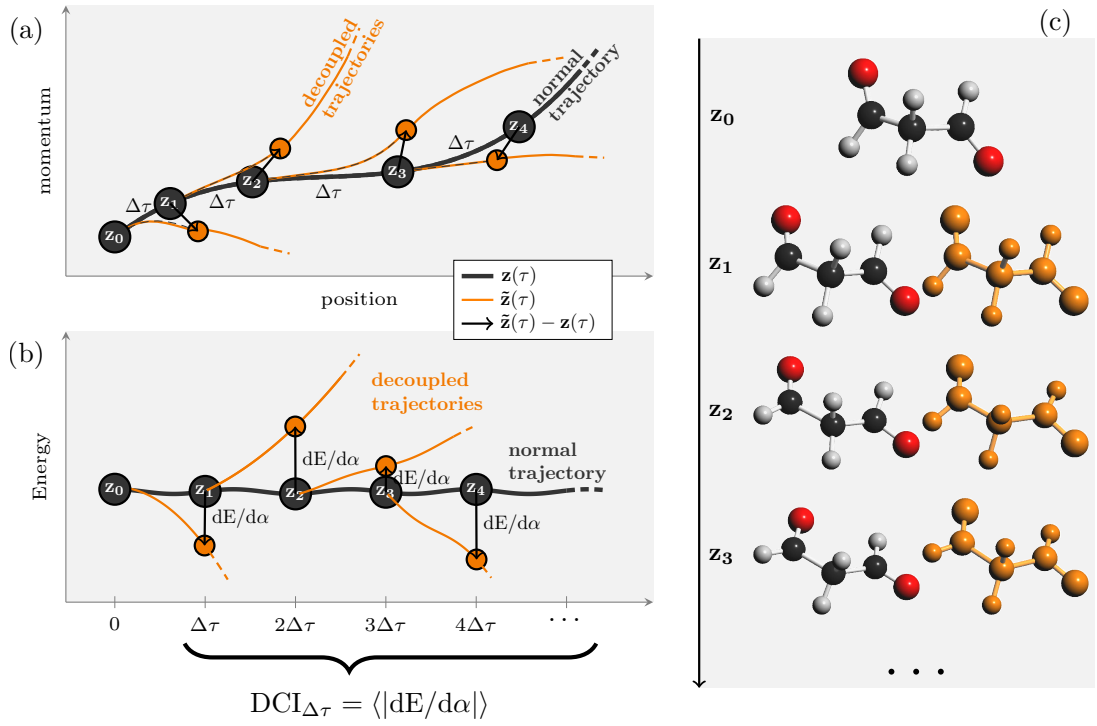


Figure 8.1: Pictorial representation of DCI, computed as the absolute difference in energy between the pair-decoupled system and the normal system after a small time interval and averaged over a statistical ensemble. Panel a) shows the normal phase space trajectory in black, with frames z_0, z_1, ..., and the corresponding frames of the decoupled trajectories in orange. Each frame of the pair decoupled trajectory originates from the normal trajectory frames after a time lapse equal to $\Delta\tau$. Panel b) shows the energy profiles of the normal and pair-decoupled trajectories, where the DCI is simply the average of the absolute value energy difference. Panel c) shows some example geometries of the normal and pair-decoupled configurations.

The calculation of DCI using the naive procedure shown in Fig. 8.1 and the SEF algorithm is possible, but very computationally expensive. To get an easily applicable formula to measure the DCI we apply Kubo's linear response function formula to the perturbed equations of motion, as described in the next section.

8.1 Theory of Decoupling Energy Loss

We begin with the generalized classical equations of motion for the pair-decoupled system

$$\begin{cases} \dot{\tilde{q}} = \dfrac{\partial \mathcal{H}}{\partial p} = \dot{q}_0(t) + \Delta\dot{q}(t) \\ \dot{\tilde{p}} = -\dfrac{\partial \mathcal{H}}{\partial q} = \dot{p}_0(t) + \Delta\dot{p}(t) \end{cases}, \tag{8.2}$$

where $\mathcal{H} = \mathcal{H}_0 + \Delta\mathcal{H}$ is the system Hamiltonian with coordinates and conjugated momenta $\tilde{q}$ and $\tilde{p}$. $\Delta\dot{q}(z)$, and $\Delta\dot{p}(z)$ are the (small) perturbations in the canonical variables. The formal solution of Eq. 8.2 is the Dyson series

$$z(t) = \mathcal{T}\int_0^t e^{i\hat{L}t'} z(0) dt', \tag{8.3}$$

where $\mathcal{T}$ is the time ordering operator and the operator $-\{\mathcal{H}, \cdot\} = i\hat{L}$ is simply the Liouville operator for the unperturbed system $z = (q, p)$ plus the Liouville operator for the perturbation variables, that is

$$\begin{aligned} i\hat{L} = -\{\mathcal{H}, \cdot\} &= -\frac{\partial\mathcal{H}}{\partial q}\frac{\partial}{\partial p} + \frac{\partial\mathcal{H}}{\partial p}\frac{\partial}{\partial q} \\ &= (\dot{p}_0(t) + \Delta\dot{p}(t))\frac{\partial}{\partial p} + (\dot{q}_0(t) + \Delta\dot{q}(t))\frac{\partial}{\partial q} \\ &= \dot{z}_0(t)\cdot\nabla_z + \Delta\dot{z}(t)\nabla_z \\ &= i\hat{L}_0 + i\Delta\hat{L} \\ &= -\{\mathcal{H}_0, \cdot\} - \{\Delta\mathcal{H}, \cdot\}, \end{aligned} \tag{8.4}$$

where ∇_z is the derivative with respect to the positions and conjugated momenta. Eq. 8.4 implies that $\partial\Delta\mathcal{H}/\partial q = -\Delta\dot{p}$ and $\partial\Delta\mathcal{H}/\partial p = \Delta\dot{q}$.

The linear response $\langle\mathcal{O}\rangle_{lin}(t)$ of the observable $\mathcal{O}$ to a time-dependent perturbation in the microcanonical ensemble at energy E_0 is given by Kubo's formula[194]

$$\begin{aligned} \langle\mathcal{O}\rangle_{lin}(t) &= \langle\mathcal{O}\rangle_0 - \int_0^t \left\langle \left\{\Delta\mathcal{H}(z(t')), \mathcal{O}(z(t))\right\}\right\rangle_{\mathcal{H}_0=E_0} dt' \\ \langle\Delta\mathcal{O}\rangle_{lin}(t) &= -\int_0^t \left\langle \frac{\partial\Delta\mathcal{H}(t')}{\partial q}\frac{\partial\mathcal{O}(t)}{\partial p} - \frac{\partial\Delta\mathcal{H}(t')}{\partial p}\frac{\partial\mathcal{O}(t)}{\partial q}\right\rangle_{\mathcal{H}_0=E_0} dt' \\ &= -\int_0^t \left\langle -\Delta\dot{p}(t')\frac{\partial\mathcal{O}(t)}{\partial p} - \Delta\dot{q}(t')\frac{\partial\mathcal{O}(t)}{\partial q}\right\rangle_{\mathcal{H}_0=E_0} dt' \\ &\text{given } \mathcal{O} = \mathcal{H}_0 \\ \langle\Delta\mathcal{H}_0\rangle_{lin}(t) &= -\int_0^t \left\langle -\Delta\dot{p}(t')\ \dot{q}_0(t) + \Delta\dot{q}(t')\ \dot{p}_0(t)\right\rangle_{\mathcal{H}_0=E_0} dt' \end{aligned} \tag{8.5}$$

Eq. 8.5 tells us that the response of the system in terms of energy change because of some perturbation is the sum of two correlation functions, each containing the displaced velocities and natural forces and the natural velocities and displaced forces.

In Eq. 8.5 the only α-dependent variables are $\Delta\dot{p}(z(t'))$ and $\Delta\dot{q}(z(t'))$. If we take

the derivative of the response w.r.t. α, we get

$$\begin{aligned}
\frac{\partial}{\partial\alpha}\left\langle\Delta\mathcal{H}_0\right\rangle_{lin}(t) &= -\int_0^t \left\langle -\dot{q}_0(t)\frac{\partial\Delta\dot{p}}{\partial\alpha}(t') + \dot{p}_0(t)\frac{\partial\Delta\dot{q}}{\partial\alpha}(t')\right\rangle_{\mathcal{H}_0=E_0} dt' \\
&= \left\langle \dot{q}_0(t)\int_0^t \frac{\partial\Delta\dot{p}}{\partial\alpha}(t')dt'\right\rangle_{\mathcal{H}_0=E_0} - \left\langle \dot{p}_0(t)\int_0^t \frac{\partial\Delta\dot{q}}{\partial\alpha}(t')dt'\right\rangle_{\mathcal{H}_0=E_0} \\
&= \left\langle \dot{q}_0(t)\frac{\partial}{\partial\alpha}(\Delta p(t)-\Delta p(0))\right\rangle_{\mathcal{H}_0=E_0} - \left\langle \dot{p}_0(t)\frac{\partial}{\partial\alpha}(\Delta q(t)-\Delta q(0))\right\rangle_{\mathcal{H}_0=E_0} \\
&= \left\langle \dot{q}_0(t)\frac{\partial\Delta p}{\partial\alpha}(t)\right\rangle_{\mathcal{H}_0=E_0} - \left\langle \dot{p}_0(t)\frac{\partial\Delta q}{\partial\alpha}(t)\right\rangle_{\mathcal{H}_0=E_0}, \qquad (8.6)
\end{aligned}$$

where in the last equation we assumed that at time $t = 0$ the perturbation is null and independent of α. Now we are left to find a way to evaluate $\frac{\partial\Delta p}{\partial\alpha}(t)$, and $\frac{\partial\Delta q}{\partial\alpha}(t)$ in terms of variables that we can manipulate.

To do that we simply expand $\tilde{p}(t)$ and $\tilde{q}(t)$ around $t = 0$, we express the force as time integral of the Hessian and take the α derivative. This procedure gives the same α dependence of $\tilde{q}$ and $\tilde{p}$ as the SEF algorithm.

$$\begin{aligned}
\frac{\partial}{\partial\alpha}\tilde{p}(t) &= \frac{\partial}{\partial\alpha}\left[\tilde{p}(0)+\dot{\tilde{p}}(0)t+\ddot{\tilde{p}}(0)\frac{t^2}{2}+\dddot{\tilde{p}}(0)\frac{t^3}{6}+\dots\right] \\
&= \frac{\partial}{\partial\alpha}\left[-\frac{\partial^2\tilde{V}}{\partial q^2}(0)\dot{q}_0(0)\frac{t^2}{2}+\left(-\frac{\partial\tilde{V}^3}{\partial q^3}(0)\dot{q}_0^2(0)+\frac{1}{m}\frac{\partial^2\tilde{V}}{\partial q^2}(0)\frac{\partial V}{\partial q}(0)\right)\frac{t^3}{6}+\dots\right] \qquad (8.7)
\end{aligned}$$

$$\begin{aligned}
\frac{\partial}{\partial\alpha}\tilde{q}(t) &= \frac{\partial}{\partial\alpha}\left[\tilde{q}(0)+\dot{\tilde{q}}(0)t+\ddot{\tilde{q}}(0)\frac{t^2}{2}+\dddot{\tilde{q}}(0)\frac{t^3}{6}+\dots\right] \\
&= \frac{1}{m}\int_0^t \frac{\partial}{\partial\alpha}\tilde{p}(t) \\
&= \frac{\partial}{\partial\alpha}\left[-\frac{1}{m}\frac{\partial^2\tilde{V}}{\partial q^2}(0)\dot{q}_0(0)\frac{t^3}{6}+\dots\right] \qquad (8.8)
\end{aligned}$$

Putting the third order approximation of Eq. 8.7 and Eq. 8.8 into Eq. 8.6 gives the approximate expression of the response,

$$\left\langle\frac{\partial}{\partial\alpha}\Delta\mathcal{H}_0\right\rangle \approx \left\langle -\frac{\partial}{\partial\alpha}\left(\frac{\partial^2\tilde{V}}{\partial q^2}\right)\dot{q}_0^2\frac{t^2}{2}\right\rangle_{\mathcal{H}_0=E_0} \qquad (8.9)$$

where the derivatives of the potential of order 3 and higher have been neglected and the other powers of the time of order 3 sum up to zero. If we include further terms in Eq. 8.7 and Eq. 8.8 we obtain further corrections of Eq. 8.9.

Eq. 8.6 (and its approximate versions, such as Eq. 8.9) corresponds to the ensemble average of the energy change after a small time τ, where the energy change is caused by an arbitrary decoupling perturbation. Since we are mostly interested into the absolute

change in energy, we take absolute value of quantity inside the braket. In the limit of no decoupling and short time, we call such quantity the *Dynamical Coupling Index*

$$\mathrm{DCI}_\tau = \lim_{\alpha\to 1} \left\langle \left| \frac{\partial}{\partial\alpha} \Delta\mathcal{H}_0 \right| \right\rangle (\tau). \tag{8.10}$$

The DCI depends on the perturbation time τ, which must be a short time, for the approximations to hold. For all the calculations presented in this work we choose τ to be equal to the time step of the simulation, which we set to 10 a.u. In more formal words DCI=DCI_{10au}. Since the DCI measures the energy lost because of a decoupling perturbation, the immediate interpretation is that the greater the DCI, the greater the dynamical dependence of the decoupled degrees of freedom.

8.2 PES details

For all our calculations we use the sGDML potential energy surfaces[181, 182] described in Table 8.1. The surfaces are provided alongside the sGDML software [181–187] and the training and test sets used to build them. Each PES is given already trained on 1000 training points and has a MAE of less than 0.7 kcal/mol/Å on the forces[182]. We also included in our analysis the N-Methyl Acetamide (NMA) molecule, with the

Table 8.1: sGDML PES details.

molecule	DFT[a]	train samples	energy MAE / kcal/mol	force MAE / kcal/mol/Å
aspirin	PBE+TS	1000	0.194	0.679
benzene	PBE−TS	1000	0.16	0.07
ethanol	PBE+TS	1000	0.072	0.335
malonaldehyde	PBE+TS	1000	0.098	0.414
naphtalene	PBE+TS	1000	0.116	0.113
paracetamol	PBE−TS	1000	0.153	0.491
salicylic acid	PBE+TS	1000	0.115	0.281
toluene	PBE+TS	1000	0.097	0.142
uracyl	PBE+TS	1000	0.107	0.241

[a] +TS and −TS indicate the presence and absence, respectively, of the Tkatchenko Scheffler correction for Van der Waals interactions [188]

DFT-B3LYP/cc-pVDZ surface by Nandi *et al.* [84] This PES was fit to about 45000 cm^{-1} of total vibrational energy and allows to simulate cis-trans interconversion. In our analysis, only the trans-version of NMA was simulated.

8.3 Tests of numerical accuracy

We use the malonaldehyde molecule to test the accuracy of our DCI calculations. The atom numbering used for malonaldehyde is displayed in Fig. 8.2

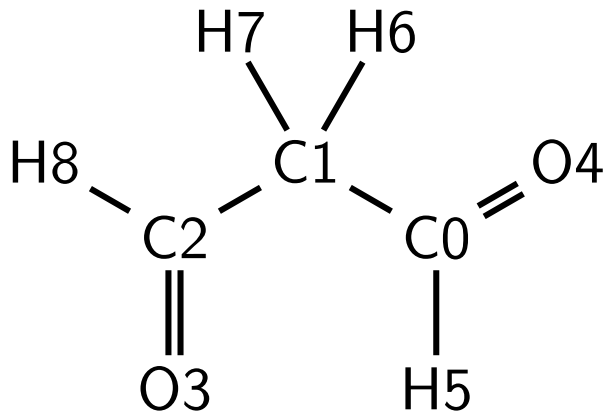


Figure 8.2: Skeletal picture of malonaldehyde with atoms numbering.

