GDAŃSK UNIVERSITY OF TECHNOLOGY

The author of the doctoral dissertation: Jan Kozicki
Scientific discipline: Physics

DOCTORAL DISSERTATION

Title of doctoral dissertation: Numerical modeling of quantum dynamical processes

Title of doctoral dissertation (in Polish): Modelowanie numeryczne procesów dynamiki kwantowej

Supervisor *signature*	Second supervisor *signature*
prof. dr hab. J.E. Sienkiewicz	dr inż. Patryk Jasik

Gdańsk, year 2022

GDAŃSK UNIVERSITY OF TECHNOLOGY

STATEMENT

The author of the doctoral dissertation: Jan Kozicki

I, the undersigned, declare that I am aware that in accordance with the provisions of Art. 27 (1) and (2) of the Act of 4th February 1994 on Copyright and Related Rights (Journal of Laws of 2021, item 1062), the university may use my doctoral dissertation entitled:

"Numerical modeling of quantum dynamical processes"

for scientific or didactic purposes.[1]

Gdańsk, ...

. . . . . .
signature of the PhD student

Aware of criminal liability for violations of the Act of 4th February 1994 on Copyright and Related Rights and disciplinary actions set out in the Law on Higher Education and Science (Journal of Laws 2021, item 478), as well as civil liability, I declare, that the submitted doctoral dissertation is my own work.
I declare, that the submitted doctoral dissertation is my own work performed under and in cooperation with the supervision of prof. dr hab. J.E. Sienkiewicz and the second supervisor dr inż. Patryk Jasik.

This submitted doctoral dissertation has never before been the basis of an official procedure associated with the awarding of a PhD degree.

All the information contained in the above thesis which is derived from written and electronic sources is documented in a list of relevant literature in accordance with Art. 34 of the Copyright and Related Rights Act.

I confirm that this doctoral dissertation is identical to the attached electronic version.

Gdańsk, ...

. . . . . .
signature of the PhD student

I, the undersigned, agree to include an electronic version of the above doctoral dissertation in the open, institutional, digital repository of Gdańsk University of Technology.

Gdańsk, ...

. . . . . .
signature of the PhD student

[1] Art 27. 1. Educational institutions and entities referred to in art. 7 sec. 1 points 1, 2 and 4–8 of the Act of 20 July 2018 – Law on Higher Education and Science, may use the disseminated works in the original and in translation for the purposes of illustrating the content provided for didactic purposes or in order to conduct research activities, and to reproduce for this purpose disseminated minor works or fragments of larger works.
2. If the works are made available to the public in such a way that everyone can have access to them at the place and time selected by them, as referred to in para. 1, is allowed only for a limited group of people learning, teaching or conducting research, identified by the entities listed in paragraph 1.

GDAŃSK UNIVERSITY OF TECHNOLOGY

DESCRIPTION OF DOCTORAL DISSERTATION

The author of the doctoral dissertation: Jan Kozicki

Title of doctoral dissertation: Numerical modeling of quantum dynamical processes

Title of doctoral dissertation in Polish: Modelowanie numeryczne procesów dynamiki kwantowej

Language of doctoral dissertation: English

Supervisor: prof. dr hab. J.E. Sienkiewicz

Second supervisor: dr inż. Patryk Jasik

Date of doctoral defense: .....

Keywords of doctoral dissertation in Polish: dynamika kwantowa, metody numeryczne, hamiltonian zależny od czasu, sprzężone równania Schrödingera, oddziaływanie molekuły z polem, fotoasocjacja, rozpad potęgowy, molekuła dwuatomowa, sprzężenie spin-orbita, moment przejścia dipolowego

Keywords of doctoral dissertation in English: quantum dynamics, numerical methods, time-dependent Hamiltonian, coupled Schrödinger equations, field-molecule interaction, photoassociation, power-law decay, diatomic molecule, spin-orbit coupling, transition dipole moment

Summary of doctoral dissertation in Polish: W niniejszej pracy doktorskiej prezentuję napisaną w C++ w wysokiej precyzji (15, 18 lub 33 miejsca znaczące) implementację algorytmów do propagacji w czasie równania Schrödingera zarówno dla Hamiltonianu niezależnego od czasu jak i zależnego od czasu z członem źródłowym. Co więcej, prezentuję rozszerzenie tych algorytmów pozwalające na liczenie dowolnej ilości sprzężonych równań Schrödingera na wielu poziomach elektronowych. Przeprowadziłem dokładną walidację tych algorytmów na wynikach referencyjnych m.in.: oscylator wymuszony z członem źródłowym lub atom znajdujący się w silnym polu laserowym. Następnie stosuję te nowe algorytmy do obliczenia rowibracyjnej predysocjacji molekuły KLi i porównania jej z eksperymentem. Pokazałem, że kwazi-związane stany molekuły KLi rozpadają się wykładniczo w czasie oraz wskazałem niezawodną metodę do wyznaczania ich czasów życia. Następnie policzyłem rozpad molekuły NaRb używając do tego trzy sprzężone stany elektronowe. Przeprowadziłem dopasowanie jej rozpadu i pokazałem, że dopasowanie z zastosowaniem prawa rozpadu potęgowego jest lepsze niż prawa rozpadu wykładniczego. Mam nadzieję, że wyniki te będą miały duże znaczenie przy projektowaniu przyszłych eksperymentów.

Summary of doctoral dissertation in English: In this dissertation I present a high-precision (15, 18 or 33 decimal places) C++ implementation of quantum dynamics time propagation algorithms for both time-independent and time-dependent Hamiltonian with an inhomogeneous source term. Moreover I present an extension of both algorithms for time propagation to handle arbitrary number of coupled electronic levels. I have performed a careful validation of these implementations comparing my results with numerous reference results such as: a forced harmonic oscillator with an inhomogeneous source term or an atom in an intense laser field. Next, I apply these new algorithms to calculate the rovibrational predissociation in the KLi molecule and compare it with experiment. In doing so I have shown that the KLi quasi-bound state decays exponentially in time and I have described a robust method to calculating the widths of quasi-bound vibrational levels. Also I have calculated the decay in the NaRb dimer using three coupled electronic states. I have performed a fit of this decay to a modified power-law and I have shown that this fit is better than a fit to the exponential decay. I hope that those results are of considerable relevance to the design of experiments.

Gdańsk University of Technology
Faculty of Applied Physics and Mathematics

Jan Kozicki

Numerical modeling of quantum dynamical processes

Doctoral thesis

supervisors: prof. dr hab. J.E. Sienkiewicz
dr inż. Patryk Jasik

Gdańsk, October 1, 2022

Contents

List of Figures

List of Tables

Code listings

for my wife, Asia

Chapter 1

Introduction

Quantum dynamics is a growing discipline at the interface of chemistry, physics and materials science [1, 2]. It allows to study the behavior of objects in a way, that emphasizes the quantum nature of their evolution in time. Quantum dynamics simulations are an indispensable tool for investigating processes such as chemical reactions [3], field-atoms interactions [4] and quantum computing [5, 6]. The particular emphasis is focused on investigating the photoinduced dynamics of breaking (dissociation) [7, 8, 9, 10, 11, 12] and creating (association) [13, 14, 15, 16, 17, 18] of the chemical bonds in the molecular systems. Photoassociation processes play a key role in the field of cold and ultracold physics and chemistry, allowing for the formation of molecules in the deeply bound ground states and investigation of their unique quantum properties [19, 20, 21, 22, 23]. On the other side, controlled photodissociation reactions enable for creation of atomic and molecular fragments in specific quantum states and use them for researching the selected properties of matter [7, 24, 25, 26, 27].

Proper, precise and accurate modeling of quantum dynamics has always been a challenge. The difficulties start when the need arises to take into account multiple coupled electronic levels with time dependent coupling or when there are more degrees of freedom, which often cause a so-called tensor explosion. The problem which I aim to solve in this thesis is the ability to simulate the time evolution of quantum systems, taking into account multiple electronic levels and their time-dependent coupling. I do so in a manner that allows high-precision calculations [28] and allows me to examine the evolution of the quantum system in great detail [29, 30, 31, 32]. The problem of tensor explosion is not tackled, and albeit the code can relatively simply be modified to allow more degrees of freedom, present calculations are done in one nuclear coordinate with arbitrary number of electronic levels. In fact an unmerged branch of YADE software, which I develop since year 2008 [28, 33, 34, 35] and which mostly deals with classical dynamics, has quantum dynamics code[1] for higher dimensional quantum dynamics. The tensor explosion requires big amounts of RAM and the largest system calculated in this implementation is a four dimensional positron electron scattering which among other example quantum dynamics scripts can be found here[2,3].

Several parts of my thesis are published in papers: the KLi tunneling from Section 4.1 is published in [29], the NaRb power-law decay from Section 4.2 is in preparations to publish and has been presented on a poster in [30] while the high and arbitrary precision numerical

[1] https://gitlab.com/cosurgi/trunk/-/merge_requests/46
[2] https://gitlab.com/cosurgi/trunk/-/tree/addNRQM/examples/qm
[3] Both particles can move in two dimensions, which amounts to four dimensions total.

calculations from Sections 2.1.1, 2.3 and 3.5 are published in [28]. Other relevant papers which used my work from Section 2.1 and Section 4.1 are published in [31, 32].

These topics fit into the program of the AttoChem research network within the European Cooperation in Science and Technology (COST), which brings together experimentalists and theorists to study the time-dependent techniques for calculating charge migration in molecules. The main goal of the AttoChem COST action is to enable design of attosecond light pulses and attosecond control strategies that can be efficiently used to induce unusual charge migration dynamics in molecules, isolated or in a solution, so that one can selectively break and form chemical bonds, thus leading to new chemical reactions and avoiding the typical ones [36]. Furthering this knowledge can have a significant impact on understanding of breaking and creation of chemical bonds occurring in several areas of physics, like photovoltaics, photophysics, structure determination and biochemistry radiation damage.

While participating in the AttoChem COST I took the opportunity to attend the summer "School on New Computational Methods for Attosecond Molecular Processes" [37], also I have presented the poster at the AttoChem conference meeting [30] and was encouraged by the COST action chairman prof. Fernando Martin to contribute in the special edition of Computer Physics Communications describing my quantum dynamics code, which is the subject of this dissertation.

The goals of my dissertation are to (1) develop a generic high-precision C++ code for time propagation of time-dependent and time-independent Hamiltonian with arbitrary number of coupled Schrödinger equations and to (2) demonstrate the capabilities of the new code on the KLi and NaRb diatomic molecules and to (3) compare the results with corresponding reference results and the experiment.

There are several other approaches to simulate the atoms, molecules and larger molecular systems. I have chosen to work directly, ab initio, with the coupled time dependent Schrödinger equations [38, 39, 40, 41]. However there is also the time dependent density functional theory (TDDFT) [42, 43, 44, 45] which is good at solving the so-called tensor explosion problem by reducing the representation into a single-electron wavefunction. There is also the multiconfiguration time-dependent Hartree (MCTDH) [46, 47, 48, 49] which also is good at solving the tensor explosion and allows to simulate up to about 12 degrees of freedom, by performing certain approximations. Finally for even larger systems there are classic molecular dynamics approaches [50] which can deal with very large molecular systems by modeling the quantum behavior with a proper selection of classical potentials between the atoms. I choose the ab initio Schrödinger approach mainly for its accuracy, since all the other methods are approximate.

There are several software packages allowing to solve the time dependent Schrödinger equation, such as Wavepacket [51, 52, 53], MOLPRO [54], t-SURFF [55, 56] or tRecX [57]. As well as other software packages dedicated to other numerical methods mentioned above, such as Octopus [43] and XCHEM [58]. None of these packages, however, have high precision computation support (18 or 33 significant digits) and most of them are written in slow interpreted languages such as Python or old and lacking static polymorphism languages[4] such as Fortran. Only software written by Scrinzi [56, 57] uses modern C++ language and has potential to support in the future high precision computations. However it is not dealing with the exact method which I need in my thesis, namely: the time propagation of time

[4]The rule of thumb being that languages without static polymorphism are either fast or generic, but they cannot be both.

dependent Schrödinger equation with time dependent Hamiltonian for arbitrary number of coupled electronic states. Therefore I want to write a code which deals with this method using high precision in a modern and fast C++ language. This approach will allow me to easily extend this software in the future and enable me to use modern mathematical libraries such as Eigen [59] and Boost [60].

There are several novelties in my thesis. I present a high-precision C++ implementation of quantum dynamics time propagation algorithms for both time-independent and time-dependent Hamiltonian with an inhomogeneous source term. Moreover I present an extension of both algorithms for time propagation to handle unlimited[5] coupled electronic levels. Also I apply these new algorithms to calculate the rovibrational predissociation in the KLi molecule and compare it with experiment. Finally I am proposing a new modified power-law decay to describe the dissociation of a chosen model NaRb molecule. All this work has full high precision support: 15, 18 or 33 decimal places, with reasonable perspective to add arbitrary precision[6]

This thesis is divided into three main parts: the theory of time propagation algorithms is in Chapter 2, the implementation with a code validation is presented in Chapter 3 and the application of these algorithms to dissociation in the KLi dimer and a power-law decay in the NaRb dimer is presented in Chapter 4. The first two chapters have a dedicated section to high-precision calculations. Finally the conclusions are presented in Chapter 5

[5] The limit is the computer's memory.

[6] Only FFT for arbitrary precision is missing and is currently under development in the Google Summer of Code programme (GSoC), with me as a mentor: `https://github.com/BoostGSoC21/math/issues`, see also the Boost GSoC report [61, 62] and Section 3.5.

Chapter 2

Solution of a system of coupled time-dependent Schrödinger equations

The time-dependent Schrödinger equation (TDSE):

$$i\hbar\frac{\partial\psi}{\partial t} = \hat{\mathbf{H}}(t)\,\psi, \tag{2.1}$$

is essential to quantum dynamics. The equation for a system of several coupled time-dependent Schrödinger equations can be written down with explicitly shown all n coupled terms inside ψ and $\hat{\mathbf{H}}(t)$:

$$i\hbar\frac{\partial}{\partial t}\begin{pmatrix}\psi_1\\ \psi_2\\ \vdots\\ \psi_n\end{pmatrix} = \begin{bmatrix}\hat{\mathbf{H}}_1(t) & \hat{\mathbf{V}}_{1,2}(t) & \dots & \hat{\mathbf{V}}_{1,n}(t)\\ \hat{\mathbf{V}}_{2,1}(t) & \hat{\mathbf{H}}_2(t) & \dots & \hat{\mathbf{V}}_{2,n}(t)\\ \vdots & \vdots & \ddots & \vdots\\ \hat{\mathbf{V}}_{n,1}(t) & \hat{\mathbf{V}}_{n,2}(t) & \dots & \hat{\mathbf{H}}_n(t)\end{bmatrix}\begin{pmatrix}\psi_1\\ \psi_2\\ \vdots\\ \psi_n\end{pmatrix} \tag{2.2}$$

Each ψ_n corresponds to a particular wavefunction at n-th electronic potential, while $\hat{\mathbf{V}}_{j,k}$ correspond to the coupling elements between the corresponding levels and $\hat{\mathbf{H}}_i(t)$ is the Hamiltonian at given level. The solution to such system provides us with understanding of fundamental quantum processes. For almost all of these processes closed form solutions do not exist. Instead of these the numerical algorithms are being developed to simulate the quantum processes from first principles. These algorithms come in two flavours: for the time-independent Hamiltonian $\hat{\mathbf{H}}$ and for the time-dependent Hamiltonian $\hat{\mathbf{H}}(t)$.

Hence, this chapter is divided into two main parts. The first part deals with the simpler time-independent Hamiltonian [63]. As noted in comparison [64] this method outperforms all other methods with time-independent Hamiltonian. Indeed thanks to high accuracy this method is commonly used in benchmarks to provide a reference solution. The second part describes a more general algorithm for various classes of problems including time-dependent Hamiltonian, nonlinear problems, non-hermitian problems and problems with an inhomogeneous source term [41]. As mentioned earlier the novelty is both the C++ implementation of these two algorithms as well as the addition to these algorithms the ability to calculate multiple coupled electronic levels.

The main advantage of presented algorithms is the ability to significantly reduce the numerical error in time propagation up to the level of the Unit in the Last Place (ULP)

error[7] of the floating point numerical representation (see Sections 2.3 and 3.5, compare also with [65]). This is made possible by:

(a) Using an exponentially converging series in the expansion of the evolution operator in the case of $\hat{\mathbf{H}}$.

(b) Iteratively applying the Duhamel's principle until required convergence criterion is met for $\hat{\mathbf{H}}(t)$. In this case the error tolerance can also be used to control the calculation speed.

This ability to produce results with error in the range of the numerical ULP error[7] of used precision, together with high precision types `long double` (18 decimal places) and `float128` (33 decimal places) and a fast C++ implementation means that the code presented in this work can be used to obtain reference results in many difficult simulation cases.

2.1 TDSE with the time-independent Hamiltonian

The goal of this section is to numerically solve the time-dependent Schrödinger equation:

$$i\hbar\frac{\partial\psi}{\partial t} = \hat{\mathbf{H}}\psi, \tag{2.3}$$

assuming a time-independent Hamiltonian $\hat{\mathbf{H}} \neq \hat{\mathbf{H}}(t)$, e.g. meaning that the potential cannot change with time, and with some given initial state $\psi_0(t = t_0)$ of the system. In that case the solution, to propagate ψ from time t to time $t + \Delta t$, can be obtained via the evolution operator:

$$\psi(\vec{r}, t + \Delta t) = \hat{\mathbf{U}}(\Delta t)\psi(\vec{r}, t), \tag{2.4}$$

where $\hat{\mathbf{U}}(\Delta t) = e^{-\frac{i\hat{\mathbf{H}}\Delta t}{\hbar}}$ is the evolution operator. To do this, first I will give introduction about numerical precision. Then I will expand e^{-ix} in terms of Chebyshev polynomials (see Appendix A) and finally in I will expand the evolution operator in the Chebyshev basis.

2.1.1 Computer subset of rational numbers $\widetilde{\mathbb{Q}}$

To apply numerically the evolution operator Eq. 2.4 a finite–precision computer will be used. The floating point computer representation of a number uses N significant bits, P exponent bits and 1 bit to store the sign (see Tab. 2.1). For example the 64–bit rational number used in computers called `double` is constructed from bits $b_0 \ldots b_{63} \in \{0, 1\}$. It is a subset $\widetilde{\mathbb{Q}}$ of rational numbers $\mathbb{Q}$ given by[8]:

$$\begin{aligned} \mathbb{Q} \supset \widetilde{\mathbb{Q}} &= \left\{ x \mid x = -1^s \left(1 + \sum_{n=1}^{52} b_{(52-n)}\, 2^{-n} \right) 2^{p-1023} \right\} \\ p &= \sum_{m=0}^{10} b_{(52+m)}\, 2^m, \end{aligned} \tag{2.5}$$

[7] ULP error for a given number x is the smallest distance ε towards the next, larger, number: $x + \varepsilon$.