In the following paragraphs we show the accuracy of the $\frac{\partial \Delta \mathcal{H}_0}{\partial \alpha}$ approximate equation compared to a finite difference calculation using the SEF4 algorithm, and the convergence of DCI with the number of trajectories.

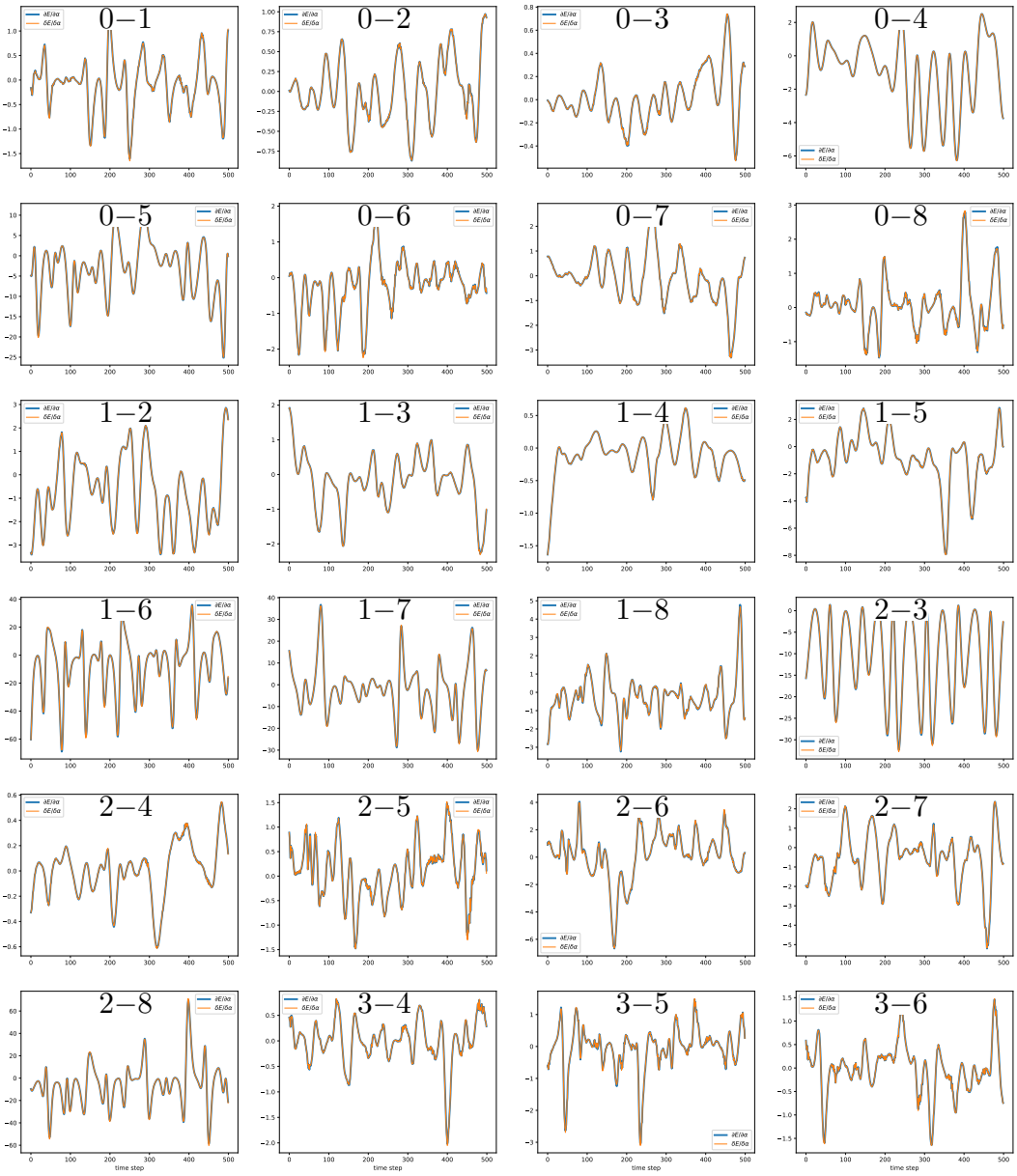


Figure 8.3: Comparison of $\frac{\partial E}{\partial \alpha}$ (blue) with $\frac{\delta E}{\delta \alpha}$ (orange) for 24 pairs of atoms of Malonaldehyde. The pair of numbers $(n - m)$ on top of each plot indicate the pair of decoupled atoms, with reference to Fig. 8.2

In Figure 8.3 we show that the truncated series of Eqs. 8.7 and 8.8 can faithfully capture the value of $\frac{\partial E}{\partial \alpha}$ by a finite difference calculation. The finite difference scheme

is obtained by computing

$$\begin{aligned}\frac{\delta E}{\delta\alpha}(\tau) &\approx \frac{\mathcal{H}_0(\tau)|_{\alpha+\delta\alpha} - \mathcal{H}_0(\tau)|_{\alpha-\delta\alpha}}{2\delta\alpha} \\ &= \frac{\mathcal{H}_0\left(e^{-\{\tilde{\mathcal{H}}(\alpha+\delta\alpha),\cdot\}}z(0)\right) - \mathcal{H}_0\left(e^{-\{\tilde{\mathcal{H}}(\alpha-\delta\alpha),\cdot\}}z(0)\right)}{2\delta\alpha},\end{aligned} \tag{8.11}$$

where $e^{-\{\tilde{\mathcal{H}}(\alpha+\delta\alpha),\cdot\}}z(0)$ is approximated by the SEF4 algorithm. We compare the results of Eq. 8.11 with Eq. 8.10 (using a sixth order expansion of $\partial\tilde{p}/\partial\alpha$ and $\partial\tilde{q}/\partial\alpha$) for some of the pairs of atoms of the Malonaldehyde molecule.

To converge the DCI integral, we run multiple independent trajectories, and compute the absolute energy loss $\left|\frac{\partial}{\partial\alpha}\Delta\mathcal{H}_0\right|$ at every time step of each trajectory. The DCI is simply the average of the collected energy losses. All the independent trajectories have the same initial geometry, equal to the equilibrium geometry. The initial momenta (in normal mode coordinates) are sampled randomly from the surface of a sphere with radius equal to $|p| = \sqrt{2mE_0}$, where E_0 is an estimate of the anharmonic ZPE. This sampling of the initial conditions ensures that each trajectory explores the very same energy shell. However, since all the simulations have initially the same geometry, the configurations nearby equilibrium are sampled more densely, making the sampling biased towards the minimum energy configuration. To estimate the (anharmonic) ZPE

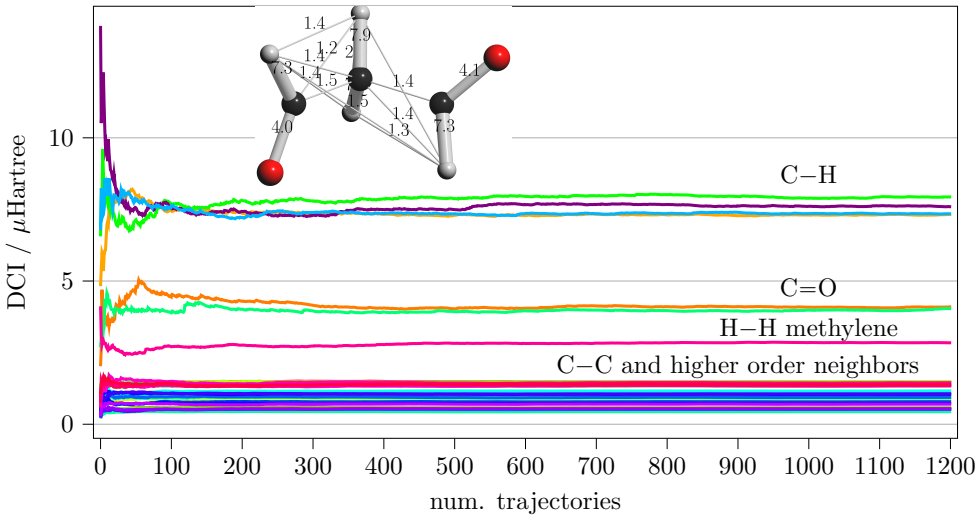


Figure 8.4: DCI convergence with the number of trajectories for malonaldehyde. The ball and stick cartoon is represented as an inset in the top of the picture. In the inset the thicker bonds represent higher values of DCI, indicated by the numbers near the bonds (in μHartree).

we use the Adiabatic Switching (AS) technique, where a swarm of trajectories is evolved under a potential that slowly switches in time, from the harmonic ZPE shell to the

anharmonic one. In the limit of infinitely long simulation (and infinitely slow switching), the adiabatic theorem ensures that the final energy of the system corresponds to the anharmonic ZPE shell.[57, 195] In Figure 8.4 we see the convergence of the DCI for all the atom-atom pairs of malonaldehyde with the number of trajectories. The first neighbors C–H pairs have both the greatest DCI value and the greatest oscillations. In

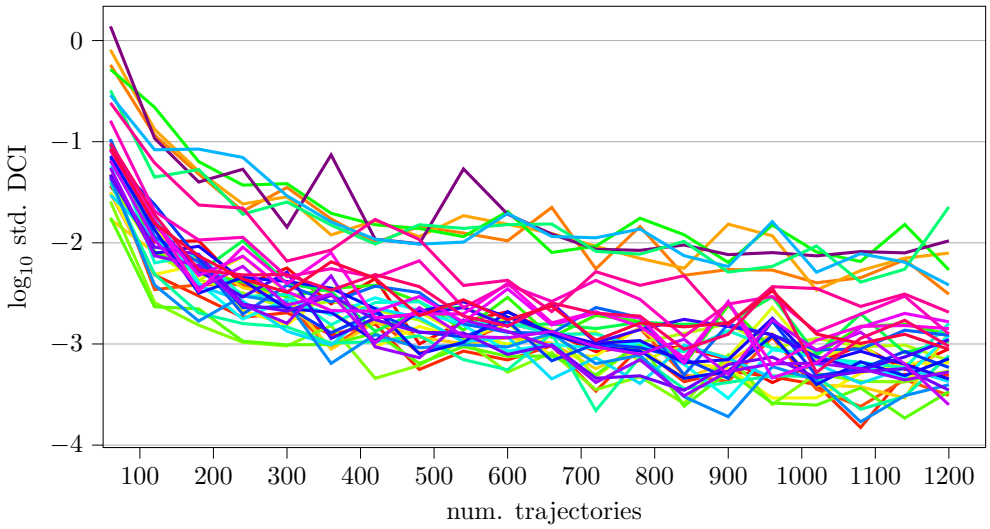


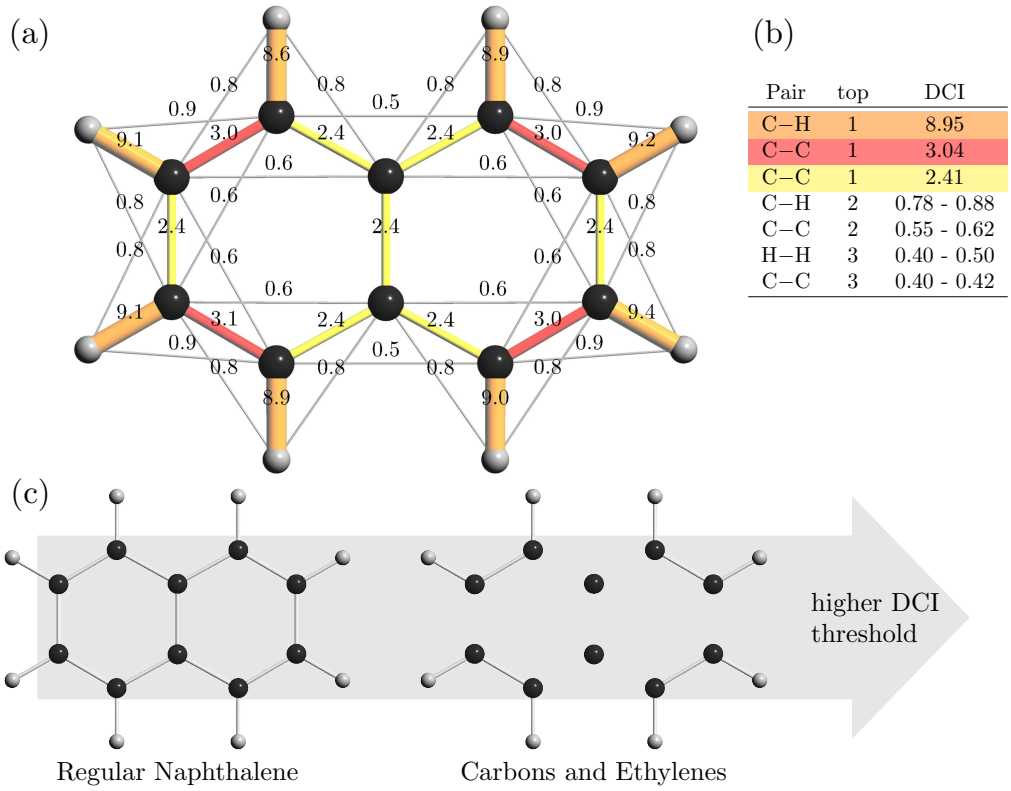
Figure 8.5: Logarithm of the DCI standard deviations for intervals of the plots in Fig. 8.4

Figure 8.5 we show the oscillations of the DCI values in Fig. 8.4, taken as the standard deviation of the DCI values for incremental intervals of 60 trajectories. The convergence analysis indicates that using about 1000 trajectories we can compute the DCI with an error on the second or third decimal place, depending on the pair of atoms. We see that pair of atoms with smaller DCI values are converged more easily, but they still carry an error in the third decimal place. To obtain the results discussed in the next section, we used about one thousand 0.7ps long trajectories for each system.

8.4 DCI Molecular Graphs

By calculating the Dynamic Coupling Index (DCI) for each pair of atoms within an organic molecule, we can gauge the significance of the dynamic couplings between each pair. A higher DCI indicates that the pair of atoms are compelled to vibrate synchronously and any perturbation of that synchronization implies a significant change in energy. Conversely, pairs of atoms with completely uncorrelated motions would exhibit a DCI value close to 0. To convey this information in a compact manner, we depict each molecule as a molecular graph, where each atom serves as a node, and the graph edges are weighted by the corresponding DCI. In visual representations,

the thickness of edges increases with larger DCI values, while the thinnest edges are assumed irrelevant and omitted. The determination of the DCI threshold value, beyond which an edge is omitted, is arbitrary. In essence, adjusting the DCI threshold allows for a spectrum of representations, ranging from a picture where every two atoms are connected by an edge, to a picture where molecules are portrayed as collections of disconnected atoms. The flexibility in choosing this threshold enables the exploration of many molecular subgraphs, encompassing a number of often unexpected structures.



Pair	top	DCI
C−H	1	8.95
C−C	1	3.04
C−C	1	2.41
C−H	2	0.78 - 0.88
C−C	2	0.55 - 0.62
H−H	3	0.40 - 0.50
C−C	3	0.40 - 0.42

Figure 8.6: Picture of naphthalene with labeled DCI values on the edges of the molecular graph and the most connected components of the graph on bottom. The table summarizes the largest (averaged) DCI values for the pairs along with the topological (top) distance in the molecular graph. All DCI values are in μHartree

As we assess the significance of dynamical correlations between atom pairs, it becomes evident that the Dynamic Coupling Index (DCI) is subject to the influence of atomic masses. Notably, lighter atoms, like hydrogen, exhibit greater displacement in response to perturbation, whereas heavier atoms experience only marginal changes in velocity. Consequently, hydrogen atoms demonstrate substantial coupling with their neighboring atoms. Additionally, the molecule's or fragment's flexibility assumes a crucial role, as conformational changes are duly considered in the ensemble average. In the case of the rigid naphthalene molecule in Figure 8.6, conformational changes are

very limited, with each atom having a non negligible interaction only with its closest neighbors. In Figure 8.6 we show only the DCI values larger than 0.5 μHartree for naphthalene. As announced above, C−H pairs have the largest DCI values, because the light H atoms gain velocity easily. In this case it is more interesting to investigate the DCI values of the carbon ring structure. In fact, given the absence of substantial conformational change, and the structural symmetry, the DCI highlights the importance of the π bonds in the C−C ring structure, where the C−C bonds highlighted in red in Figure 8.6 (conventionally called 1-2, 3-4, 5-6, and 7-8) have the largest DCI value of about 3.04 μHartree among all the C−C pairs. All the other first neighbors C−C pairs (in yellow in the Figure) have a lower DCI value, that is on average about 2.41 μHartree. The yellow bonds are not chemically equivalent, yet they have similar DCI values. This shows that the dynamical coupling index, which accounts for both the kinetics and the potential energy contributions to the perturbation, perceives the ring structure as composed of two types of C−C bonds: The more rigid (red) bonds are those that must vibrate synchronously, and the more flexible (yellow) bonds are allowed to vibrate more asynchronously. In the lower part of Figure 8.6 we show that increasing the DCI threshold between 1.0 μHartree and 2.5μHartree we obtain the traditional ball and stick picture of naphthalene. If instead we choose a DCI threshold between 2.5 and 3.0 μHartree we obtain a picture of C_2H_2 fragments and two disconnected C atoms. This last representation of naphthalene with DCI threshold between 2.5 and 3.0 μHartree is consistent to the picture of the π system given by the Electron Localization Function (ELF). [196]

8.4.1 The Methyl and Methylene Graphs

The Dynamical Coupling Index captures the importance of correlated motion, thus showing when a pair of atoms cannot move but synchronously. One interesting case occurs when a sp^3 hybridized carbon atom can rotate around one of its bond's axis without changing the electronic structure, such as in the case of rotating methyl group. In all the molecules in which there is a methyl group, (these are aspirin, ethanol, paracetamol, toluene, and N-methyl acetamide) the DCI picture of the methyl is a pyramidal shape (see the left side of Figure 8.7), where the H−H pairs are strongly correlated. This is a manifestation of the sp^3 hybridization of the carbon atom. In fact, for the methyl group to be able to rotate, the three hydrogens must move in a concerted manner, by pushing and pulling their neighbors. Any uncorrelated type of motion would result in de-hybridization of the carbon, resulting in significant changes of the electronic structure for the whole molecule. A similar argument can be made for the methylene group, which is always shown with a triangular shape, as in the right side of Figure 8.7.

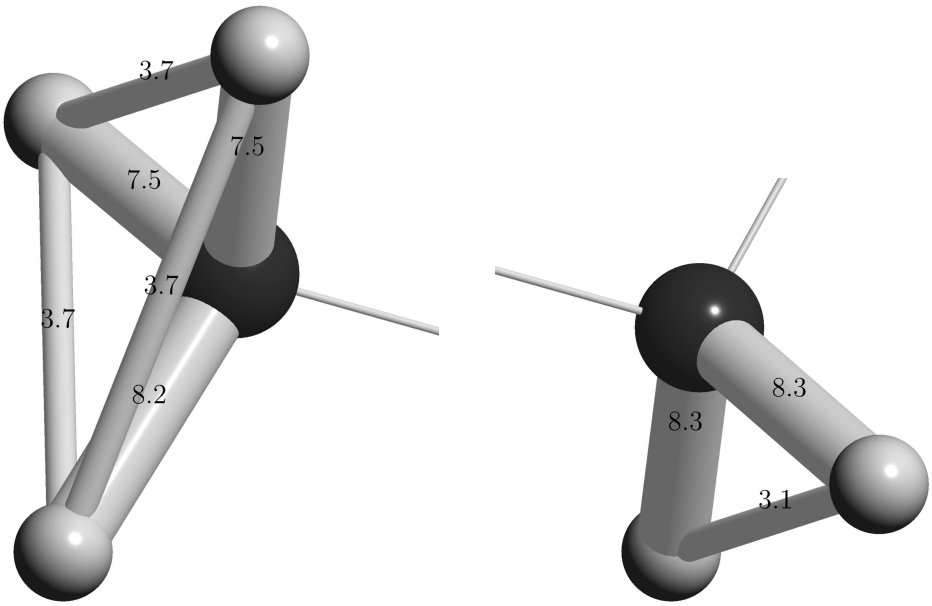


Figure 8.7: Methyl and methylene DCI graphs of the ethanol molecule. DCI values are reported in μHartree on top of the corresponding edges

8.5 DCI Molecular Blocks

In Figure 8.8 we show the DCI graphs, along with their decompositions into independently vibrating molecular blocks for the four middle sized organic molecules: Aspirin, salicylic acid, uracil, and paracetamol. The upper section of Figure 8.8 illustrates the connected molecular graphs, each encapsulated in a dark gray bubble. Progressing down the background's gray arrows, increasing the DCI threshold, we represent the decomposition of each molecular graph into smaller subgraphs, each enclosed in a colored bubble. In the bottom part of Figure 8.8, where the DCI threshold value is 2.2 μHartree or larger, the molecular graphs are decomposed into their minimal molecular blocks, which are shared across all systems. The minimal blocks that we can recognize in Figure 8.8 are the benzene ring (bright orange), the methyl groups (blue), the carbonyl (magenta), hydroxyl (bright green), secondary amine N−H (yellow), and ethylene (mold green). The oxygen atom of the ester group in aspirin (pink bubble) is an exception, as it remains pretty much uncorrelated from the other atoms even at relatively low values of the DCI threshold. Each of the four molecules in Figure 8.8 has its own individual fragmentation pattern. For instance, most of the fragments of aspirin are connected with each other with a DCI value that is lower than 1.6 μHartree. Instead, to decompose the Salicylic acid into its minimal fragments, one needs a DCI threshold larger than 2.1 μHartree. Notice that, compared to the traditional functional group picture of organic molecules, the picture offered by calculation of the DCI does not distinguish between the C=O of an amide or a carboxyl group, or between the

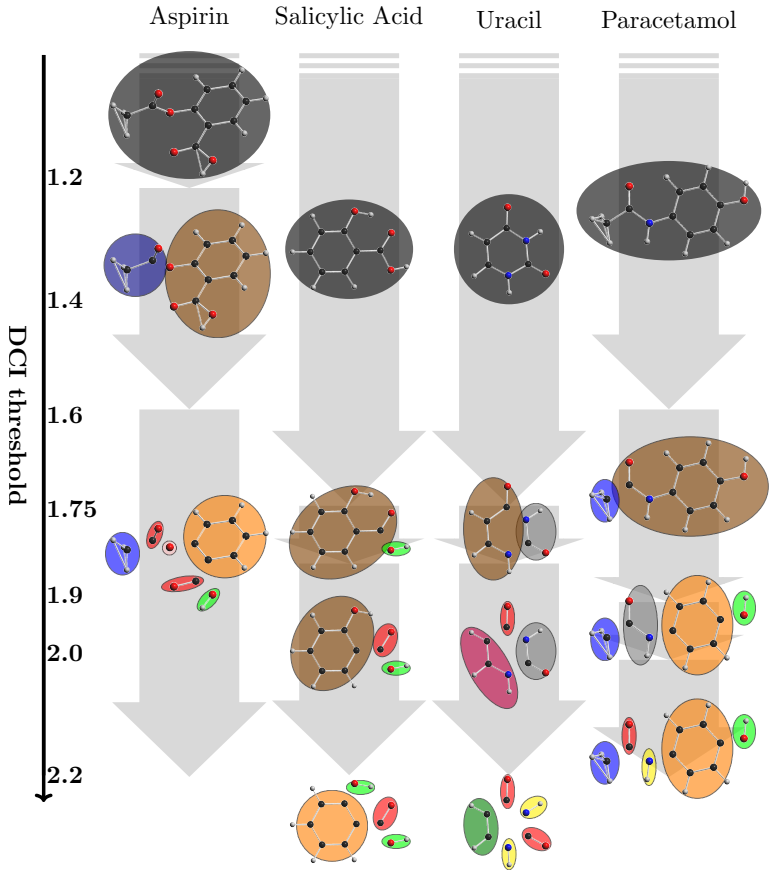


Figure 8.8: Decompositions into fragments of four molecules in the DCI interval from 1.0 to 2.2 μHartree. Paracetamol, aspirin, salicylic acid, uracil and their fragments are represented each on a node of a vertical arrow diagram. The threshold DCI value at which the significant fragmentations occur is displayed on the left axis.

O−H of an alcohol or a carboxylic acid. The only difference between these groups is their relative DCI value, which however, varies significantly from molecule to molecule, because it strongly depends on the chemical neighborhood. For instance, the DCI threshold necessary to separate the O−H alcohol block is 1.9 μHartree for paracetamol and 2.2 for the salicylic acid. In both cases the alcohol is bonded to a aromatic ring, but in the former case O−H is only slightly interacting with its neighborhood, while in the latter it is involved in a hydrogen bond. This sensibility towards the chemical neighborhood makes the DCI a good candidate for a molecular descriptor. [129]

In Figure 8.8 the aromatic benzene ring consistently exhibits C−C DCI pairs exceeding 2.3 μHartree in all instances. In contrast, the uracil molecule, characterized by a weaker aromaticity, demonstrates DCI values ranging from 1.6 to 2.2 μHartree for the C−N pairs, with C=C pairs having a larger DCI value of 3.4 (refer to Figure 8.9). Notably, this suggests a more pronounced correlation between atoms in double bonds compared to those within aromatic rings, as evidenced by the C−C pairs in aromatic

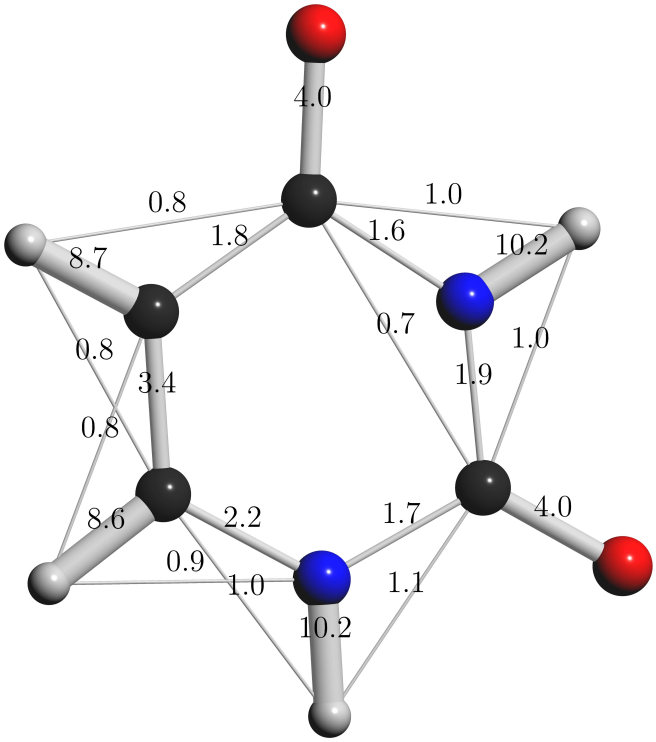


Figure 8.9: DCI graph of uracil. DCI values are reported in μHartree on top of the corresponding edges

molecules ranging from 2.4 to 2.8 in different molecules. This evidence is consistent with the current chemical intuition, according to which double bonds are stiffer towards stretching motion than conjugated and aromatic bonds. The additional information provided by the DCI is a quantification of this stiffness, that is, the C=C double bond of uracil is at least 20% stiffer than the aromatic C=C bonds of the other investigated molecules.

Chapter 9

Summary and Conclusions

This PhD dissertation describes new numerical algorithms for classical and semiclassical molecular dynamics simulations, along with vibrational spectroscopy applications on model systems and realistic molecules. The common ground for this research is the identification of vibrationally independent groups of degrees of freedom, which can then be used for methodologies that rely on that approximation, such as the Divide-and-Conquer method for semiclassical vibrational spectroscopy.

In chapter 3 we introduce heuristic algorithms designed to autonomously identify optimal subspaces for the Divide-and-Conquer SCIVR method. We demonstrate the efficacy of the Probability Graph Evolutionary Algorithm (PG-EA) on both model systems and the N-Methyl Acetamide (NMA) molecule, yielding vibrational spectra that closely align with experimental measurements (MAE $\approx$ 30 cm^{-1}). Chapter 4 further extends the applicability of the DC-SCIVR method to larger systems by exploring various approaches to approximate the Hessian matrix along trajectories for Semiclassical Dynamics. We propose to use the unsupervised machine learning algorithm "Neural Gas" to optimally clusterize the trajectory in a convenient representation of the position space. Our NGas implementation demonstrates systematic improvement over the Dynamical dataBase of Hessian (DBH) method. Utilizing the NGas Hessian approximation technique and the PG-EA algorithm, we compute the DC-SCIVR spectrum for four NH_2 stretching modes of a small synthetic peptide comprising 46 atoms. The resulting spectroscopic frequencies approach experimental values, systematically improving classical signals. Remarkably, our approach achieves a MAE close to 30 $cm^{-1}$with respect to experimental values, even when employing approximate Hessians.

In chapter 6 we introduce the new "pair-decoupling" concept, which gives an entirely new perspective of the dynamical correlation between pairs of nuclear degrees of freedom. When two pieces of a molecular system are pair-decoupled, they do not perceive each other's displacements: Each piece vibrates as if the other piece remains at

its initial (equilibrium) configuration. The pair-decoupling idea can be applied to any coordinate representation of the nuclei, as long as the matrix elements of the potential second derivative are nonzero. Therefore, we adopt a Cartesian representation of the coordinate system, and we decouple those coordinates pertaining to any two pair of atoms. In this way we can investigate the importance of atom-atom couplings, which is appealing to chemical intuition. To enforce the pair-decoupling idea in a MD simulation we derived the SEF algorithm in chapter 6, by building upon the Symplectic integration techniques described in chapter 5. SEF enables the simulation of systems with atoms exhibiting dynamically uncorrelated motion, while preserving the geometric properties of Hamiltonian integration. In the context of simulating vibrations with the SEF algorithm, the decoupled pair of atoms consistently encounters the force contribution as if the decoupled atom were held at equilibrium. Additionally, SEF allows for the application of partial decoupling, wherein the pair perceives only a fraction of the actual force. The SEF algorithm is constructed to preserve the classical symplectic structure in time, and uses a local harmonic approximation of the potential energy surface to propagate the pair-decoupled force. We used SEF to determine the crucial importance of the hydrogen bond orientation in the Salicylic Acid molecule. If the oxygen and hydrogen atoms involved do not correlate, then the chemistry of Salicylic Acid would change dramatically, and the Carboxylic group would start twisting out of plane.