[8] note that number of significant bits in the Table 2.1 lists also the 1 first implicit bit (always equal to 1 which is present before the $\sum$ sign in Eq. 2.5), hence for example `double` uses 53 significant bits, but the formula Eq. 2.5 performs summation on only 52 bits.

Table 2.1: List of high-precision types.

Type	Total bits	Decimal places	Exponent bits: P	Significant bits: N	Notes
`float`	32	6	8	24	only for testing
`double`	64	15	11	53	hardware accelerated
`long double`†	80	18	15‡	64	hardware accelerated
`float128`§	128	33	15	113	may be h/w accelerated
`mpfr`§♯	N	$N\log_{10}(2)$	—	—	MPFR library
`cpp_bin_float`§♯	N	$N\log_{10}(2)$	—	—	no MPFR dep., slower♭

† The specifics of `long double` depend on the particular compiler and hardware; the values in this table correspond to the most common x86 platform and the `g++` compiler.
‡ All types use 1 bit to store the sign and all types except `long double` have an implicit first bit=1, hence here the sum $15+64\neq 80$.
§ The complete C++ type names for the Boost high-precision types are as follows: `boost::multiprecision::float128`, `boost::multiprecision::mpfr_float_backend` and `boost::multiprecision::cpp_bin_float`.
♯ Arbitrary precision support is currently a work in progress with only the FFT implementation missing (used by the kinetic energy operator), see footnote[6] on page 3 and Section 3.5.
♭ The `cpp_bin_float` is native to C++ Boost library thus can be easily used (no MPFR dependence), but it is slower.

where the sign $s = b_{63}$. In decimal representation there are $53\log_{10}(2) \approx 15.955$ significant decimal digits, the numbers possible within the representation range from 10^{-308} to 10^{308}.

In general any floating point precision type is constructed in a similar manner replacing 52 and 1023 in Eq. 2.5 with $N-1$ and $2^{P-1}-1$ respectively using the values in Tab. 2.1:

$$\begin{aligned}\mathbb{Q} \supset \widetilde{\mathbb{Q}} = \left\{ x \,|\, x = -1^{s} \left(1 + \sum_{n=1}^{N-1} b_{(N-n-1)}\, 2^{-n}\right) 2^{p-(2^{P-1}-1)} \right\} \\ p = \sum_{m=0}^{P-1} b_{(N+m-1)}\, 2^{m}\end{aligned} \tag{2.6}$$

Some values of p are reserved by the standard floating point implementation (called „IEEE 754 Standard" [66]) to mark ∞ or $-\infty$ or „`Not a Number`", but those details will be skipped here.

The density of numbers representable by Eq. 2.6 in the real numbers $\mathbb{R}$ varies with distance from 0, for example between 2^{52} and 2^{53} the representable numbers are exactly the integer numbers only, for the next range $[2^{53}, 2^{54}]$ only every second (even) integers are representable. One should take note that a simple operation of adding 1 to a floating point number may not change this number at all! This is due to the target number being impossible to express in this representation. This is the main reason for frequent surprises when doing numerical computations. The most typical example is that it is not possible to represent the number $\frac{1}{10}$ in this representation. In C++ a simple number `0.1` is interpreted by compiler as being of type `double`, so when it is assigned to `float128` it results in a number `0.1000000000000000055511151231257783` with 33 decimal places[9]. Similar type of error

[9] One has to use the literal suffix `Q` in C++ to denote quad-precision numbers and should write `0.1Q` instead.

(successive multiplication by `0.1` with an accumulating error) was the cause for a Patriot missile failure on 25 February 1991, which resulted in several fatalities [67]. Doing computation in high precision is of prime importance since if more bits were used to represent a number, the explosion of an Ariane 5 rocket launched by the European Space Agency on 4 June 1996 could have been prevented [68, 69, 70, 71] as it was a result of an inappropriate conversion from a 64 bit floating point number into a 16 bit signed integer. Indeed, the 64 bit floating point number was too big to be represented as a 16 bit signed integer. See [65] for a detailed discussion about how to deal with this kind of problems. Another example are chaotic systems, e.g. a triple pendulum behaves differently depending on precision used in the calculation [28].

A more detailed exposition of high precision and quantum dynamics computations is presented in Sections 2.3 and 3.5. We will now return to the time propagation of the time-independent Hamiltonian having in mind the discretization restrictions described here.

2.1.2 Expansion of e^{-ix} using the Chebyshev polynomials basis

Before expressing the evolution operator in terms of the Chebyshev polynomials we will focus our attention on e^{-ix}. Thanks to the spectral theorem the results from this section will be applicable to $e^{-\frac{i\hat{\mathbf{H}}\Delta t}{\hbar}}$. We start with the fact that e^{-ix} can be expanded in an orthogonal polynomial basis:

$$e^{-ix} = \sum_{k=0}^{\infty} a_k T_k(x)\,, \tag{2.7}$$

where Chebyshev polynomials T_k (see Appendix A) form an orthogonal basis only in the range $x \in [-1,1]$. With that in mind, first we will find the a_k coefficients in this range, then we will extend this range to $x \in [x_{\min}, x_{\max}]$. Integrating both sides of Eq. 2.7 by applying the following $\frac{2-\delta_{k,0}}{\pi}\int_{-1}^{1}\frac{T_k(x)\bullet}{\sqrt{1-x^2}}dx$, and putting each side into the $\bullet$ we obtain:

$$\frac{2-\delta_{k,0}}{\pi}\int_{-1}^{1}\frac{T_k(x)e^{-ix}}{\sqrt{1-x^2}}dx = \frac{2-\delta_{k,0}}{\pi}\int_{-1}^{1}\frac{T_k(x)\sum_{k'=0}^{\infty}a_{k'}T_{k'}(x)}{\sqrt{1-x^2}}dx\,. \tag{2.8}$$

By linearity of the integral and orthogonality of Chebyshev polynomials and noting that each a_k is a constant, the term $\sum_{k'=0}^{\infty}a_{k'}$ can be moved outside of the integral, so the right side becomes simply the a_k term like this:

$$\frac{2-\delta_{k,0}}{\pi}\int_{-1}^{1}\frac{T_k(x)\sum_{k'=0}^{\infty}a_{k'}T_{k'}(x)}{\sqrt{1-x^2}}dx = \sum_{k'=0}^{\infty}a_{k'}\left(\frac{2-\delta_{k,0}}{\pi}\int_{-1}^{1}\frac{T_k(x)T_{k'}(x)}{\sqrt{1-x^2}}dx\right) = \sum_{k'=0}^{\infty}a_{k'}\delta_{k,k'} = a_k\,. \tag{2.9}$$

The left hand side, by identity given in Eq. A.5 with $R=-1$ becomes:

$$\frac{2-\delta_{k,0}}{\pi}\int_{-1}^{1}\frac{T_k(x)e^{ixR}}{\sqrt{1-x^2}}dx = \left(2-\delta_{k,0}\right)i^k J_k(R)\,, \tag{2.10}$$

where $J_k(R)$ is the Bessel function. Hence the full expansion of e^{-ix} for $x \in [-1,1]$ is:

$$\begin{aligned} e^{-ix} &= \sum_{k=0}^{\infty} a_k T_k(x) \\ a_k &= \left(2-\delta_{k,0}\right)i^k J_k(-1)\,. \end{aligned} \tag{2.11}$$

Doing the same for $x \in [x_{\min}, x_{\max}]$ adds only one additional requirement that the argument of $T_k(\omega)$ lies in the range $\omega \in [-1,1]$, which can be achieved by linearly scaling the argument x:

$$\omega = 2\frac{x - x_{min}}{x_{max} - x_{min}} - 1, \quad \text{for} \quad x \in [x_{\min}, x_{\max}]. \tag{2.12}$$

The goal is to find the expansion coefficients a_k for e^{-ix} while assuming $x \in [x_{\min}, x_{\max}]$:

$$e^{-ix} = \sum_{k=0}^{\infty} a_k T_k \left(2\frac{x - x_{min}}{x_{max} - x_{min}} - 1 \right). \tag{2.13}$$

Let us rewrite this expression using ω. For this we solve Eq. 2.12 for x with respect to ω:

$$x = \frac{(\omega+1)(x_{max} - x_{min})}{2} + x_{min} = \left(\frac{x_{max} - x_{min}}{2} \right) \omega + \left(\frac{x_{max} - x_{min}}{2} \right) + x_{min} \tag{2.14}$$

to simplify further calculations let us introduce the following definitions:

$$\begin{aligned} R &\stackrel{\text{def}}{=} \frac{x_{max} - x_{min}}{2} \\ G &\stackrel{\text{def}}{=} x_{min}, \end{aligned} \tag{2.15}$$

so that now we can rewrite Eq. 2.14:

$$x = \omega R + R + G, \tag{2.16}$$

and express Eq. 2.13 using ω as the argument:

$$e^{-i(\omega R + R + G)} = \sum_{k=0}^{\infty} a_k T_k(\omega). \tag{2.17}$$

Redoing the calculations in Eq. 2.9 we arrive at solution similar to Eq. 2.11 with an extra $e^{-i(R+G)}$ term in the a_k coefficient:

$$a_k = e^{-i(R+G)} \left(2 - \delta_{k,0}\right) i^k J_k(-R). \tag{2.18}$$

Next, to write Eq. 2.17 in terms of x, first we need to note the following relation from Eq. 2.12:

$$\omega = 2\frac{x - x_{min}}{x_{max} - x_{min}} - 1 = \frac{x - G}{R} - 1 = \frac{x}{R} - \left(1 + \frac{G}{R}\right) \tag{2.19}$$

and finally e^{-ix} in terms of $x \in [x_{\min}, x_{\max}]$ can be written as:

$$\begin{aligned} e^{-ix} &= \sum_{k=0}^{\infty} a_k T_k \left(\frac{x}{R} - \left(1 + \frac{G}{R}\right) \right) \\ a_k &= e^{-i(R+G)} \left(2 - \delta_{k,0}\right) i^k J_k(-R) \\ R &= \frac{x_{max} - x_{min}}{2} \\ G &= x_{min} \end{aligned} \tag{2.20}$$

This is the important result from this section and soon we will apply it to the evolution operator. But first let us observe that the Bessel function $J_k(-R)$ has a very useful property of exponential decay like e^{-k} for $k > |R|$ as shown in Figure 2.1. This means that for numerical purposes only finite number of J_k terms above $k > R$ in the series will be necessary to achieve desired finite numerical precision, because all higher terms will be below computer numerical precision (see Section 2.1.1).