The SEF algorithm can be employed to simulate the vibrational spectroscopy from correlation functions with classical or semiclassical dynamics methods, allowing to simulate the effects of a specific dynamical correlation on the whole molecular system. In chapter 7 we show that a modified version of the SEF algorithm can be used to split exactly the total potential energy into contributions coming from each degree of freedom, accounting for all the couplings between said degree of freedom and the rest of the molecule. The potential energy contribution of each degree of freedom can be used as the projected potential for the Divide-and-Conquer approach to semiclassical spectroscopy.[49] We show that this approach makes the DC-SCIVR method exact for any (arbitrary) partition of the system's degrees of freedom.

In chapter 8 we show how to measure the effect of a pair-decoupling perturbation for a short time, in the limit of no decoupling using response function theory. We call "Dynamical Coupling Index" (DCI) the averaged energy change caused by a decoupling perturbation. We use it to quantify the importance of dynamic atom-atom couplings of many middle-sized organic molecules. The DCI exhibits greater values for lighter atoms, which are more susceptible to the effects of artificial decouplings compared to heavier atoms. Furthermore, DCI accurately captures motions that necessitate strictly concerted movements, such as the rotation of methyl groups. In the case of a rotating methyl group, the dynamical couplings between the methyl hydrogens is forced by the

sp^3 hybridization of the carbon atom, which is perfectly portrayed by a pyramidal shape of the methyl group in the picture of the DCI molecular graph. In the case of rigid molecules, such as naphthalene, the DCI analysis can recover the ball and stick picture of the said molecule. The example of naphthalene in section 8.4 also shows that the picture provided by the DCI analysis is consistent with the picture given by the Electron Localization Function (ELF) of the π system.

Finally, serving as a quantitative measure, DCI enables the differentiation between strongly correlated and weakly correlated groups of atoms. By applying a threshold to the DCI values, it becomes possible to identify those molecular blocks that are compelled to vibrate synchronously. The analysis in Section 8.5 reveals that strongly correlated groups of atoms often involve a decomposition of the traditional functional groups into minimal molecular blocks that contain small, rigidly vibrating fragments. Examining the DCI value for the C=C double bond of uracil and the C=C bonds of aromatic rings, we can quantify that the double bond of uracil is between 20% and 40% stiffer, depending on the bond and on the system. This measure of stiffness accounts for the dynamics of the system, which is allowed to explore extensively the ground state potential energy surface.

Appendices

Appendix A

Semiclassical Approximation

A.1 Propagator in Path Integral Formulation

The state of a quantum system at time t given an initial state and the time-independent Hamiltonian $\mathcal{H}$ is given by

$$|\Psi(t)\rangle = e^{-i\mathcal{H}t}\,|\Psi(0)\rangle \tag{A.1}$$

$$|\Psi(t)\rangle = \int_{-\infty}^{\infty} e^{-i\mathcal{H}t}\,|q_0\rangle\,\langle q_0|\Psi(0)\rangle\,dq_0, \tag{A.2}$$

where $|q_0\rangle$ is an eigenvector of the initial position operator. The wavefunction in position representation is given by a projection of $|\Psi(t)\rangle$ on the position basis $|q\rangle$, that is

$$\langle q|\Psi(t)\rangle = \int_{-\infty}^{\infty} \left\langle q|e^{-i\mathcal{H}t}|q_0\right\rangle \langle q_0|\Psi(0)\rangle\,dq_0. \tag{A.3}$$

The kernel function $K(q,q_0,t) = \left\langle q|e^{-i\mathcal{H}t}|q_0\right\rangle$ is the propagator in position basis from an initial configuration q_0 to the final configuration q. The propagator inside $K(q,q_0,t)$ can be expanded as a product of infinitesimal time propagators, and the identities $1 = \int |q_i\rangle\,\langle q_i|\,\mathrm{d}q_i$ can be inserted between the products.

$$K(q,q_0,t) = \lim_{N\to\infty} \langle q|\left(e^{-i\mathcal{H}t/N}\right)^N |q_0\rangle \tag{A.4}$$

$$= \lim_{N\to\infty} \int \mathrm{d}q_1 \ldots \mathrm{d}q_{N-1}\,\langle q|\left(e^{-i\mathcal{H}t/N}\right)|q_{N-1}\rangle\,\langle q_{N-1}|\left(e^{-i\mathcal{H}t/N}\right)|q_{N-2}\rangle \ldots \langle q_1|\left(e^{-i\mathcal{H}t/N}\right)|q_0\rangle \tag{A.5}$$

$$= \lim_{N\to\infty} \int \mathrm{d}q_1 \ldots \mathrm{d}q_{N-1} \prod_{j=0}^{N-1} \left\langle q_{j+1}\left|e^{-i\mathcal{H}t/N}\right|q_j\right\rangle. \tag{A.6}$$

The Trotter-Suzuki formula $e^{-i\mathcal{H}t} = \lim_{N\to\infty}\left(e^{-i\mathcal{V}t/N}e^{-i\mathcal{K}t/N}\right)^N$ allows us to approximate the propagator as an infinite product of simpler propagators evolved for an infinitesimal time. Substituting this expression into in Eq. A.6 gives

$$K(q,q_0,t) = \lim_{N\to\infty}\int \mathrm{d}q_0\ldots\mathrm{d}q_{N-1}\prod_{j=0}^{N-1}\left\langle q_{j+1}\left|e^{-i\mathcal{H}t/N}\right|q_j\right\rangle \tag{A.7}$$

$$= \lim_{N\to\infty}\int \mathrm{d}q_0\ldots\mathrm{d}q_{N-1}\prod_{j=0}^{N-1}\left\langle q_{j+1}\left|e^{-i\mathcal{K}t/N}e^{-i\mathcal{V}t/N}\right|q_j\right\rangle \tag{A.8}$$

$$= \lim_{N\to\infty}\int \mathrm{d}q_0\ldots\mathrm{d}q_{N-1}\prod_{j=0}^{N-1}\left\langle q_{j+1}\left|e^{-i\mathcal{K}\Delta t}\right|q_j\right\rangle e^{-iV(q_j)\Delta t} \tag{A.9}$$

$$= \lim_{N\to\infty}\left(\frac{m}{2\pi i\Delta t}\right)^{\frac{N}{2}}\int \mathrm{d}q_0\ldots\mathrm{d}q_{N-1}\prod_{j=0}^{N-1}e^{i\frac{m}{2}\left(\frac{q_j-q_{j+1}}{\Delta t}\right)^2\Delta t}e^{-iV(q_j)\Delta t} \tag{A.10}$$

$$= \lim_{N\to\infty}\left(\frac{m}{2\pi i\Delta t}\right)^{\frac{N}{2}}\int \mathrm{d}q_0\ldots\mathrm{d}q_{N-1}e^{i\Delta t\sum_{j=0}^{N-1}\left[\frac{m}{2}\left(\frac{q_j-q_{j+1}}{\Delta t}\right)^2-V(q_j)\right]}, \tag{A.11}$$

where we introduced the new variables $\Delta t = t/N$, and, in Eq. A.10, we applied the squared momentum operator to the position vectors $|q_j\rangle$ and $\langle q_{j+1}|$. In the limit $N\to\infty$, the quantity in the exponent of Eq. A.11 corresponds to the action of the path that goes from q_0 to q. Thus we can write the final expression for the path integral representation of the Kernel as

$$K(q,q_0,t) = \left\langle q\left|e^{-i\mathcal{H}t}\right|q_0\right\rangle \tag{A.12}$$

$$= \lim_{N\to\infty}\left(\frac{m}{2\pi i\Delta t}\right)^{\frac{N}{2}}\int \mathrm{d}q_0\ldots\mathrm{d}q_{N-1}e^{i\int_0^t\left[\frac{m}{2}\dot{q}^2(t')-V(q(t'))\right]dt'}, \tag{A.13}$$

$$= \int e^{iS[q(t)]}\mathcal{D}q(t). \tag{A.14}$$

For a graphical representation of Eq. A.14 one can imagine all conceivable pathways connecting q_0 and q. Each path is assigned a numerical value, that is the action functional $S[q(t)]$, which is then multiplied by i and exponentiated. The sum of all these exponentials gives the transition probability amplitude $K(q,q_0,t)$. However, the integrand in Eq. A.14 is an oscillating function of the paths, which leads to most contributions to cancel out upon integration. Because of these oscillating properties, numerical integration of Eq. A.14 is an open challenge, often referred as the path integral version of the numerical sign problem.

A.2 Van-Vleck Propagator

Although in Eq. A.14 all conceivable trajectories are rigorously accounted for, clearly some of them bring a larger contribution to the integral and do not cancel out. Such trajectories are those which have small values for the action functional, corresponding to slow oscillations of the integrand near the origin. In particular, the trajectories that minimize the action are the classical trajectories. Thus a convenient strategy to obtain an approximate propagator is to neglect the non-stationary paths, expand the action to second order of $q(t)$, and evaluate the integral only over the stationary paths. Such procedure corresponds to the *stationary phase* approximation, which, in turn, corresponds to the *steepest descent* integration method of the Wick rotated action. The stationary phase approximation of Eq. A.14 gives

$$K(q,q_0,t) \approx \left(\frac{m}{2\pi i\Delta t}\right)^{\frac{N}{2}} \sum_{y|S'(y)=0} \int e^{iS(y)} e^{\frac{i}{2}S''(y)(q-y)^2} \mathrm{d}q \tag{A.15}$$

$$= \sum_{y|S'(y)=0} \left(\frac{m}{2\pi i\Delta t}\right)^{\frac{N}{2}} \left(\frac{1}{2\pi i}\right)^{\frac{1}{2}} \left|\frac{\partial^2 S(y)}{\partial q_i \partial q_j}\right|^{-\frac{1}{2}} e^{iS(y)}. \tag{A.16}$$

Eq. A.16 is an expression of the propagator in a semiclassical approximation. The only trajectories $y(t)$ accounted for are those that make the action stationary, that is $S'[y(t)] = 0$. It is convenient to express the action derivatives so that it depends only on the initial and final configurations of the classical paths. As a matter of fact the initial and final positions q_0 and $q_N = q(t) = q$ are the only variables of Eq. A.16, while the symbols q_i for $i = 1, 2, \ldots, N-1$ are just parameters that depend on q_0 and q_N. Furthermore, because the action is evaluated on the classical paths, the derivatives $\partial S_N/\partial q_i = 0$ when $0 < i < N$. Which means that most of the entries in the second derivative matrix are in fact zero.

We aim to transform the partial derivatives of the action to a more amenable form $\frac{\partial^2 S}{\partial q_i \partial q_j} \to \frac{\partial^2 S}{\partial q_0 \partial q_N}$. The derivative of the action w.r.t. the parameter q_i gives

$$\frac{\partial S_N}{\partial q_i} = \frac{Nm}{t}\left(2q_i - q_{i+1} - q_{i-1}\right) - \frac{t}{N}\frac{\partial V(q_i)}{\partial q_i}. \tag{A.17}$$

Deriving this last expression w.r.t. the variable q_N and setting $\tau = t/N$ gives

$$\begin{aligned}
\frac{\partial^2 S_N}{\partial q_N \partial q_i} &= \frac{m}{\tau}\left(2\frac{\partial q_i}{\partial q_N} - \frac{\partial q_{i+1}}{\partial q_N} - \frac{\partial q_{i-1}}{\partial q_N}\right) - \tau \frac{\partial^2 V(q_i)}{\partial q_i^2}\frac{\partial q_i}{\partial q_N} \\
&= \frac{m}{\tau}\left[\left(2 - \frac{\tau^2}{m}\frac{\partial^2 V(q_i)}{\partial q_i^2}\right)\frac{\partial q_i}{\partial q_N} - \frac{\partial q_{i+1}}{\partial q_N} - \frac{\partial q_{i-1}}{\partial q_N}\right] \\
&= \sum_j \frac{m}{\tau} K_{ij}\frac{\partial q_j}{\partial q_N} - \frac{m}{\tau}\delta_{i,N-1},
\end{aligned} \tag{A.18}$$

where the K matrix is tridiagonal, with entries

$$\begin{cases}
K_{i,i} = 1 - \frac{\tau^2}{m}V''(q_1) \\
K_{i,i+1} = K_{i+1,i} = -1 \\
K_{i,i+n} = 0 \quad \forall n > 1.
\end{cases} \tag{A.19}$$

Computing the inverse of the K matrix and comparing it to the action second derivative $\partial^2 S/\partial q_i \partial q_j$ one can obtain the expression of the Van Vleck propagator in terms of derivatives of the initial and final positions only, that is

$$K(q_N, q_0, t) \approx \sum_{\substack{q(0)=q_0 \\ q(t)=q_N}} \sqrt{-\frac{1}{2\pi i}\left|\frac{\partial^2 S}{\partial q_N \partial q_0}\right|} e^{iS(q)}. \tag{A.20}$$

The derivative of the action w.r.t. the initial and final coordinates can be demonstrated to be

$$\begin{cases}
\dfrac{\partial S}{\partial q_0} = -p_0 \\
\dfrac{\partial S}{\partial q_N} = -p_N.
\end{cases} \tag{A.21}$$

Therefore we can transform the second derivative of the action to

$$\frac{\partial^2 S}{\partial q_N \partial q_0} = \frac{\partial}{\partial q_N}\frac{\partial S}{\partial q_0} = -\frac{\partial}{\partial q_N}p_0 = -\left(\frac{\partial q_N}{\partial p_0}\right)^{-1}, \tag{A.22}$$

which, upon substitution into Eq. A.20 gives the expression of the Van Vleck propagator depending only on the determinant of the $\partial q(t)/\partial p_0$ Jacobian

$$K(q_N, q_0, t) \approx \sum_{\substack{q(0)=q_0 \\ q(t)=q_N}} \sqrt{\frac{1}{2\pi i}\left|\frac{\partial q(t)}{\partial p_0}\right|^{-1}} e^{iS(q(t))}. \tag{A.23}$$

The state vector's survival amplitude probability then becomes

$$\begin{aligned}\left\langle \Psi(0) \left| e^{-i\mathcal{H}t} \right| \Psi(0) \right\rangle &= \sum_{\substack{q(0)=q_0 \\ q(t)=q_N}} \iint \mathrm{d}q_N \mathrm{d}q_0 \times \\ &\times \langle \Psi(t)|q_N\rangle \sqrt{\frac{1}{2\pi i}\left|\frac{\partial q(t)}{\partial p_0}\right|^{-1}} e^{iS(t)} \langle q_0|\Psi(0)\rangle . \end{aligned} \quad \text{(A.24)}$$

Equation A.24 is numerically unconvenient, simply because it requires a sampling over all the initial $q_0 = q(0)$ and final $q_t = q(t)$ positions, making it a double boundary condition problem. By a convenient change of variables Eq. A.24 can be recast to a sampling over the initial positions and momentum with a change in the integration variables known as the Initial Value Representation (IVR) trick [104].

$$\sum_{\substack{q(0)=q_0 \\ q(t)=q_N}} \int dq_N = \int \mathrm{d}p_0 \left|\frac{\partial q(t)}{\partial p_0}\right| . \quad \text{(A.25)}$$

Inserting Eq. A.25 into Eq. A.24 gives the Van Vleck IVR expression for the survival probability amplitude

$$\left\langle \Psi(0) \left| e^{-i\mathcal{H}t} \right| \Psi(0) \right\rangle = \iint \mathrm{d}p_0 \mathrm{d}q_0 \langle \Psi(t)|q_t\rangle \sqrt{\frac{1}{2\pi i}\left|\frac{\partial q(t)}{\partial p_0}\right|} e^{iS(t)} \langle q_0|\Psi(0)\rangle . \quad \text{(A.26)}$$

A.3 Herman-Kluk Propagator

In Eq. A.26 the state function $\Psi(t)$ is projected to position representation. If one expresses the wavefunction in a basis of coherent states the Van-Vleck propagator may be recast to the Heller-Herman-Kluk-Kay propagator. [35, 62, 104]

$$\left\langle \Psi(0)|e^{-i\mathcal{H}t}|\Psi(0)\right\rangle \approx \frac{1}{2\pi}\iint \mathrm{d}p_0 \mathrm{d}q_0 C_{HK}(t) e^{iS(t)} \langle \Psi(t)|p_t, q_t\rangle \langle p_0, q_0|\Psi(0)\rangle , \quad \text{(A.27)}$$

where the Harman-Kluk prefactor $C_{HK}(t)$ is

$$C_{HK}(t) = \sqrt{\left(\frac{1}{2}\right)^F \det\left(\frac{\partial p(t)}{\partial p_0} + \frac{\partial q(t)}{\partial q_0} - i\gamma\frac{\partial q(t)}{\partial p_0} + \frac{i}{\gamma}\frac{\partial p(t)}{\partial q_0}\right)}. \quad \text{(A.28)}$$

The prefactor $C_{HK}(t)$ is a function of the classical system dependence from its initial conditions (p_0, q_0). The partial derivatives in A.28 depend on time and follow the

equations of motion

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial p(t)}{\partial p_0} = \frac{\partial}{\partial p_0}\left(-\frac{\partial V}{\partial q}\right) = -\frac{\partial^2 V}{\partial q^2}\frac{\partial q}{\partial p_0} \tag{A.29}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial q(t)}{\partial p_0} = \frac{\partial}{\partial p_0}(\dot{q}) = \frac{1}{m}\frac{\partial p}{\partial p_0} \tag{A.30}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial p(t)}{\partial q_0} = \frac{\partial}{\partial q_0}\left(-\frac{\partial V}{\partial q}\right) = -\frac{\partial^2 V}{\partial q^2}\frac{\partial q}{\partial q_0} \tag{A.31}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial q(t)}{\partial q_0} = \frac{\partial}{\partial q_0}(\dot{q}) = \frac{1}{m}\frac{\partial p}{\partial q_0}, \tag{A.32}$$

which can be propagated by a symplectic map.

A.4 Time-Averaging

Finally, a time-averaging technique [36, 58] can be employed to mitigate the fast oscillatory behavior of the integrand in Eq. A.27. The basic idea is to compute the phase space integral of a time-averaged quantity A instead of the simple phase space integral of said quantity, as

$$I = \iint dp_0 dq_0 A(p_0, q_0) \tag{A.33}$$

$$= \iint dp_0 dq_0 \frac{1}{T}\int_0^T dt A(p_t, q_t). \tag{A.34}$$

The two expressions are equivalent because and can be transformed by switching the order of integration and applying a change of variables $(q_0, p_0) \to (q_t, p_t)$, for which the Jacobian determinant is equal to 1.

The time-averaged spectrum of the Herman-Kluk propagated survival probability becomes

$$I(E) = \frac{1}{(2\pi)^F}\iint dp_0 dq_0 \frac{1}{\pi T}\int_0^T dt_1 \mathfrak{Re}\left[\int_0^\infty dt e^{iEte^{iS_{t_1+t}}} \langle\Psi|p_{t+t_1}, q_{t+t_1}\rangle \langle p_{t_1}, q_{t_1}|\Psi\rangle C_{t_1+t,t}\right], \tag{A.35}$$

where the classical action is evaluated from t_1 to $t_1 + t$, that is $S_{t_1+t}(p_{t_1}, q_{t_1}) = S_{t_1+t}(p_0, q_0) - S_{t_1}(p_0, q_0)$. The two time prefactor $C_{t_1+t,t}$ contains the determinants of Jacobian matrices of the type $\partial a_{t+t1}/\partial b_t$, with a and b being position or momentum. The separable approximation described by Kaledin and Miller [36] is particularly advantageous. It consists into writing the two-time prefactor as a product of the two traditional (one-time) prefactors, that is

$$C_{t_1+t,t} \approx C_{t_1+t,0}C_{t_1,0} \tag{A.36}$$

$$= e^{i\phi_{t_1+t}}e^{-i\phi_{t_1}}. \tag{A.37}$$

where the variables ϕ_t are assumed to be the phases of the Herman-Kluk prefactor. This approximation is exact for harmonic potentials and assuming coherent states as product of one dimensional functions. Applying the separable approximation to Eq. A.35 and evaluating the innermost integral until the total simulation time T, one gets the Time-Averaged expression for the power spectrum

$$I(E) = \frac{1}{(2\pi)^F} \iint dp_0 dq_0 \frac{1}{2\pi T} \left| \int_0^T dt e^{i(S_t + \phi_t + Et)} \langle \Psi | p_t, q_t \rangle \right|^2 . \tag{A.38}$$

This last expression is approximate, as the prefactor has been substituted by its phase. However, this is advantageous from a numerical point of view, because the numerical integration of the prefactor is known to be numerically challenging, as the elements of the monodromy matrix increase fast for chaotic systems. Thus retaining only the phase ensures that the contribution of the prefactor to Eq. A.38 is limited to $\phi_t \in [-\pi, \pi]$. Furthermore, Eq. A.38 is now a phase space integral of a positive definite quantity, that can be easily approached with Monte Carlo techniques, where the initial conditions are sampled from an arbitrary distribution, such as Husimi's.

Appendix B

Sketch Proof of Baker-Campbell-Hausdorff Formula

When working with exponentials of noncommuting operators (or matrices) the simple rules of product of exponentials that applies for scalars do not hold anymore. Mathematicians have defined rules to work with such objects. These rules fall under the hood of the Lie algebra, of which we want to use only one result, that is the Baker-Campbell-Hausdorff (BCH) or Baker-Campbell-Hausdorff-Dynkin (BCHD) formula. The BCHD formula

$$\begin{aligned} e^X e^Y &= e^Z \\ Z &= X + Y + \frac{1}{2}[X,Y] + \\ &\quad + \frac{1}{12}([X,[X,Y]] + [Y,[Y,X]]) + \\ &\quad - \frac{1}{24}[Y,[X,[X,Y]]] + \\ &\quad + \dots \end{aligned} \tag{B.1}$$

applies to any pair of (noncommuting) operators or matrices X and Y, where our simplified notation implies that the operator e^X is the exponential map of the X operator and $[\cdot,\cdot]$ is the commutator. Here we report only a sketch of the demonstration, for the complete and reader-friendly version of this proof see Hall's book.[197] For our simplified derivation of the BCHD formula, we need two results we are not going to demonstrate, but comment briefly. These are (i) the exponential map derivative, and (ii) a relationship between the exponential maps of commutators.

(i) The derivative of the exponential map is

$$\frac{\mathrm{d}}{\mathrm{d}t} e^{X(t)} = e^{X(t)} \frac{1 - e^{-[X,\cdot]}}{[X,\cdot]} \frac{\mathrm{d}}{\mathrm{d}t} X(t). \tag{B.2}$$

Clearly Eq. B.2 differs from the usual derivative of an exponential function because of the second factor on the r.h.s. We rewrite this term by expanding the exponential and simplifying the power series with the denominator,

$$\frac{1-e^{-[X,\cdot]}}{[X,\cdot]} = \frac{1-\sum_{n=0}^{\infty}\frac{(-1)^n}{n!}[X,\cdot]^n}{[X,\cdot]} \tag{B.3}$$

$$= \sum_{n=0}^{\infty}\frac{(-1)^n}{(n+1)!}[X,\cdot]^n. \tag{B.4}$$

It is natural to see the equivalence between Eqs. B.3 and B.4 when X is a matrix, but the relationship is also valid when X is an operator. Putting Eq. B.4 into Eq. B.2 we see that the exponential map derivative simplifies to the usual derivative of an exponential function when X is anything that commutes with its time derivative. In fact, if $\left[x(t), \frac{\mathrm{d}x(t)}{\mathrm{d}t}\right] = 0$, then

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}e^{x(t)} &= e^{x(t)}\sum_{n=0}^{\infty}\frac{(-1)^n}{(n+1)!}[x,\cdot]^n\frac{\mathrm{d}}{\mathrm{d}t}x(t)\\ &= e^{x(t)}\left[1+\sum_{n=1}^{\infty}\frac{(-1)^n}{(n+1)!}[x,\cdot]^n\right]\frac{\mathrm{d}}{\mathrm{d}t}x(t)\\ &= e^{x(t)}\frac{\mathrm{d}}{\mathrm{d}t}x(t).\end{aligned}$$

ii) Given the exponential map $e^Z = e^X e^Y$, the following relation holds for any X, Y and Z that belong to the same Lie group

$$e^{[Z,\cdot]} = e^{[X,\cdot]}e^{[Y,\cdot]}. \tag{B.5}$$

This identity is readily proven when X and Y commute. It holds true also when the two symbols do not commute. The ingredient we need to this proof is the identity $e^{[X,\cdot]}A = e^X A e^{-X}$, that we can verify straightforwardly by Taylor expansions of both sides

$$\begin{aligned}e^{[X,\cdot]}A &= e^X A e^{-X}\\ \left(1+[X,\cdot]+\frac{1}{2!}[X,[X,\cdot]]+\dots\right)A &= \left(1+X+\frac{1}{2!}X^2+\dots\right)A\left(1-X+\frac{1}{2!}X^2+\dots\right)\\ A+[X,A]+\frac{1}{2!}[X,[X,A]]+\dots &= A+XA-AX+\frac{1}{2}AXX+\frac{1}{2}XXA-XAX+\dots,\end{aligned}$$

which are identical expressions. Thus

$$\begin{aligned}e^{[Z,\cdot]}A &= e^Z A e^{-Z}A\\ &= e^X e^Y A\left(e^X e^Y\right)^{-1}\\ &= e^X e^Y A e^{-Y}e^{-X}\\ &= e^{[X,\cdot]}e^{[Y,\cdot]},\end{aligned}$$

where in the second step one must recognize that the inverse operator must have the inverted operators in opposite order, so that $(e^X e^Y)(e^X e^Y)^{-1} = 1 = e^X e^Y e^{-Y}e^{-X}$.

To prove BCHD formula we use the following recipe, described in the book of Hall:[197] We first define $Z(t) = \log\left(e^X e^{tY}\right)$; we take the derivative using Eq. B.2; we integrate between $t = 0$ and $t = 1$; we expand the integral expression to recover the explicit form of BCHD formula.

The operator Y can be defined as

$$\begin{aligned} Y &= \left(e^X e^{tY}\right)^{-1} \left(e^X e^{tY}\right) Y. \\ &= \left(e^X e^{tY}\right)^{-1} \frac{\mathrm{d}}{\mathrm{d}t} \left(e^X e^{tY}\right) \\ &= e^{-Z(t)} \frac{\mathrm{d}}{\mathrm{d}t} e^{Z(t)}. \end{aligned} \tag{B.6}$$

Using the derivative formula of Eq. B.2 on Eq. B.6 we have

$$Y = \left(\frac{1 - e^{-[Z(t),\cdot]}}{[Z(t),\cdot]}\right) \frac{\mathrm{d}Z(t)}{\mathrm{d}t}. \tag{B.7}$$

If the first factor on the right-hand-side of Eq. B.7 is invertible, which is the case when Z is small enough for a first order approximation ($e^{-[Z(t),\cdot]} \approx 1 - [Z(t),\cdot]$), then we can express the derivative of $Z(t)$ as

$$\frac{\mathrm{d}Z(t)}{\mathrm{d}t} = \left(\frac{1 - e^{-[Z(t),\cdot]}}{[Z(t),\cdot]}\right)^{-1} Y. \tag{B.8}$$

Now, using Eq. B.5 on both numerator and denominator of Eq. B.8 we get

$$\frac{\mathrm{d}Z(t)}{\mathrm{d}t} = \left(\frac{1 - \left(e^{[X,\cdot]} e^{t[Y,\cdot]}\right)^{-1}}{\log\left(e^{[X,\cdot]} e^{t[Y,\cdot]}\right)}\right)^{-1} Y. \tag{B.9}$$

Now we can integrate the differential equation between $t = 0$ and $t = 1$

$$Z(1) = Z(0) + \int_0^1 \left(\frac{1 - \left(e^{[X,\cdot]} e^{t[Y,\cdot]}\right)^{-1}}{\log\left(e^{[X,\cdot]} e^{t[Y,\cdot]}\right)}\right)^{-1} Y dt \tag{B.10}$$

$$= X + \int_0^1 \left(\frac{1 - z^{-1}}{\log z}\right)^{-1} Y dt, \tag{B.11}$$

where $z = e^{[X,\cdot]} e^{t[Y,\cdot]}$. We now temporarily forget that z is an operator, just to obtain an asymptotically exact series that we can evaluate term by term, that is

$$Z(1) = X + \int_0^1 \left(\frac{z \log z}{z - 1}\right) Y dt \tag{B.12}$$

$$= X + \int_0^1 \left(1 + \sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{n(n+1)} (z-1)^n\right) Y dt. \tag{B.13}$$

Finally we can recall the actual definition of z and evaluate the first elements of the

series

$$Z(1) = X + \int_0^1 \left(1 + \frac{1}{2}\left(e^{[X,\cdot]}e^{t[Y,\cdot]} - 1\right) - \frac{1}{6}\left(e^{[X,\cdot]}e^{t[Y,\cdot]} - 1\right)^2 + \ldots\right) Y dt. \tag{B.14}$$

We can now simply expand the exponential maps $e^{[X,\cdot]}$ and $e^{t[Y,\cdot]}$ to the desired order and group the equivalent commutators. The second order expansions of the maps give

$$e^{[X,\cdot]} = 1 + [X,\cdot] + \frac{1}{2}[X,[X,\cdot]] + \ldots \tag{B.15}$$

$$e^{t[Y,\cdot]} = 1 + t[Y,\cdot] + \frac{t^2}{2}[Y,[Y,\cdot]] + \ldots \tag{B.16}$$

The product of these two and its square, up to second order are

$$\begin{aligned}
\left(e^{[X,\cdot]}e^{[Y,\cdot]} - 1\right)Y &= [X,\cdot]\,Y + t\,[Y,\cdot]\,Y + \frac{1}{2}[X,[X,\cdot]]\,Y + \frac{t^2}{2}[Y,[Y,\cdot]]\,Y + t\,[X,[Y,\cdot]]\,Y + \ldots \\
&= [X,Y] + \frac{1}{2}[X,[X,Y]] + \ldots \\
\left(e^{[X,\cdot]}e^{[Y,\cdot]} - 1\right)^2 Y &= [X,[X,\cdot]]\,Y + t^2\,[Y,[Y,\cdot]]\,Y + t\,[X,[Y,\cdot]]\,Y + t\,[Y,[X,\cdot]]\,Y + \ldots \\
&= [X,[X,Y]] + t\,[Y,[X,Y]] + \ldots
\end{aligned}$$

When these expressions (up to second order) are plugged in Eq. B.14, neglecting higher orders, you get

$$\begin{aligned}
Z(1) &\approx X + \int_0^1 \left[Y + \frac{1}{2}[X,Y] + \frac{1}{4}[X,[X,Y]] - \frac{1}{6}[X,[X,Y]] - \frac{t}{6}[Y,[X,Y]]\right] dt \\
&= X + Y + \frac{1}{2}[X,Y] + \frac{1}{12}[X,[X,Y]] - \frac{1}{12}[Y,[X,Y]]\,.
\end{aligned} \tag{B.17}$$

Eq. B.17 contains the first five elements of the explicit BCHD formula. More terms can be obained analogously, by retaining more terms in the expansion of Eqs. B.14 to B.17.