Figure 2.2 shows the approximation of e^{-ix} for $x \in [-10, 50]$ (which means that $R = 30$ and $G = -10$) using only 30 elements in the series. The approximation error is apparent. However when we increase the number of terms in the series to 60 as in Figure 2.3 the error is significantly smaller, which is shown in Figure 2.4. Using even more elements in the series causes the error to decrease exponentially and quickly satisfy the ULP numerical precision limit of the floating point numbers used. This is the main reason why this method is often used for producing reference solutions [64].

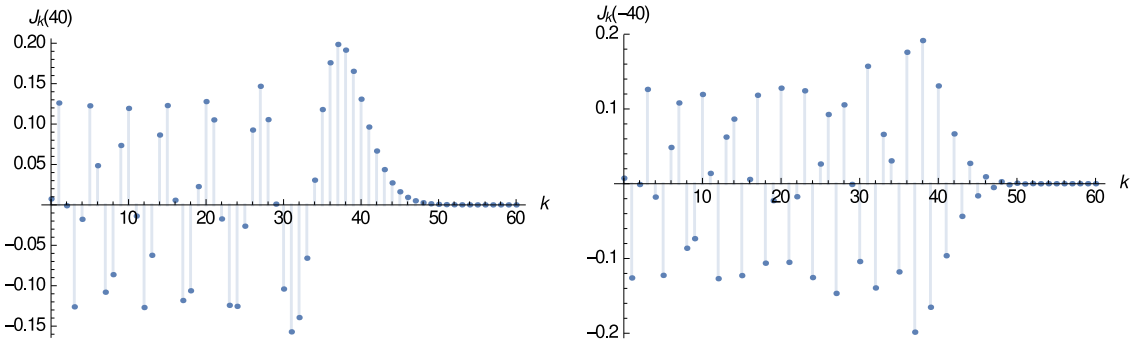


Figure 2.1: Values of Bessel functions $J_k(R)$ for $R = 40$ (left) and $R = -40$ (right) as a function of natural number $k \in \mathbb{N}$. The exponential decay for $J_k(-40)$ is less apparent because $J_k(-x) = (-1)^k J_k(x)$ (see [72] page 361, Eq.9.1.35).

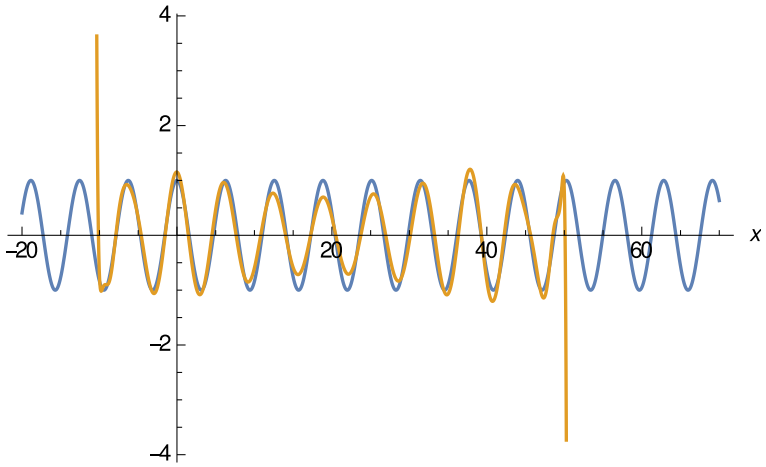


Figure 2.2: Comparison between real parts of e^{-ix} (blue line) and $\sum_{k=0}^{30} a_k T_k\left(\frac{x}{30} - \left(1 + \frac{(-10)}{30}\right)\right)$ (orange line), in the range $x_{min} = -10$ and $x_{max} = 50$. The approximation error is noticeable, outside this range the approximation loses its validity.

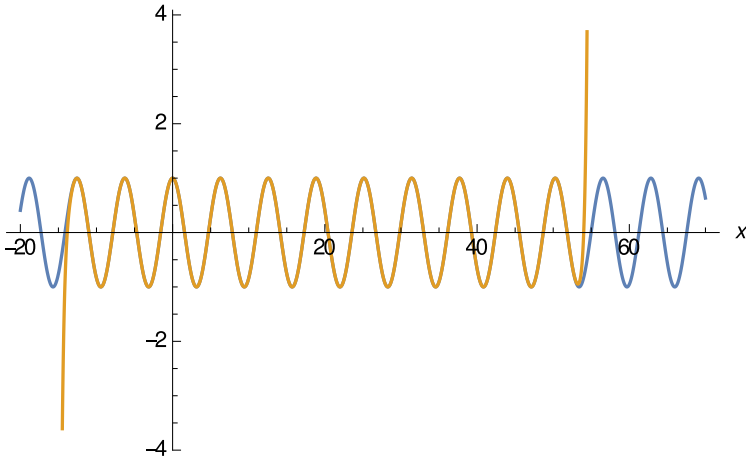


Figure 2.3: The same comparison, but using 60 elements in the series: e^{-ix} (blue line), $e^{-ix} \approx \sum_{k=0}^{60} a_k T_k\left(\frac{x}{30} - \left(1 + \frac{(-10)}{30}\right)\right)$ (orange line).

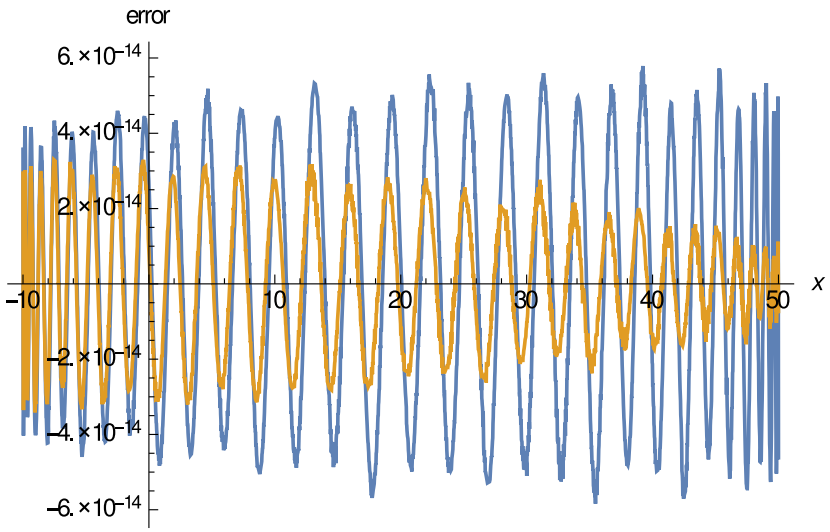


Figure 2.4: Approximation error in the range $x \in [-10, 50]$ calculated as $e^{-ix} - \sum_{k=0}^{60} a_k T_k\left(\frac{x}{30} - \left(1 + \frac{(-10)}{30}\right)\right)$, real (blue line) and imaginary parts (orange line) are plotted separately.

2.1.3 The evolution operator in the Chebyshev basis

The x_{min} and x_{max} from the previous section correspond to the spectral energy range E_{min} and E_{max} of the Hamiltonian. We need to numerically approximate this energy range, despite the fact that the kinetic energy operator is not bounded (see numerical approximation of ∇^2 with Fast Fourier Transform (FFT) in Appendix B). Let us assume Hamiltonian of the form:

$$\hat{\mathbf{H}} = \frac{-\hbar^2}{2m}\nabla^2 + V(\vec{r}). \tag{2.21}$$

To approximate the operation $\hat{\mathbf{U}}(\Delta t)\psi(\vec{r},t)$ the initial wavefunction $\widetilde{\psi}_0(x,t=t_0)$ is discretized on a grid:

$$\widetilde{\psi}_0(x_{min}+n\Delta x, t=t_0) \stackrel{\text{def}}{=} \psi_0(x_{min}+n\Delta x, t=t_0), \quad \text{for} \quad n\in[0,N)\subset\mathbb{N}_0\,, \tag{2.22}$$

where Δx is the grid step and N is the number of points. The approximated $\widetilde{\psi}_0(x,t=t_0)$ covers the wavefunction in the range from x_{min} to $x_{max} = x_{min} + (N-1)\Delta x$. It equals the $\psi_0(x,t=t_0)$ at the grid points, and everywhere else is undefined. Figure 2.5 shows an example wavefunction and its approximation. Similarly the one dimensional potential $V(x)$ will be approximated numerically by $\widetilde{V}(x)$:

$$\widetilde{V}(x_{min}+n\Delta x) \stackrel{\text{def}}{=} V(x_{min}+n\Delta x), \quad \text{for} \quad n\in[0,N)\subset\mathbb{N}_0\,. \tag{2.23}$$

It is useful to select x_{min} and Δx such that the grid points avoid singularities in the potential. I have obtained roughly $10\times$ larger numerical errors in calculations when the Coulomb singularity was lying on the grid point, regardless of how „big" was the floating point number in there (it had to be finite). Put differently, having singularities placed between the grid points reduces the errors tenfold. Thus the discretized potential always lies within the energy range $\widetilde{V}(x) \in [\widetilde{V}_{min}, \widetilde{V}_{max}]$. Figure 2.6 shows an example numerical approximation of Coulomb potential, with singularity placed between the grid points.