Appendix C

Pair-Decoupled Integration Algorithm[1]

C.1 Modified Symplectic Map

In this section we use the results of derivation we have described in chapter 5 to carry out a symplectic map for the integration of the pair-decoupled system. We define the square matrix A which has the same dimensionality of the Hessian matrix (in whatever coordinate system is used for the decoupling), and write the pair decoupled Hessian matrix as $A \odot \partial^2 V/\partial q^2$, where $\odot$ is the direct product, i.e. given the matrix B and C of same sizes $(B \odot C)_{ij} = B_{ij}C_{ij}$. A is defined such that $A_{ii} = 1$ and $A_{i \neq j} \in [0, 1]$. The matrix A can be interpreted as the adjacency matrix of the graph of the couplings, i.e. if no decoupling is applied, then A is the adjacency matrix of a complete simple undirected graph. Instead, when the decoupling is applied, some of the edges are weighted by α (or absent if $\alpha = 0$).

Let us define a pair-decoupling operator $\hat{D}_\alpha$, which operates on the potential and transforms it into the pair-decoupled potential $\hat{D}_\alpha V = \tilde{V}$. Now we can write the time

[1]This appendix is a reproduction with minor modifications of the content of the supporting information for the paper **Michele Gandolfi**, and **Michele Ceotto**, "Molecular Dynamics of Artificially Pair-Decoupled Systems: An Accurate Tool for Investigating the Importance of Intramolecular Couplings"

propagation for the momentum variable as

$$\begin{aligned}
\tilde{p}(t+b_k\tau) &= e^{-\tau b_k \hat{D}_\alpha \{V,\cdot\}} p(t) = e^{-\tau b_k \left\{\tilde{V},\cdot\right\}} p(t) && \text{(C.1)}\\
&= p(t) - \tau b_k \hat{D}_\alpha \left\{V,p\right\}(t) && \text{(C.2)}\\
&= p(t) - \tau b_k \hat{D}_\alpha \left[-\frac{\partial V}{\partial q}\right](t) = p(t) - \tau b_k \left[-\frac{\partial \tilde{V}}{\partial q}(t)\right] && \text{(C.3)}\\
&\approx p(t) - \tau b_k \hat{D}_\alpha \left[-\frac{\partial V}{\partial q}(t - c_{k-1}\tau) - \frac{d}{dt}\left(\frac{\partial V}{\partial q}\right)(t) c_k \tau\right] && \text{(C.4)}\\
&= p(t) - \tau b_k \left[-\frac{\partial \tilde{V}}{\partial q}(t - c_{k-1}\tau) - \frac{\partial^2 \tilde{V}}{\partial q^2}(t)\dot{q}(t) c_k \tau\right] && \text{(C.5)}\\
&= p(t) - \tau b_k \left[-\frac{\partial \tilde{V}}{\partial q}(t - c_{k-1}\tau) - \left(A \odot \frac{\partial^2 V}{\partial q^2}(t)\right)\dot{q}(t) c_k \tau\right]. && \text{(C.6)}
\end{aligned}$$

In step C.4 we linearized the force $-\partial V/\partial q$ around the time $t - c_{k-1}\tau$, where the c_k are real parameters to be optimized, so that the canonical variables are equal to their corresponding time Taylor expansion up to a given order of τ. Because of this linearization, we say that the force is updated with a local harmonic approximation. In step C.5 we used the abstract definition $\hat{D}_\alpha V = \tilde{V}$, and in the last step we used the definition we gave of pair-decoupling, that is $\left[\partial^2 \tilde{V}/\partial q^2\right]_{ij} = \alpha \left[\partial^2 V/\partial q^2\right]_{ij}$. In Eq. C.6 the pair decoupled force $-\partial \tilde{V}/\partial q$ is evaluated at the previous time $t - c_{k-1}\tau$, and comparing Eq. C.6 with the rightmost hand-side of C.3, the reader sees that the pair decoupled force at time t corresponds to $-\frac{\partial \tilde{V}}{\partial q}(t) \approx \left[-\frac{\partial \tilde{V}}{\partial q}(t - c_{k-1}\tau) - \left(A \odot \frac{\partial^2 V}{\partial q^2}(t)\right)\dot{q}(t) c_k \tau\right]$. For practical purposes we can write this expression as $\tilde{F}_k = \tilde{F}_{k-1} - c_k \tau \partial^2 \tilde{V}/\partial q^2 \cdot \tilde{p}_k/m$. Thus, given an initial value of the pair decoupled force $\tilde{F}_0$, we can propagate it in time.

We can now write the map

$$\mathcal{M}_n \approx e^{-\tau b_n \hat{D}\{V,\cdot\}} e^{-\tau a_n \{K,\cdot\}} \ldots e^{-\tau b_1 \hat{D}\{V,\cdot\}} e^{-\tau a_1 \{K,\cdot\}}, \qquad \text{(C.7)}$$

that depends on the additional c_k coefficients and α. The actual algorithm to update the forces is just a loop over the index k of the following 4 steps.

$$\tilde{p}_k = \tilde{p}_{k-1} + b_k \tau \tilde{F}_{k-1} \tag{C.8}$$

$$\tilde{q}_k = \tilde{q}_{k-1} + a_k \tau \tilde{p}_k / m \tag{C.9}$$

$$\tilde{q}_{aux} = \tilde{q}_k + \tilde{p}_k \tau \sum_j^k (b_j - a_j) / m \tag{C.10}$$

$$\tilde{F}_k = \tilde{F}_{k-1} - c_k \tau \frac{\partial^2 \tilde{V}(q_{k,aux})}{\partial \tilde{q}^2} \cdot \tilde{p}_k / m, \tag{C.11}$$

where $\tilde{q}_k = \tilde{q}\left(\tau \sum_j^k a_j\right)$, $\tilde{p}_k = \tilde{p}\left(\tau \sum_j^k b_j\right)$ and $\tilde{F}_k = \tilde{F}\left(\tau \sum_j^k c_j\right)$. Notice that the force update requires all the variables: force, position and momentum. The last two must be at the same time t. This is the reason why we introduce the ausiliary variable $\tilde{q}_{aux}$, which brings the ausiliary position forward (or backward) in time to match the momentum. In this way the force variable $\tilde{F}(t)$ is integrated alongside $\tilde{q}(t)$ and $\tilde{p}(t)$, on the same footing. However, one should bear in mind that α influences only how the force is updated. Therefore it is necessary to give the initial value of $\tilde{F}$ as an input to the algorithm. A good choice would be to initiate the simulation at a potential minimum, assuming that the pair-decoupled force is zero, just like the normal force. In case another initial condition is chosen, the system would behave as if the pair decoupling were applied when the simulation begins.

To determine the c_k coefficients we focus on the case $\alpha = 1$ (which implies $\tilde{q} \approx q$, $\tilde{p} \approx p$, and $\tilde{F} \approx F$). The reader might have noticed that, for $\tilde{F}_n$ to be most accurate, $c_k \approx a_k$, because $-\tilde{F}_n$ should approximate the gradient at coordinate q_n. This observation is correct for the harmonic oscillator, for which Eq. C.11 matches the exact Taylor series of the force. However, for non-harmonic potentials we found that this is not always the case. In fact, other coefficients might be more appropriate. The reason is that with Eq. C.11 we neglect the contributions coming from the potential derivatives with order higher than 2, thus we necessarily introduce a small error that we do not want to accumulate during the simulation.

The map in Eq. C.7 can be solved for the desired order of k by comparing the position, momentum and force variables with their Taylor expansion, in the same way

as we did in section 5. The second order ($k = 2$) solution is given by the coefficients

$$\begin{cases} a_1 &= \frac{1}{2} \\ a_2 &= \frac{1}{2} \\ b_1 &= 0 \\ b_2 &= 1 \\ c_1 &= \frac{1}{2} \\ c_2 &= \frac{1}{2} \end{cases} \tag{C.12}$$

which corresponds to a Leapfrog algorithm that runs on a locally harmonic potential. Notice that, since we enforce $c_n = a_n$, this solution does not introduce further approximations for harmonic potentials. In fact, for harmonic or bilinear potentials, the dynamics would be indistinguishable from a standard symplectic leapfrog.

Since the position and momentum update operators are not influenced by the force update operator, we can use the set of coefficients $\{a_k\}$ and $\{b_k\}$ we derived in section 5. In particular, we tried both the solution in Eq. 5.48 (also reported in Ref. [162]), and the solution in Eq. 5.49 (also reported in Ref. [167]). Once the $\{a_k\}$ and $\{b_k\}$ are established, we derive the $\{c_k\}$ coefficients to get the 4^{th} order map. Enforcing the coefficients in Eq. 5.48 is straightforward and the optimal solution is unambiguously $\{c_k\} = \{a_k\}$. This is our first choice. On the other hand, enforcing the coefficients in Eq. ,5.49 which are usually superior in terms of energy conservation for molecular dynamics, we find that it is possible to derive an algorithm that is accurate to the second order in τ with respect to position, momentum and force. If we enforce a third order accuracy on the momentum, we get:

$$\begin{aligned} c_1 &= c_1 \\ c_2 &= -\frac{2}{23}(12\sqrt{3}+31)c_1 + \frac{28}{69}\sqrt{3} + \frac{19}{23} \\ c_3 &= -2c_1(4\sqrt{3}+7) + \frac{4}{3}\sqrt{3} + 3 \\ c_4 &= \frac{1}{23}c_1(208\sqrt{3}+361) - \frac{40}{23}\sqrt{3} - \frac{65}{23}, \end{aligned}$$

and if we enforce the third order accuracy on the position, we get

$$
\begin{aligned}
c_1 &= c_1 \\
c_2 &= -c_1 + \frac{1}{6}\sqrt{3} + \frac{1}{2} \\
c_3 &= c_1(4\sqrt{3}+7) - \frac{7}{6}\sqrt{3} - \frac{3}{2} \\
c_4 &= -c_1(4\sqrt{3}+7) + \sqrt{3} + 2.
\end{aligned}
$$

The system is undetermined by one degree of freedom in either case. There are at least five options to approach this problem. The first option is to enforce accuracy up to the third order for both $q(\tau)$ and $p(\tau)$. In this case we obtain a saturated system that is actually accurate to the *fourth* order, with coefficients:

$$
\begin{cases} c_k = a_k \,. \end{cases} \tag{C.13}
$$

Although this solution looks appealing, the integrator is not time-reversible and it works well only for harmonic or bilinear potentials. In the harmonic and bilinear potential cases, Eq. C.11 is not an approximation, and the integrator is as accurate as the one in Ref. [167]. The second option is to enforce third order accuracy in $p(\tau)$ and $c_1 = c_4$.

$$
\begin{cases}
c_1 = \frac{5}{26} \\
c_2 = \frac{8}{39}\sqrt{3} + \frac{4}{13} \\
c_3 = -\frac{8}{39}\sqrt{3} + \frac{4}{13} \\
c_4 = \frac{5}{26}
\end{cases} \tag{C.14}
$$

The third option is to enforce third order accuracy in $p(\tau)$ and $c_1 = -c_4$

$$
\begin{cases}
c_1 = \frac{5}{48}\sqrt{3} \\
c_2 = \frac{1}{8}\sqrt{3} + \frac{1}{2} \\
c_3 = -\frac{1}{8}\sqrt{3} + \frac{1}{2} \\
c_4 = -\frac{5}{48}\sqrt{3}
\end{cases} \tag{C.15}
$$

The fourth option is to enforce third order accuracy in $q(\tau)$ and $c_1 = c_4$:

$$\begin{cases} c_1 &= \frac{1}{4} \\ c_2 &= \frac{1}{6}\sqrt{3} + \frac{1}{4} \\ c_3 &= -\frac{1}{6}\sqrt{3} + \frac{1}{4} \\ c_4 &= \frac{1}{4} \end{cases} \tag{C.16}$$

And finally the fifth option is to enforce third order accuracy in $q(\tau)$ and $c_1 = -c_4$:

$$\begin{cases} c_1 &= \frac{1}{6}\sqrt{3} \\ c_2 &= \frac{1}{2} \\ c_3 &= \frac{1}{2} \\ c_4 &= -\frac{1}{6}\sqrt{3} \end{cases} \tag{C.17}$$

Apart from the first otpion, all solutions are stable enough for general potentials, in the sense that they conserve the total energy over time, with energy fluctuations that are comparable to second order methods, such as the Velocity-Verlet[198] or Symplectic Leapfrog methods. However, we recommend the second and the third options, because they produce much smaller errors, at least for rigid systems.

We call all integration rules described in this appendix the "Symplectic Explicit with Force" integration (SEF) algorithm. If the second order ($k = 2$) scheme is used, the coefficients are reported in Eq.C.12, and we call it "SEF2". If one of the fourth order schemes is used, we call it "SEF4".

C.2 Evolution of the pair-decoupled Monodromy matrix

The SEF algorithm can be easily modified to integrate also the monodromy matrix in a similar fashion of what is described in the Appendix of Ref. [167]. The modified algorithm consists of the following steps iterated twice (for SEF2) or four times (for

SEF4):

$$
\begin{aligned}
\tilde{p}_k &= \tilde{p}_{k-1} + b_k \tau \tilde{F}_{k-1} \\
\tilde{q}_k &= \tilde{q}_{k-1} + a_k \tau \tilde{p}_k
\end{aligned}
$$

$$
\begin{aligned}
q_{tmp} &= \tilde{q}_k + \tilde{p}_k \tau \sum_i^k (b_i - a_i) \\
\tilde{h} &= \frac{\partial^2 \tilde{V}(q_{tmp})}{\partial \tilde{q}^2} \\
\tilde{F}_k &= \tilde{F}_{k-1} - c_k \tau \tilde{h} \cdot \tilde{p}_k
\end{aligned}
$$

$$
\begin{aligned}
\tilde{M}_{pp,k} &= \tilde{M}_{pp,k-1} - b_k \tau \tilde{h} \cdot \tilde{M}_{qp,k-1} \\
\tilde{M}_{pq,k} &= \tilde{M}_{pq,k-1} - b_k \tau \tilde{h} \cdot \tilde{M}_{qq,k-1} \\
\tilde{M}_{qp,k} &= \tilde{M}_{qp,k-1} + a_k \tau \tilde{M}_{pp,k} \\
\tilde{M}_{qq,k} &= \tilde{M}_{qq,k-1} + a_k \tau \tilde{M}_{qp,k},
\end{aligned}
$$

where $\tilde{M}_{pp} = \partial\tilde{p}/\partial p_0$, $\tilde{M}_{pq} = \partial\tilde{p}/\partial q_0$, $\tilde{M}_{qp} = \partial\tilde{q}/\partial p_0$, $\tilde{M}_{qq} = \partial\tilde{q}/\partial q_0$ are four square blocks of the monodromy matrix. $\tilde{M}(0)$ is initialized equal to the canonical symplectic matrix $\mathcal{J}$. Notice that the only information required for the evolution of $\tilde{M}$ is encoded in the pair-decoupled Hessian matrix $\tilde{h}$. In fact, a_n, b_n and τ are parameters of the simulation. Hence, $\tilde{M}$ is the pair-decoupled monodromy matrix. In Figure C.1, we show that SEF2 and SEF4 algorithms preserve the properties $\Upsilon(t)$ and $\tau(t)$ (see Eqs. 6.10 and 6.11) for the H_2O molecule even when the pair of H atoms are decoupled with $\alpha = 0$.