In approximation of the momentum operator $-i\hbar\nabla$ we note, that there exists a maximum momentum representable on a grid, based directly on the shortest possible wavelength $\lambda_{min} = 2\Delta x$ (see Figure 2.7), and $k_{max} = \frac{2\pi}{\lambda_{min}}$. Hence maximum momentum:

$$\widetilde{p}_{max} = \hbar k_{max} = \hbar\frac{2\pi}{\lambda_{min}} = \frac{\hbar\pi}{\Delta x}, \tag{2.24}$$

puts a limit on maximum numerically representable kinetic energy:

$$\widetilde{T}_{max} = \frac{p_{max}^2}{2m} = \frac{\hbar^2\pi^2}{2m\Delta x^2} \tag{2.25}$$

With these numerical limitations in place we can pinpoint the spectral energy range of the numerically approximated Hamiltonian $\widetilde{\mathbf{H}}$:

$$\widetilde{E} \in \sigma(\widetilde{\mathbf{H}}) \stackrel{\text{def}}{=} \widetilde{\mathbb{Q}} \cap [\widetilde{V}_{min}, \widetilde{V}_{max} + \widetilde{T}_{max}], \tag{2.26}$$

where $\sigma(\widetilde{\mathbf{H}})$ denotes the spectrum of $\widetilde{\mathbf{H}}$, $\widetilde{\mathbb{Q}}$ is the subset of rational numbers used by computer (Section 2.1.1) and $\widetilde{V}_{min}$, $\widetilde{V}_{max}$, $\widetilde{T}_{max}$ are given in the preceding paragraph. Next, we split the Hamiltonian into discrete part which can be handled by the computer, and the unbounded continuum part:

$$\hat{\mathbf{H}} = (\widetilde{\mathbf{H}})_{\sigma(\widetilde{\mathbf{H}})} + (\hat{\mathbf{H}})_{\sigma(\hat{\mathbf{H}})\setminus\sigma(\widetilde{\mathbf{H}})}, \tag{2.27}$$

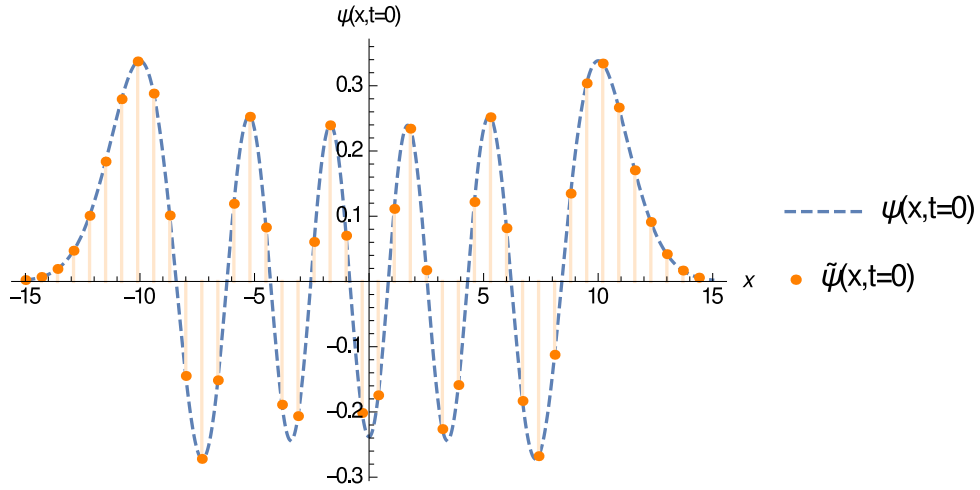


Figure 2.5: Example wavefunction $\psi_0(x,t=t_0)$, the harmonic oscillator eigenfunction at 10^{th} excitation level (real part is plotted, imaginary part is zero) compared with its discretization $\widetilde{\psi}_0(x,t=t_0)$.

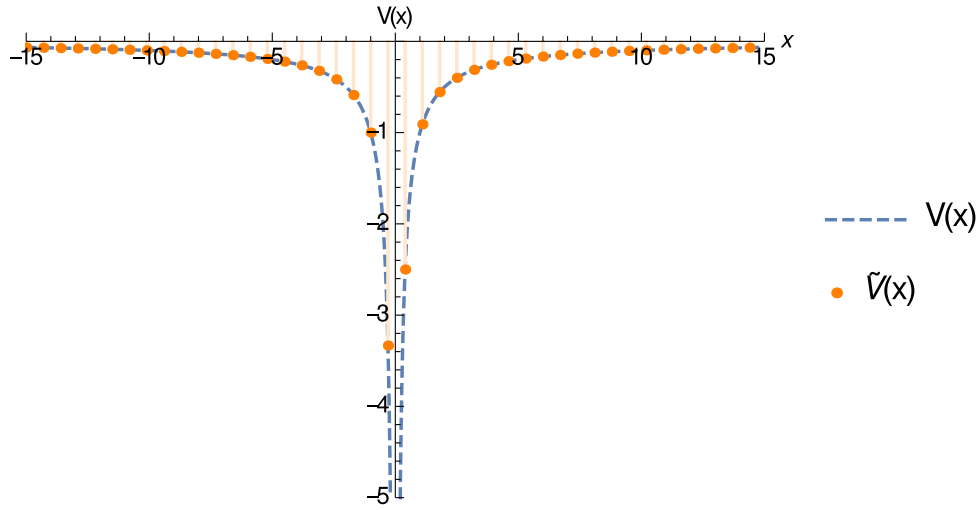


Figure 2.6: Example Coulomb potential $V(x) = -\dfrac{Ze^2}{4\pi\varepsilon_0 r}$ compared with its discretization $\widetilde{V}(x)$.

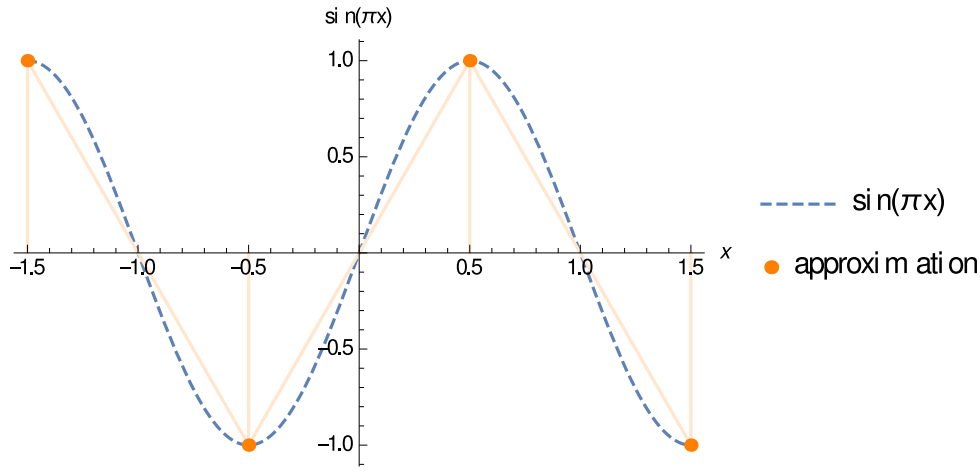


Figure 2.7: The shortest wavelength possible $\lambda_{min} = 2\Delta x$ on a grid with spacing $\Delta x = 1$.

where the subscript $\sigma(\bullet)$ denotes the spectrum range of each operator. The numerical approximation of TDSE will have following form:

$$i\hbar\frac{\partial\widetilde{\psi}}{\partial t}=\widetilde{\mathbf{H}}\widetilde{\psi}, \tag{2.28}$$

the other part involving $(\hat{\mathbf{H}})_{\sigma(\hat{\mathbf{H}})\setminus\sigma(\widetilde{\mathbf{H}})}$ cannot be solved numerically. The numerically approximated Hamiltonian has the form:

$$\widetilde{\mathbf{H}}=\frac{-\hbar^2}{2m}\widetilde{\nabla}^2+\widetilde{V}(\vec{r}), \tag{2.29}$$

The solution in one dimensional case is represented by the numerical approximation of the evolution operator:

$$\widetilde{\psi}(x,t+\Delta t)=e^{-\frac{i\widetilde{\mathbf{H}}\Delta t}{\hbar}}\widetilde{\psi}(x,t)=\widetilde{\mathbf{U}}(\Delta t)\widetilde{\psi}(x,t). \tag{2.30}$$

Similarly as in section 2.1.2 (on page 8) we will express $\widetilde{\mathbf{H}}$ in Chebyshev basis by introducing the normalized numerical Hamiltonian $\widetilde{\mathbf{H}}_{norm}$ [10]:

$$\begin{aligned}\widetilde{\mathbf{H}}_{norm}&=2\frac{\widetilde{\mathbf{H}}-\widetilde{E}_{min}}{\widetilde{E}_{max}-\widetilde{E}_{min}}-1=\frac{\widetilde{\mathbf{H}}\Delta t}{\hbar R}-\left(1+\frac{G}{R}\right)\\ \frac{\widetilde{\mathbf{H}}\Delta t}{\hbar}&=R\widetilde{\mathbf{H}}_{norm}+R+G\end{aligned} \tag{2.31}$$

with following definitions for R and G:

$$\begin{aligned}R&\stackrel{\text{def}}{=}\tfrac{\Delta t}{2\hbar}\left(\widetilde{E}_{max}-\widetilde{E}_{min}\right)\\ G&\stackrel{\text{def}}{=}\tfrac{\Delta t}{\hbar}\widetilde{E}_{min},\end{aligned} \tag{2.32}$$

and the current goal is to find the a_k coefficients in the following equation, expressed either in terms of $\widetilde{\mathbf{H}}$:

$$e^{\frac{-i\widetilde{\mathbf{H}}\Delta t}{\hbar}}=\sum_{k=0}^{\infty}a_k T_k\left(\frac{\Delta t\widetilde{\mathbf{H}}}{\hbar R}-\left(1+\frac{G}{R}\right)\right), \tag{2.33}$$

or in terms of $\widetilde{\mathbf{H}}_{norm}$ (please note that a_k coefficients have the same value in both equations):

$$e^{-i\left(R\widetilde{\mathbf{H}}_{norm}+R+G\right)}=\sum_{k=0}^{\infty}a_k T_k\left(\widetilde{\mathbf{H}}_{norm}\right). \tag{2.34}$$

Proceeding in the same manner as before (on page 8) we find that:

$$e^{-i(R+G)}\left(\frac{2-\delta_{k,0}}{\pi}\int_{-1}^{1}\frac{T_k(\lambda)e^{-iR\lambda}}{\sqrt{1-\lambda^2}}d\lambda\right)=\frac{2-\delta_{k,0}}{\pi}\int_{-1}^{1}\frac{T_k(\lambda)\sum_{k'=0}^{\infty}a_{k'}T_{k'}(\lambda)}{\sqrt{1-\lambda^2}}d\lambda \tag{2.35}$$

[10] The $\widetilde{E}_{max}$ and $\widetilde{E}_{min}$ are the energy range of the Hamiltonian. In Section 2.2.6 I will describe the Arnoldi approach in which knowledge of the energy spectrum range is not necessary (see appendix B.2 in [41])

Putting everything together the numerical approximation of the evolution operator in the Chebyshev basis has the following form:

$$\begin{aligned} e^{\frac{-i\widetilde{\mathbf{H}}\Delta t}{\hbar}} &= \sum_{k=0}^{\infty} a_k T_k\left(\frac{\Delta t\widetilde{\mathbf{H}}}{\hbar R} - \left(1+\frac{G}{R}\right)\right) \\ a_k &= e^{-i(R+G)}\left(2-\delta_{k,0}\right) i^k J_k(-R) \\ R &= \frac{\Delta t}{2\hbar}\left(\widetilde{E}_{max} - \widetilde{E}_{min}\right) \\ G &= \frac{\Delta t}{\hbar}\widetilde{E}_{min}. \end{aligned} \tag{2.36}$$

Now it is possible to find the numerical approximation of the wavefunction in the next timestep:

$$\widetilde{\psi}(x,t+\Delta t) = e^{-\frac{i\widetilde{\mathbf{H}}\Delta t}{\hbar}}\widetilde{\psi}(x,t).$$

This is done by applying the Chebyshev recurrence relation given in Eq. A.2 which says that:

$$\begin{cases} T_0(\widetilde{\mathbf{H}}_{norm}) = 1 \\ T_1(\widetilde{\mathbf{H}}_{norm}) = \widetilde{\mathbf{H}}_{norm} \\ T_k(\widetilde{\mathbf{H}}_{norm}) = 2\widetilde{\mathbf{H}}_{norm}T_{k-1}(\widetilde{\mathbf{H}}_{norm}) - T_{k-2}(\widetilde{\mathbf{H}}_{norm}). \end{cases} \tag{2.37}$$

We can calculate the next element in the series by using the two previous elements of the series, after the Hamiltonian of each previous element of the series has acted on the wavefunction:

$$\begin{aligned} \widetilde{\psi}(\vec{r},t+\Delta t) &= \sum_{k=0}^{\infty} a_k T_k(\widetilde{\mathbf{H}}_{norm})\widetilde{\psi}(\vec{r},t) \\ &= a_0 T_0(\widetilde{\mathbf{H}}_{norm})\widetilde{\psi}(\vec{r},t) + a_1 T_1(\widetilde{\mathbf{H}}_{norm})\widetilde{\psi}(\vec{r},t) + a_2 T_2(\widetilde{\mathbf{H}}_{norm})\widetilde{\psi}(\vec{r},t) + \dots \\ &= a_0 \underbrace{1\widetilde{\psi}(\vec{r},t)}_{=T_0\widetilde{\psi}(\vec{r},t)} + a_1 \underbrace{\widetilde{\mathbf{H}}_{norm}\widetilde{\psi}(\vec{r},t)}_{=T_1\widetilde{\psi}(\vec{r},t)} + a_2 \underbrace{\left(2\widetilde{\mathbf{H}}_{norm}T_1\widetilde{\psi}(\vec{r},t) - T_0\widetilde{\psi}(\vec{r},t)\right)}_{=T_2\widetilde{\psi}(\vec{r},t)} + \dots \end{aligned} \tag{2.38}$$

or more explicitly, without using $\widetilde{\mathbf{H}}_{norm}$:

$$\begin{aligned} \widetilde{\psi}(\vec{r},t+\Delta t) &= a_0 \underbrace{1\widetilde{\psi}(\vec{r},t)}_{=T_0\widetilde{\psi}(\vec{r},t)} \\ &+ a_1 \underbrace{\left(\frac{\Delta t}{\hbar R}\widetilde{\mathbf{H}}\widetilde{\psi}(\vec{r},t) - \left(1+\frac{G}{R}\right)\widetilde{\psi}(\vec{r},t)\right)}_{=T_1\widetilde{\psi}(\vec{r},t)} \\ &+ a_2 \underbrace{\left(2\left(\frac{\Delta t}{\hbar R}\widetilde{\mathbf{H}}T_1\widetilde{\psi}(\vec{r},t) - \left(1+\frac{G}{R}\right)T_1\widetilde{\psi}(\vec{r},t)\right) - T_0\widetilde{\psi}(\vec{r},t)\right)}_{=T_2\widetilde{\psi}(\vec{r},t)} + \dots \end{aligned} \tag{2.39}$$

In above formula the application of momentum operator on $\widetilde{\psi}$ is calculated as described in Appendix B. Only two previous elements of the series have to be kept in computer memory: $T_{k-1}\psi(\vec{r},t)$ and $T_{k-2}\psi(\vec{r},t)$, which helps significantly for larger systems. The coefficient $e^{-i(R+G)}$ from a_k is put outside the series and multiplication by it is performed only once.

2.1.4 A note on accuracy, grid size Δx and momentum operator cutoff

As mentioned in section 2.1.2 the Bessel function $J_k(-R)$ decreases exponentially like e^{-k} for $k > |R|$ (see Figure 2.1), but here the value of R directly depends on timestep Δt. This means that timestep can be arbitrarily large and still the error in the solution can be arbitrarily small, provided enough components of the series are calculated, so that the magnitude of a_k is smaller than numerical precision $\widetilde{\mathbb{Q}}$. This is computationally very efficient thanks to this property of Bessel function $J_k(R)$. Time propagation methods which can have arbitrarily large Δt are called global methods. This eliminates only the numerical error in $\widetilde{\mathbf{U}}$, but it cannot eliminate the error introduced by replacing $\hat{\mathbf{U}}$ with $\widetilde{\mathbf{U}}$. It is the task of the user of this method to pick appropriate energy cutoff $\widetilde{E}_{max}$ and $\widetilde{E}_{min}$, so that the effects of neglecting part of the energy spectrum have negligible effect on results. The most important part of this energy cutoff is estimating grid step Δx which directly influences maximum representable kinetic energy $\widetilde{T}_{max}$. Kosloff [73] suggests some estimates based on virial theorem: $T_{max} = \frac{n}{2} V_{max}$ where n is the power in which spatial coordinate appears in potential description. So for example for harmonic oscillator the grid size should be:

$$\Delta x \leq \sqrt{\frac{h}{m\omega N}} \tag{2.40}$$

and in Coulomb potential it should be:

$$\Delta x \leq \frac{\pi h \varepsilon_0}{2 m q_1 q_2}. \tag{2.41}$$

These considerations will become unnecessary in the Section 2.2.6. The estimations of the spectral range of the operator will be not needed by the Arnoldi algorithm used in the time-dependent Hamiltonian.