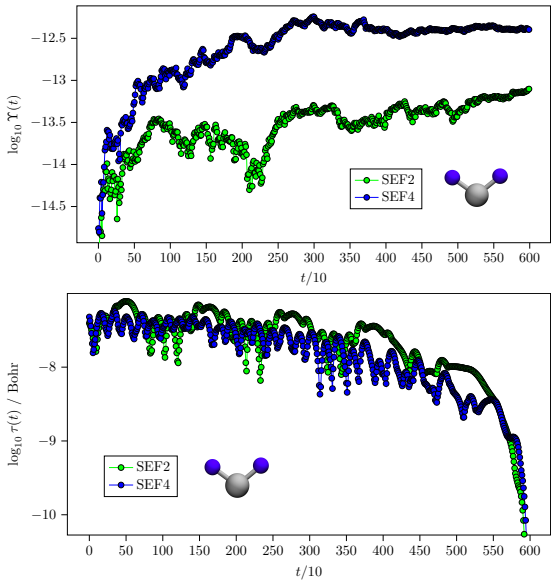


Figure C.1: $\Upsilon(t)$ and $\tau(t)$ for the water molecule with H−H decoupling with $\alpha = 0$. A cartoon of the water molecule is depicted in the bottom part of both graphs with blue colored H atoms to indicate that the H atoms are decoupled. This picture should be considered in comparison with Fig. 1 of the paper

As a proof that the decoupled degrees of freedom are indeed not coupled, we show that the Monodromy matrix can be factorized as a two block matrix preserving the spanned volume: one block of the matrix spans the bending mode and the other the two stretching modes. This is displayed in Fig. C.2, where $\Upsilon(t)$ (see Eq. 6.10) is shown to be close to 0 at all times for both the subsystems.

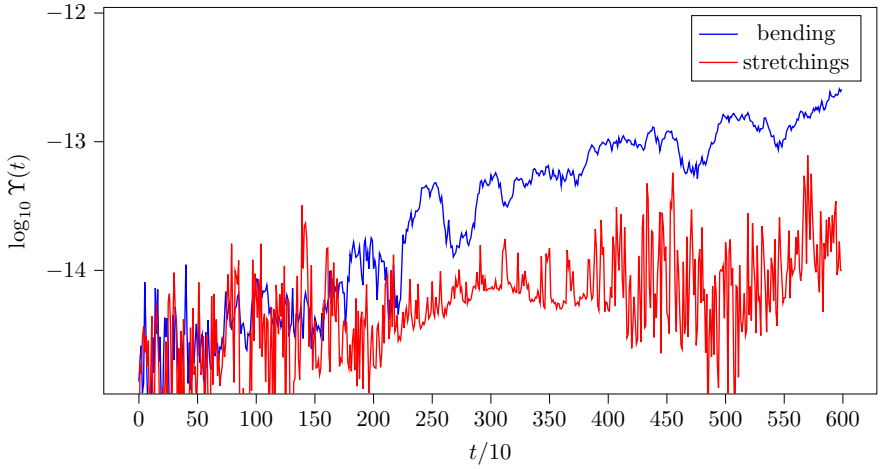


Figure C.2: $\Upsilon(t)$ for the bending and stretching modes of the water molecule

Bibliography

(1) Keutsch, F. N.; Saykally, R. J. *Proceedings of the National Academy of Sciences* **2001**, *98*, 10533–10540.

(2) Richardson, J. O.; Pérez, C.; Lobsiger, S.; Reid, A. A.; Temelso, B.; Shields, G. C.; Kisiel, Z.; Wales, D. J.; Pate, B. H.; Althorpe, S. C. *Science* **2016**, *351*, 1310–1313.

(3) Rognoni, A.; Conte, R.; Ceotto, M. *Chemical Science* **2021**, *12*, 2060–2064.

(4) Lipinski, C.; Hopkins, A. *Nature* **2004**, *432*, 855–861.

(5) Huang, B.; von Lilienfeld, O. A. *Nature chemistry* **2020**, *12*, 945–951.

(6) Von Lilienfeld, O. A.; Müller, K.-R.; Tkatchenko, A. *Nature Reviews Chemistry* **2020**, *4*, 347–358.

(7) Aschi, M.; Spezia, R.; Di Nola, A.; Amadei, A. *Chemical physics letters* **2001**, *344*, 374–380.

(8) Amadei, A.; Daidone, I.; Di Nola, A.; Aschi, M. *Current opinion in structural biology* **2010**, *20*, 155–161.

(9) Morzan, U. N.; Alonso de Armino, D. J.; Foglia, N. O.; Ramirez, F.; Gonzalez Lebrero, M. C.; Scherlis, D. A.; Estrin, D. A. *Chemical reviews* **2018**, *118*, 4071–4113.

(10) Del Galdo, S.; Aschi, M.; Amadei, A. *Chemical Physics Letters* **2021**, *763*, 138168.

(11) Beck, M. H.; Jäckle, A.; Worth, G. A.; Meyer, H.-D. *Physics reports* **2000**, *324*, 1–105.

(12) Wan, Y.; Stratt, R. M. *The Journal of chemical physics* **1994**, *100*, 5123–5138.

(13) Larsen, R. E.; Goodyear, G.; Stratt, R. M. *The Journal of chemical physics* **1996**, *104*, 2987–3002.

(14) Gu, B.; Garashchuk, S. *Theoretical Chemistry Accounts* **2015**, *134*, 1–9.

(15) Zhang, R. M.; Xu, X.; Truhlar, D. G. *The Journal of Physical Chemistry A* **2022**, *126*, 3006–3014.

(16) Meyer, H.-D.; Manthe, U.; Cederbaum, L. S. *Chem. Phys. Lett.* **1990**, *165*, 73–78.

(17) Meng, Q.; Meyer, H.-D. *The Journal of Chemical Physics* **2013**, *138*, 014313.

(18) Baiardi, A.; Reiher, M. *The Journal of Chemical Physics* **2020**, *152*, 040903.
(19) Larsson, H. R. *The Journal of Chemical Physics* **2019**, *151*, 204102.
(20) Garashchuk, S.; Rassolov, V. A. *Chemical physics letters* **2002**, *364*, 562–567.
(21) Garashchuk, S.; Rassolov, V. A. *The Journal of chemical physics* **2003**, *118*, 2482–2490.
(22) Dutra, M.; Wickramasinghe, S.; Garashchuk, S. *Journal of chemical theory and computation* **2019**, *16*, 18–34.
(23) Gatti, F.; Iung, C. *Physics Reports* **2009**, *484*, 1–69.
(24) Rassolov, V. A.; Garashchuk, S.; Schatz, G. C. *The Journal of Physical Chemistry A* **2006**, *110*, 5530–5536.
(25) Mendive-Tapia, D.; Meyer, H.-D.; Vendrell, O. *Journal of Chemical Theory and Computation* **2023**, *19*, 1144–1156.
(26) Merrick, J. P.; Moran, D.; Radom, L. *J. Phys. Chem. A* **2007**, *111*, 11683–11700.
(27) Karplus, M.; Porter, R. N.; Sharma, R. D. *The Journal of Chemical Physics* **1965**, *43*, 3259–3287.
(28) Bonnet, L.; Rayez, J. *Chem. Phys. Lett.* **1997**, *277*, 183–190.
(29) Craig, I. R.; Manolopoulos, D. E. *The Journal of chemical physics* **2004**, *121*, 3368–3373.
(30) Habershon, S.; Manolopoulos, D. E.; Markland, T. E.; Miller III, T. F. *Annu. Rev. Phys. Chem.* **2013**, *64*, 387–413.
(31) Cao, J.; Voth, G. A. *J. Chem. Phys.* **1994**, *100*, 5093–5105.
(32) Witt, A.; Ivanov, S. D.; Shiga, M.; Forbert, H.; Marx, D. *J. Chem. Phys.* **2009**, *130*, 194510.
(33) Van Vleck, J. H. *Proc. Natl. Acad. Sci.* **1928**, *14*, 178–188.
(34) Heller, E. J. *J. Chem. Phys.* **1981**, *75*, 2923–2931.
(35) Herman, M. F.; Kluk, E. *Chem. Phys.* **1984**, *91*, 27–34.
(36) Kaledin, A. L.; Miller, W. H. *J. Chem. Phys.* **2003**, *118*, 7174–7182.
(37) Miller, W. H. *Proc. Natl. Acad. Sci. USA* **2005**, *102*, 6660–6664.
(38) Ceotto, M.; Atahan, S.; Tantardini, G. F.; Aspuru-Guzik, A. *J. Chem. Phys.* **2009**, *130*, 234113.
(39) Begusic, T.; Tapavicza, E.; Vanicek, J. *Journal of Chemical Theory and Computation* **2022**, *18*, 3065–3074.
(40) Kröninger, F.; Lasser, C.; Vanıček, J. J. *Frontiers in Physics* **2023**, *11*, 1106324.
(41) Conte, R.; Aspuru-Guzik, A.; Ceotto, M. *J. Phys. Chem. Lett.* **2013**, *4*, 3407–3412.
(42) Wehrle, M.; Sulc, M.; Vanicek, J. *J. Chem. Phys.* **2014**, *140*, 244114.
(43) Gabas, F.; Conte, R.; Ceotto, M. *J. Chem. Theory Comput.* **2017**, *13*, 2378.
(44) Di Liberto, G.; Conte, R.; Ceotto, M. *J. Chem. Phys.* **2018**, *148*, 104302.

(45) Gabas, F.; Di Liberto, G.; Conte, R.; Ceotto, M. *Chem. Sci.* **2018**, *9*, 7894–7901.
(46) Bertaina, G.; Di Liberto, G.; Ceotto, M. *J. Chem. Phys.* **2019**, *151*, 114307.
(47) Gabas, F.; Di Liberto, G.; Ceotto, M. *J. Chem. Phys.* **2019**, *150*, 224107.
(48) Cazzaniga, M.; Micciarelli, M.; Moriggi, F.; Mahmoud, A.; Gabas, F.; Ceotto, M. *J. Chem. Phys.* **2020**, *152*, 104104.
(49) Ceotto, M.; Di Liberto, G.; Conte, R. *Phys. Rev. Lett.* **2017**, *119*, 010401.
(50) Di Liberto, G.; Conte, R.; Ceotto, M. *J. Chem. Phys.* **2018**, *148*, 014307.
(51) Gandolfi, M.; Rognoni, A.; Aieta, C.; Conte, R.; Ceotto, M. *The Journal of Chemical Physics* **2020**, *153*, 204104.
(52) Rognoni, A.; Conte, R.; Ceotto, M. *The Journal of Chemical Physics* **2021**, *154*, 094106.
(53) Landau, L. D.; Lifshitz, E. M., *Mechanics*; Elsevier: 1982.
(54) Martin, J.; Lee, T. J.; Taylor, P. R. *Journal of Molecular Spectroscopy* **1993**, *160*, 105–116.
(55) Bussi, G.; Donadio, D.; Parrinello, M. *The Journal of chemical physics* **2007**, *126*.
(56) Galimberti, D. R.; Milani, A.; Tommasini, M.; Castiglioni, C.; Gaigeot, M.-P. *Journal of chemical theory and computation* **2017**, *13*, 3802–3813.
(57) Conte, R.; Parma, L.; Aieta, C.; Rognoni, A.; Ceotto, M. *J. Chem. Phys.* **2019**, *151*, 214107.
(58) Kaledin, A. L.; Miller, W. H. *J. Chem. Phys.* **2003**, *119*, 3078–3084.
(59) Kluk, E.; Herman, M. F.; Davis, H. L. *J. Chem. Phys.* **1986**, *84*, 326–334.
(60) Kay, K. G. *J. Chem. Phys.* **1994**, *101*, 2250–2260.
(61) Kay, K. G. *J. Chem. Phys.* **1994**, *100*, 4377–4392.
(62) Kay, K. G. *J. Chem. Phys.* **1994**, *100*, 4432–4445.
(63) Kay, K. G. *Chem. Phys.* **2006**, *322*, 3–12.
(64) Heller, E. J. *J. Chem. Phys.* **1991**, *94*, 2723–2729.
(65) Di Liberto, G.; Ceotto, M. *J. Chem. Phys.* **2016**, *145*, 144107.
(66) De Leon, N.; Heller, E. J. *J. Chem. Phys.* **1983**, *78*, 4005–4017.
(67) Ceotto, M.; Atahan, S.; Shim, S.; Tantardini, G. F.; Aspuru-Guzik, A. *Phys. Chem. Chem. Phys.* **2009**, *11*, 3861–3867.
(68) Ceotto, M.; Tantardini, G. F.; Aspuru-Guzik, A. *J. Chem. Phys.* **2011**, *135*, 214108.
(69) Sinkhorn, R.; Knopp, P. *Pacific Journal of Mathematics* **1967**, *21*, 343–348.
(70) Higham, N. J. *Wiley Interdisciplinary Reviews: Computational Statistics* **2011**, *3*, 230–238.
(71) Cayley, A. *Quart. J. Math.* **1889**, *23*, 376–378.
(72) Sokal, R. R. *Univ. Kansas, Sci. Bull.* **1958**, *38*, 1409–1438.

(73) Lee, T. J.; Martin, J. M.; Taylor, P. R. *J. Chem. Phys.* **1995**, *102*, 254–261.
(74) Gray, D.; Robiette, A. *Molecular Physics* **1979**, *37*, 1901–1920.
(75) Raynes, W.; Lazzeretti, P.; Zanasi, R.; Sadlej, A.; Fowler, P. *Molecular Physics* **1987**, *60*, 509–525.
(76) Carter, S.; Shnider, H. M.; Bowman, J. M. *J. Chem. Phys.* **1999**, *110*, 8417–8423.
(77) Qu, C.; Bowman, J. M. *J. Chem. Phys.* **2019**, *150*, 141101.
(78) Chen, X.; Schweitzer-Stenner, R.; Asher, S. A.; Mirkin, N. G.; Krimm, S. *The Journal of Physical Chemistry* **1995**, *99*, 3074–3083.
(79) Kubelka, J.; Keiderling, T. A. *The Journal of Physical Chemistry A* **2001**, *105*, 10922–10928.
(80) Torii, H.; Tatsumi, T.; Kanazawa, T.; Tasumi, M. *The Journal of Physical Chemistry B* **1998**, *102*, 309–314.
(81) Ataka, S.; Takeuchi, H.; Tasumi, M. *Journal of Molecular Structure* **1984**, *113*, 147–160.
(82) Mayne, L. C.; Hudson, B. *The Journal of Physical Chemistry* **1991**, *95*, 2962–2967.
(83) Triggs, N. E.; Valentini, J. J. *The journal of physical chemistry* **1992**, *96*, 6922–6931.
(84) Nandi, A.; Qu, C.; Bowman, J. M. *J. Chem. Phys.* **2019**, *151*, 084306.
(85) Wallace, W. E. *NIST Chemistry WebBook, NIST Standard Reference Database Number 69, Eds. P.J. Linstrom and W.G. Mallard, National Institute of Standards and Technology, Gaithersburg MD, 20899* **2020**.
(86) Kaledin, A.; Bowman, J. *The Journal of Physical Chemistry A* **2007**, *111*, 5593–5598.
(87) Conte, R.; Botti, G.; Ceotto, M. *Vibrational Spectroscopy* **2020**, *106*, 103015.
(88) Wang, Y.; Bowman, J. M. *J. Chem. Phys.* **2012**, *136*, 144113.
(89) Liu, H.; Wang, Y.; Bowman, J. M. *J. Phys. Chem. Lett.* **2012**, *3*, 3671–3676.
(90) Wang, Y.; Bowman, J. M. *J. Phys. Chem. Lett.* **2013**, *4*, 1104–1108.
(91) Begusic, T.; Roulet, J.; Vanicek, J. *J. Chem. Phys.* **2018**, *149*, 244115.
(92) Patoz, A.; Begusic, T.; Vanicek, J. *J. Phys. Chem. Lett.* **2018**, *9*, 2367–2372.
(93) Wehrle, M.; Oberli, S.; Vanicek, J. *J. Phys. Chem. A* **2015**, *119*, 5685–5690.
(94) Martinetz, T.; Schulten, K. *Artificial Neural Networks* **1991**, 397–402.
(95) Gandolfi, M.; Ceotto, M. *Journal of Chemical Theory and Computation* **2021**, *17*, 6733–6746.
(96) Lourderaj, U.; Song, K.; Windus, T. L.; Zhuang, Y.; Hase, W. L. *The Journal of chemical physics* **2007**, *126*, 044105.
(97) Bakken, V.; Millam, J. M.; Bernhard Schlegel, H. *The Journal of chemical physics* **1999**, *111*, 8773–8777.

(98) Hratchian, H.; Schlegel, H. *Journal of chemical theory and computation* **2005**, *1*, 61–69.
(99) Schlegel, H. B. *Theoretica chimica acta* **1984**, *66*, 333–340.
(100) Schlegel, H. B. **in***Modern Electronic Structure Theory: Part I* World Scientific: 1995, **pages** 459–500.
(101) Kindt, J.; Schmuttenmaer, C. *The Journal of chemical physics* **1997**, *106*, 4389–4400.
(102) Imoto, S.; Marx, D. *The Journal of chemical physics* **2019**, *150*, 084502.
(103) Allen, A. E.; Payne, M. C.; Cole, D. J. *Journal of chemical theory and computation* **2018**, *14*, 274–281.
(104) Miller, W. H. *J. Phys. Chem. A* **2001**, *105*, 2942–2955.
(105) Richardson, J. O. *International reviews in physical chemistry* **2018**, *37*, 171–216.
(106) Ceotto, M. *Molecular Physics* **2012**, *110*, 547–559.
(107) Gaigeot, M.-P. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy* **2021**, 119864.
(108) Wu, H.; Rahman, M.; Wang, J.; Louderaj, U.; Hase, W.; Zhuang, Y. *J. Chem. Phys.* **2010**, *133*, 074101.
(109) Zhuang, Y.; Siebert, M. R.; Hase, W. L.; Kay, K. G.; Ceotto, M. *J. Chem. Theory Comput.* **2012**, *9*, 54–64.
(110) Ceotto, M.; Zhuang, Y.; Hase, W. L. *J. Chem. Phys.* **2013**, *138*, 054116.
(111) Conte, R.; Gabas, F.; Botti, G.; Zhuang, Y.; Ceotto, M. *J. Chem. Phys.* **2019**, *150*, 244118.
(112) Broyden, C. G. *Mathematics of computation* **1965**, *19*, 577–593.
(113) Powell, M. J. *Mathematical Programming* **1971**, *1*, 26–57.
(114) Dennis Jr, J. E.; Moré, J. J. *SIAM review* **1977**, *19*, 46–89.
(115) Nocedal, J. *Acta numerica* **1992**, *1*, 199–242.
(116) Bofill, J. M. *Journal of Computational Chemistry* **1994**, *15*, 1–11.
(117) Denzel, A.; Kästner, J. *Journal of Chemical Theory and Computation* **2020**, *16*, 5083–5089.
(118) Quinonero-Candela, J.; Rasmussen, C. E. *The Journal of Machine Learning Research* **2005**, *6*, 1939–1959.
(119) Williams, C. K.; Rasmussen, C. E., *Gaussian processes for machine learning*; 3; MIT press Cambridge, MA: 2006; **volume** 2.
(120) Deringer, V. L.; Bartók, A. P.; Bernstein, N.; Wilkins, D. M.; Ceriotti, M.; Csányi, G. *Chemical Reviews* **2021**.
(121) Laude, G.; Calderini, D.; Tew, D. P.; Richardson, J. O. *Faraday discussions* **2018**, *212*, 237–258.

(122) Martinetz, T. M.; Berkovich, S. G.; Schulten, K. J. *IEEE transactions on neural networks* **1993**, *4*, 558–569.
(123) Fritzke, B. *Advances in neural information processing systems* **1995**, *7*, 625–632.
(124) Martinetz, T.; Schulten, K. *Neural Networks* **1994**, *7*, 507–522.
(125) Prudent, Y.; Ennaji, A. **in***Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.* 2005; **volume** 2, **pages** 1211–1216.
(126) Marsland, S.; Shapiro, J.; Nehmzow, U. *Neural networks* **2002**, *15*, 1041–1058.
(127) Cottrell, M.; Hammer, B.; Hasenfuß, A.; Villmann, T. *Neural Networks* **2006**, *19*, 762–771.
(128) Tropsha, A. *Molecular informatics* **2010**, *29*, 476–488.
(129) Todeschini, R.; Consonni, V., *Handbook of molecular descriptors*; John Wiley & Sons: 2008; **volume** 11.
(130) Cherkasov, A.; Muratov, E. N.; Fourches, D.; Varnek, A.; Baskin, I. I.; Cronin, M.; Dearden, J.; Gramatica, P.; Martin, Y. C.; Todeschini, R. *Journal of medicinal chemistry* **2014**, *57*, 4977–5010.
(131) Ekins, S.; Puhl, A. C.; Zorn, K. M.; Lane, T. R.; Russo, D. P.; Klein, J. J.; Hickey, A. J.; Clark, A. M. *Nature materials* **2019**, *18*, 435–441.
(132) Dara, S.; Dhamercherla, S.; Jadav, S. S.; Babu, C.; Ahsan, M. J. *Artificial Intelligence Review* **2021**, 1–53.
(133) Von Lilienfeld, O. A. *Angewandte Chemie International Edition* **2018**, *57*, 4164–4169.
(134) Botu, V.; Ramprasad, R. *International Journal of Quantum Chemistry* **2015**, *115*, 1074–1083.
(135) Ye, S.; Hu, W.; Li, X.; Zhang, J.; Zhong, K.; Zhang, G.; Luo, Y.; Mukamel, S.; Jiang, J. *Proceedings of the National Academy of Sciences* **2019**, *116*, 11612–11617.
(136) Gastegger, M.; Behler, J.; Marquetand, P. *Chemical science* **2017**, *8*, 6924–6935.
(137) De, S.; Bartók, A. P.; Csányi, G.; Ceriotti, M. *Physical Chemistry Chemical Physics* **2016**, *18*, 13754–13769.
(138) Moosavi, S. M.; Nandy, A.; Jablonka, K. M.; Ongari, D.; Janet, J. P.; Boyd, P. G.; Lee, Y.; Smit, B.; Kulik, H. J. *Nature communications* **2020**, *11*, 1–10.
(139) Gómez-Bombarelli, R.; Wei, J. N.; Duvenaud, D.; Hernández-Lobato, J. M.; Sánchez-Lengeling, B.; Sheberla, D.; Aguilera-Iparraguirre, J.; Hirzel, T. D.; Adams, R. P.; Aspuru-Guzik, A. *ACS central science* **2018**, *4*, 268–276.
(140) Conte, R.; Ceotto, M., *Semiclassical Molecular Dynamics for Spectroscopic Calculations*; Wiley: book chapter, accepted.
(141) Lele, S. K. *Journal of computational physics* **1992**, *103*, 16–42.

(142) Lynch, R. E.; Rice, J. R. *Proceedings of the National Academy of Sciences* **1978**, *75*, 2541–2544.
(143) Zhuang, Y.; Sun, X.-H. *Advances in Engineering Software* **2000**, *31*, 585–591.
(144) Zhuang, Y.; Sun, X.-H. *Journal of Computational Physics* **2001**, *171*, 79–94.
(145) Karandashev, K.; Vanicek, J. *J. Chem. Phys.* **2019**, *151*, 174116.
(146) Bowman, J. M.; Wierzbicki, A.; Zuniga, J. *Chem. Phys. Lett.* **1988**, *150*, 269–274.
(147) Conte, R.; Qu, C.; Houston, P. L.; Bowman, J. M. *Journal of chemical theory and computation* **2020**, *16*, 3264–3272.
(148) Valiev, M.; Bylaska, E.; Govind, N.; Kowalski, K.; Straatsma, T.; Dam, H. V.; Wang, D.; Nieplocha, J.; Apra, E.; Windus, T.; de Jong, W. *Comput. Phys. Commun.* **2010**, *181*, 1477–1489.
(149) Biswal, H. S.; Gloaguen, E.; Loquais, Y.; Tardivel, B.; Mons, M. *J. Phys. Chem. Lett.* **2012**, *3*, 755–759.
(150) Begusic, T.; Cordova, M.; Vanicek, J. *J. Chem. Phys.* **2019**, *150*, 154117.
(151) Donnelly, D.; Rogers, E. *American Journal of Physics* **2005**, *73*, 938–945.
(152) Suzuki, M. *Communications in Mathematical Physics* **1976**, *51*, 183–190.
(153) Hatano, N.; Suzuki, M. **in***Quantum annealing and other optimization methods* Springer: 2005, **pages** 37–68.
(154) Cromer, A. *American Journal of Physics* **1981**, *49*, 455–459.
(155) McLachlan, R. I.; Quispel, G. R. W. *Acta Numerica* **2002**, *11*, 341–434.
(156) Strang, G. *SIAM journal on numerical analysis* **1968**, *5*, 506–517.
(157) Suzuki, M. *Physics Letters A* **1990**, *146*, 319–323.
(158) Yoshida, H. *Physics letters A* **1990**, *150*, 262–268.
(159) Chin, S. A. *American Journal of Physics* **2020**, *88*, 883–894.
(160) Creutz, M. *Lattice fields and strong interactions*; techreport; Brookhaven National Lab., Upton, NY (USA), 1989.
(161) Rangarajan, G.; Neri, F.; Dragt, A. *Part. Accel.* **1990**, *28*, 119–124.
(162) Forest, E.; Ruth, R. D. *Physica D: Nonlinear Phenomena* **1990**, *43*, 105–117.
(163) Campostrini, M.; Rossi, P. *Nuclear Physics B* **1990**, *329*, 753–764.
(164) Candy, J.; Rozmus, W. *Journal of Computational Physics* **1991**, *92*, 230–256.
(165) Dragt, A.; Neri, F.; Rangarajan, G.; Douglas, D. R.; Healy, L. M.; Ryne, R. D. *Annual Review of Nuclear and Particle Science* **1988**, *38*, 455–496.
(166) The Sage Developers SageMath, the Sage Mathematics Software System (Version 9.5), `https://www.sagemath.org`, 2022.
(167) Brewer, M. L.; Hulme, J. S.; Manolopoulos, D. E. *J. Chem. Phys.* **1997**, *106*, 4832–4839.

(168) Hamm, P.; Lim, M.; Hochstrasser, R. M. *The Journal of Physical Chemistry B* **1998**, *102*, 6123–6138.
(169) Ghosh, A.; Ostrander, J. S.; Zanni, M. T. *Chemical reviews* **2017**, *117*, 10726–10759.
(170) Ramesh, P.; Loring, R. F. *The Journal of Physical Chemistry B* **2018**, *122*, 3647–3654.
(171) Jansen, T. l. C.; Saito, S.; Jeon, J.; Cho, M. *The Journal of Chemical Physics* **2019**, *150*, 100901.
(172) Milo, A.; Bess, E. N.; Sigman, M. S. *Nature* **2014**, *507*, 210–214.
(173) Bess, E. N.; Guptill, D. M.; Davies, H. M.; Sigman, M. S. *Chemical science* **2015**, *6*, 3057–3062.
(174) Yada, A.; Nagata, K.; Ando, Y.; Matsumura, T.; Ichinoseki, S.; Sato, K. *Chemistry Letters* **2018**, *47*, 284–287.
(175) Torrie, G. M.; Valleau, J. P. *Journal of Computational Physics* **1977**, *23*, 187–199.
(176) Voter, A. F. *The Journal of chemical physics* **1997**, *106*, 4665–4677.
(177) Ceotto, M.; Ayton, G. S.; Voth, G. A. *Journal of Chemical Theory and Computation* **2008**, *4*, 560–568.
(178) Barducci, A.; Bonomi, M.; Parrinello, M. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2011**, *1*, 826–843.
(179) Moscato, D.; Gabas, F.; Conte, R.; Ceotto, M. *Journal of Biomolecular Structure and Dynamics* **2023**.
(180) Dressler, S.; Thiel, W. *Chem. Phys. Lett.* **1997**, *273*, 71–78.
(181) Chmiela, S.; Tkatchenko, A.; Sauceda, H. E.; Poltavsky, I.; Schütt, K. T.; Müller, K.-R. *Science Advances* **2017**, *3*, e1603015.
(182) Chmiela, S.; Sauceda, H. E.; Müller, K.-R.; Tkatchenko, A. *Nature Communications* **2018**, *9*, 3887.
(183) Chmiela, S.; Sauceda, H. E.; Poltavsky, I.; Müller, K.-R.; Tkatchenko, A. *Computer Physics Communications* **2019**, *240*, 38–45.
(184) Sauceda, H. E.; Chmiela, S.; Poltavsky, I.; Müller, K.-R.; Tkatchenko, A. *The Journal of Chemical Physics* **2019**, *150*, 114102.
(185) Chmiela, S.; Sauceda, H. E.; Tkatchenko, A.; Müller, K.-R. **in***Machine Learning Meets Quantum Physics* Springer International Publishing: 2020, **pages** 129–154.
(186) Sauceda, H. E.; Chmiela, S.; Poltavsky, I.; Müller, K.-R.; Tkatchenko, A. **in***Machine Learning Meets Quantum Physics* Springer International Publishing: 2020, **pages** 277–307.

(187) Sauceda, H. E.; Gastegger, M.; Chmiela, S.; Müller, K.-R.; Tkatchenko, A. *The Journal of Chemical Physics* **2020**, *153*, 124109.
(188) Tkatchenko, A.; Scheffler, M. *Physical review letters* **2009**, *102*, 073005.
(189) Yahagi, T.; Fujii, A.; Ebata, T.; Mikami, N. *The Journal of Physical Chemistry A* **2001**, *105*, 10673–10680.
(190) Sobolewski, A. L.; Domcke, W. *The Journal of Physical Chemistry A* **2004**, *108*, 10917–10922.
(191) Abou El, E. A. E.-H.; Fujii, A.; Ebata, T.; Mikami, N. *Chemical physics letters* **2003**, *376*, 788–793.
(192) Abd El-Hakam Abou El-Nasr, E.; Fujii, A.; Ebata, T.; Mikami*, N. *Molecular Physics* **2005**, *103*, 1561–1572.
(193) Raeker, T.; Hartke, B. *The Journal of Physical Chemistry A* **2017**, *121*, 5967–5977.
(194) Kubo, R. *Journal of the Physical Society of Japan* **1957**, *12*, 570–586.
(195) Qu, C.; Bowman, J. M. *J. Phys. Chem. A* **2016**, *120*, 4988–4993.
(196) Santos, J.; Tiznado, W.; Contreras, R.; Fuentealba, P. *The Journal of chemical physics* **2004**, *120*, 1670–1673.
(197) Hall, B. C. *arXiv preprint math-ph/0005032* **2000**, 56–64.
(198) Verlet, L. *Physical review* **1967**, *159*, 98.