2.1.5 Time evolution for coupled electronic states

Let us now focus on numerical approximation of TDSE with a Hamiltonian for several coupled electronic states.

$$i\hbar \frac{\partial}{\partial t} \begin{pmatrix} \psi_1 \\ \psi_2 \\ \vdots \\ \psi_n \end{pmatrix} = \begin{bmatrix} \widetilde{\mathbf{H}}_1 & \widetilde{\mathbf{V}}_{1,2} & \dots & \widetilde{\mathbf{V}}_{1,n} \\ \widetilde{\mathbf{V}}_{2,1} & \widetilde{\mathbf{H}}_2 & \dots & \widetilde{\mathbf{V}}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \widetilde{\mathbf{V}}_{n,1} & \widetilde{\mathbf{V}}_{n,2} & \dots & \widetilde{\mathbf{H}}_n \end{bmatrix} \begin{pmatrix} \psi_1 \\ \psi_2 \\ \vdots \\ \psi_n \end{pmatrix} \tag{2.42}$$

The evolution operator is very similar to Eq. 2.33:

$$e^{\left(\frac{-i\Delta t}{\hbar} \begin{bmatrix} \widetilde{\mathbf{H}}_1 & \widetilde{\mathbf{V}}_{1,2} & \dots & \widetilde{\mathbf{V}}_{1,n} \\ \widetilde{\mathbf{V}}_{2,1} & \widetilde{\mathbf{H}}_2 & \dots & \widetilde{\mathbf{V}}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \widetilde{\mathbf{V}}_{n,1} & \widetilde{\mathbf{V}}_{n,2} & \dots & \widetilde{\mathbf{H}}_n \end{bmatrix} \right)} = \sum_{k=0}^{\infty} a_k T_k \left(\frac{\Delta t}{\hbar R} \begin{bmatrix} \widetilde{\mathbf{H}}_1 & \widetilde{\mathbf{V}}_{1,2} & \dots & \widetilde{\mathbf{V}}_{1,n} \\ \widetilde{\mathbf{V}}_{2,1} & \widetilde{\mathbf{H}}_2 & \dots & \widetilde{\mathbf{V}}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \widetilde{\mathbf{V}}_{n,1} & \widetilde{\mathbf{V}}_{n,2} & \dots & \widetilde{\mathbf{H}}_n \end{bmatrix} - \left(1 + \frac{G}{R}\right) \right) \tag{2.43}$$

Then a direct application of Eq. 2.39 yields:

$$\begin{pmatrix} \widetilde{\psi}(\vec{r},t+\Delta t)_1 \\ \widetilde{\psi}(\vec{r},t+\Delta t)_2 \\ \vdots \\ \widetilde{\psi}(\vec{r},t+\Delta t)_n \end{pmatrix} = a_0 \quad \underbrace{1 \begin{pmatrix} \widetilde{\psi}(\vec{r},t)_1 \\ \widetilde{\psi}(\vec{r},t)_2 \\ \vdots \\ \widetilde{\psi}(\vec{r},t)_n \end{pmatrix}}_{=T_0\widetilde{\psi}(\vec{r},t) \overset{\text{def}}{=} \begin{pmatrix} T_{\widetilde{\psi}0,1} \\ T_{\widetilde{\psi}0,2} \\ \vdots \\ T_{\widetilde{\psi}0,n} \end{pmatrix}}$$

$$+a_1 \underbrace{\left(\frac{\Delta t}{\hbar R} \begin{bmatrix} \widetilde{\mathbf{H}}_1 & \widetilde{\mathbf{V}}_{1,2} & \dots & \widetilde{\mathbf{V}}_{1,n} \\ \widetilde{\mathbf{V}}_{2,1} & \widetilde{\mathbf{H}}_2 & \dots & \widetilde{\mathbf{V}}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \widetilde{\mathbf{V}}_{n,1} & \widetilde{\mathbf{V}}_{n,2} & \dots & \widetilde{\mathbf{H}}_n \end{bmatrix} \begin{pmatrix} \widetilde{\psi}(\vec{r},t)_1 \\ \widetilde{\psi}(\vec{r},t)_2 \\ \vdots \\ \widetilde{\psi}(\vec{r},t)_n \end{pmatrix} - \left(1+\frac{G}{R}\right) \begin{pmatrix} \widetilde{\psi}(\vec{r},t)_1 \\ \widetilde{\psi}(\vec{r},t)_2 \\ \vdots \\ \widetilde{\psi}(\vec{r},t)_n \end{pmatrix} \right)}_{=T_1\widetilde{\psi}(\vec{r},t) \overset{\text{def}}{=} \begin{pmatrix} T_{\widetilde{\psi}1,1} \\ T_{\widetilde{\psi}1,2} \\ \vdots \\ T_{\widetilde{\psi}1,n} \end{pmatrix}}$$

$$+a_2 \underbrace{\left(2 \left(\frac{\Delta t}{\hbar R} \begin{bmatrix} \widetilde{\mathbf{H}}_1 & \widetilde{\mathbf{V}}_{1,2} & \dots & \widetilde{\mathbf{V}}_{1,n} \\ \widetilde{\mathbf{V}}_{2,1} & \widetilde{\mathbf{H}}_2 & \dots & \widetilde{\mathbf{V}}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \widetilde{\mathbf{V}}_{n,1} & \widetilde{\mathbf{V}}_{n,2} & \dots & \widetilde{\mathbf{H}}_n \end{bmatrix} \begin{pmatrix} T_{\widetilde{\psi}1,1} \\ T_{\widetilde{\psi}1,2} \\ \vdots \\ T_{\widetilde{\psi}1,n} \end{pmatrix} - \left(1+\frac{G}{R}\right) \begin{pmatrix} T_{\widetilde{\psi}1,1} \\ T_{\widetilde{\psi}1,2} \\ \vdots \\ T_{\widetilde{\psi}1,n} \end{pmatrix} \right) - \begin{pmatrix} T_{\widetilde{\psi}0,1} \\ T_{\widetilde{\psi}0,2} \\ \vdots \\ T_{\widetilde{\psi}0,2} \end{pmatrix} \right)}_{=T_2\widetilde{\psi}(\vec{r},t) \overset{\text{def}}{=} \begin{pmatrix} T_{\widetilde{\psi}2,1} \\ T_{\widetilde{\psi}2,2} \\ \vdots \\ T_{\widetilde{\psi}2,n} \end{pmatrix}}$$

$$+\dots \tag{2.44}$$

Each row can be called a separate channel, and the j–th channel satisfies:

$$\begin{aligned} \widetilde{\psi}(\vec{r},t+\Delta t)_j \quad &= a_0 \underbrace{1\widetilde{\psi}(\vec{r},t)_j}_{=T_{\widetilde{\psi}0,j}} \\ &+a_1 \underbrace{\left(\frac{\Delta t}{\hbar R} \left(\widetilde{\mathbf{H}}_j \widetilde{\psi}(\vec{r},t)_j + \sum_{k\neq j} \widetilde{V}_{jk} \widetilde{\psi}(\vec{r},t)_k \right) - \left(1+\frac{G}{R}\right) \widetilde{\psi}(\vec{r},t)_j \right)}_{=T_{\widetilde{\psi}1,j}} \\ &+a_2 \underbrace{\left(2 \left(\frac{\Delta t}{\hbar R} \left(\widetilde{\mathbf{H}}_j T_{\widetilde{\psi}1,j} + \sum_{k\neq j} \widetilde{V}_{jk} T_{\widetilde{\psi}1,k} \right) - \left(1+\frac{G}{R}\right) T_{\widetilde{\psi}1,j} \right) - T_{\widetilde{\psi}0,j} \right)}_{=T_{\widetilde{\psi}2,j}} + \dots \end{aligned} \tag{2.45}$$

Thus the calculations of several coupled electronic states can be performed with additional computation cost proportional to the number of these states.

2.1.6 Numerical damping using Chebyshev recurrence

The numerical damping follows the idea from [74, 75] by modifying the recursion relation of the Chebyshev polynomials to add a damping term $e^{-\gamma(R)}$. The new polynomials Q_k use the following definition (cf. Eq. 3.1 in [75]), similar to Eq. 2.37:

$$\begin{cases} Q_0(\widetilde{\mathbf{H}}_{norm}) = 1 \\ Q_1(\widetilde{\mathbf{H}}_{norm}) = e^{-\gamma(R)}\widetilde{\mathbf{H}}_{norm} \\ Q_k(\widetilde{\mathbf{H}}_{norm}) = e^{-\gamma(R)}\left(2\widetilde{\mathbf{H}}_{norm}Q_{k-1}(\widetilde{\mathbf{H}}_{norm}) - e^{-\gamma(R)}Q_{k-2}(\widetilde{\mathbf{H}}_{norm})\right), \end{cases} \tag{2.46}$$

where the R is the wavefunction coordinate in position representation and $e^{-\gamma(R)}$ is a soft rectangular function: equal to 1 on the inside of the grid and decaying exponentially to 0 at the grid boundaries. The position R_{max} at which the decaying starts is the so-called position at which the detectors are positioned. The remaining part of $R > R_{max}$ is the damping band and similarly for $R < R_{min}$. The soft damping part $\gamma(R)$ is constructed using the following formula:

$$\gamma(R) = A\left(\frac{1}{1-\xi(R)} - 1\right) \tag{2.47}$$

$$\xi(R) = \begin{cases} \left(1 - \dfrac{R}{R_{min}}\right)^2 & \text{for} \quad R \le R_{min} \\ \left(\dfrac{R - R_{max}}{L - R_{max}}\right)^2 & \text{for} \quad R \ge R_{max} \\ 0 & \text{for} \quad R_{min} < R < R_{max} \end{cases} \tag{2.48}$$

An example of such function is shown on Figure 2.8 using following parameters: $A = 3$ and the band margins at $R_{min} = 20\,a_0$, $R_{max} = 80\,a_0$ using the total width $L = 100\,a_0$.

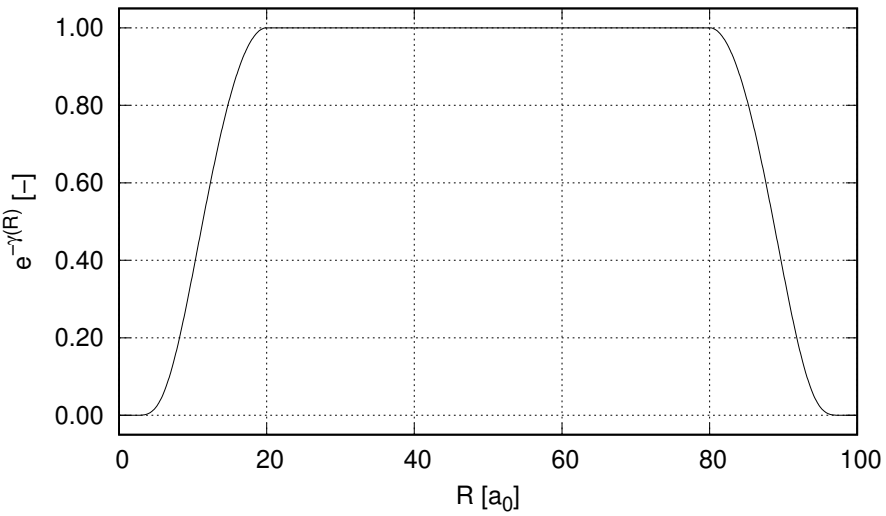


Figure 2.8: Example soft rectangular function used for damping.

2.2 TDSE with the time-dependent Hamiltonian

The time-dependent Hamiltonian is useful for ultrafast spectroscopy, high harmonic generation or coherent control problems. In some cases the Hamiltonian may become nonlinear depending explicitly on the state $\psi(t)$. Such case occurs in mean field approximation, in Gross–Pitaevskii approximation [76], time-dependent Hartree [46, 47, 48, 49] and time-dependent DFT [42, 43, 44, 45]. Another complication may arise when adding a source term to the Schrödinger equation, such as in scattering problems [77]. All these cases will be possible to calculate with the *semi-global* [41, 78] method described in this section.

The global scheme[11] described in the previous section assumes the knowledge of the eigenvalue domain of the Hamiltonian (i.e. the E_{min} and E_{max}). Usually such knowledge is missing, especially for time-dependent or non-Hermitian problems. To overcome this difficulty the method below is implemented with the Arnoldi approach. The main advantage of this approach is that the algorithm determines the energy range while constructing the Krylov space.

Several following sections summarize the detailed derivations presented in [41] and thus are not a novelty in this dissertation, however they are a crucial part to the final, novel, Section 2.2.7 where the method discussed here is extended to several coupled time-dependent Schrödinger equations (such as Eq. 2.2). This final extension and its novel high precision (up to 33 decimal digits) implementation in C++ allows to simulate complex multi level diatomic molecular systems.

In other, less sophisticated, algorithms the method to overcome the Hamiltonian time dependence is to use a small timestep and assume constant Hamiltonian in the single step. This becomes equivalent to method being first order in time and there occurs a loss of precision which was gained by using a higher order method. More sophisticated methods such as Magnus expansion [79] or high order splitting [80] do not assume stationary Hamiltonian, but these methods are still local methods (they require relatively small Δt) with a limited radius of convergence.

The method presented in this section is a semi-global method, because it combines the elements of local propagation and global propagation methods [78]. A fully global time-dependent method was also developed, but it turned out to be too expensive computationally [81]. The semi-global algorithm described in this section is efficient with respect to accuracy compared to numerical effort.

2.2.1 Establishing notation

The time-dependent Schrödinger equation Eq. 2.1 is rewritten in a discretized form similar to Eq. 2.28:

$$\frac{\partial \vec{\widetilde{\psi}}(t)}{\partial t} = -\frac{i}{\hbar}\,\widetilde{\mathbf{H}}(t)\vec{\widetilde{\psi}}(t), \tag{2.49}$$

the time derivative of a discrete state vector $\vec{\widetilde{\psi}}$ of finite size[12] is equal to a matrix (with

[11] „global" refers to the algorithm being independent to the size of the timestep, contrary to typical Taylor based methods where the power n in Δt^n refers to the order of the method. It treats „globally" the entire time span of the calculation.

[12] in the notation of $\vec{\widetilde{\psi}}$ the tilde $\widetilde{\bullet}$ indicates that it is discretized, while vector $\vec{\bullet}$ indicates that it is a state vector.

time dependence) operating on the same vector. Let us introduce a more general version of Eq. 2.49 by adding a source term $\vec{\vec{\mathbf{s}}}$ and by adding a dependence of $\widetilde{\mathbf{H}}$ on $\vec{\vec{\psi}}(t)$:

$$\frac{\partial \vec{\vec{\psi}}(t)}{\partial t} = -\frac{i}{\hbar}\,\widetilde{\mathbf{H}}(\vec{\vec{\psi}}(t),t)\,\vec{\vec{\psi}}(t) + \vec{\vec{\mathbf{s}}}(t). \tag{2.50}$$

This is in fact a general set of ordinary differential equations (ODE). For convenience let us define:

$$\widetilde{\mathbf{G}}(\vec{\vec{\psi}}(t),t) \stackrel{\text{def}}{=} -\frac{i}{\hbar}\,\widetilde{\mathbf{H}}(\vec{\vec{\psi}}(t),t), \tag{2.51}$$

by incorporating the imaginary unit and Planck's constant into $\widetilde{\mathbf{G}}(\vec{\vec{\psi}}(t),t)$. Thus we obtain the following set of ODEs to solve:

$$\frac{\partial \vec{\vec{\psi}}(t)}{\partial t} = \widetilde{\mathbf{G}}(\vec{\vec{\psi}}(t),t)\,\vec{\vec{\psi}}(t) + \vec{\vec{\mathbf{s}}}(t), \tag{2.52}$$

and given the initial condition $\vec{\vec{\psi}}_0 \stackrel{\text{def}}{=} \vec{\vec{\psi}}(t=0)$ the method described in this section will allow propagation of the discrete state vector to the next timestep $\vec{\vec{\psi}}(t+\Delta t)$.

2.2.2 Short summary for time-independent Hamiltonian

Since some general insights from Section 2.1 might have been lost in the details, we shall emphasize the main perk of all global time propagation methods: that the evolution operator is expressed as a function of a matrix Hamiltonian expanded over the energy spectrum using orthogonal polynomial basis, thus allowing arbitrarily large timestep. Let us for a brief moment consider again a simpler case without the time dependence and without the source term:

$$\frac{\partial \vec{\vec{\psi}}(t)}{\partial t} = \widetilde{\mathbf{G}}_0\,\vec{\vec{\psi}}(t), \tag{2.53}$$

where $\widetilde{\mathbf{G}}_0 \stackrel{\text{def}}{=} \widetilde{\mathbf{G}}(t=0)$ loses time dependence. The evolution operator is then expressed as an expansion in a truncated (to K number of terms, see Tab. 2.2 on page 26 for a summary of parameters in this method) polynomial series, similarly to Eq. 2.33. Hence the function $f(x) = e^{x\Delta t}$ (Δt is a parameter) is approximated as:

$$f(x) \approx \sum_{n=0}^{K-1} a_n P_n(x), \tag{2.54}$$

where $P_n(x)$ is a polynomial of degree n (e.g. a Chebyshev polynomial, like in Eq. 2.33, but other polynomials can also be used here) and a_n is the expansion coefficient. Then applying this evolution operator:

$$\vec{\vec{\psi}}(t+\Delta t) = e^{-\frac{i\widetilde{\mathbf{H}}\Delta t}{\hbar}}\,\vec{\vec{\psi}} \approx \sum_{n=0}^{K-1} a_n P_n(\widetilde{\mathbf{G}}_0)\,\vec{\vec{\psi}}(t), \tag{2.55}$$

we obtain the solution at time $t+\Delta t$. The emphasis here lying on the „global" property of the method: the error does not depend on the timestep Δt. The solution is obtained directly at the final time $t+\Delta t$, which can be arbitrarily large. We shall note however that the expansion Eq. 2.55 has to be accurate in the eigenvalue domain of $\widetilde{\mathbf{G}}_0$.

We might consider the following polynomials P_n:

(a) Use the Taylor polynomials $P_n(x) = x^n$ and expand the evolution operator in a Taylor series (a common approach in the local time integration methods), but it is a poor choice: they are not orthogonal. To the contrary: as n increases they are getting more and more parallel in the function space.

(b) Use the Chebyshev polynomials $P_n(x) = T_n(x)$. The fact that they are orthogonal to each other provides two useful properties: (1) the series converges fast and (2) the expansion coefficients a_n are given by a scalar product of the $P_n(x)$ with $f(x)$ (such as Eq. 2.9). This approach is used in Section 2.1.

(c) Another possible approach is to use a Newton interpolation polynomial at the Chebyshev points of the eigenvalue domain. This method will be summarized in Section 2.2.5 and it is used in the currently presented semi-global approach.

We shall note that in Eq. 2.55, we obtain the solution only at the chosen time $t + \Delta t$. It is desirable to follow the evolution of the physical process at a smaller timestep, so that the time dependence of the Hamiltonian can be more accurately captured. It is possible to obtain these intermediate time points $\Delta t_j \in [0, \Delta t)$ via negligible additional cost: using the same matrix vector operations $P_n(\widetilde{\mathbf{G}}_0)$ (Hamiltonian acting on the wavefunction) but with different precomputed scalar coefficients $a_{n,j}$ (where j corresponds to an intermediate time point in the evolution):

$$\vec{\tilde{\psi}}(t + \Delta t_j) = e^{-\frac{i\widetilde{\mathbf{H}}\Delta t_j}{\hbar}} \vec{\tilde{\psi}}_0 \approx \sum_{n=0}^{K-1} a_{n,j} P_n(\widetilde{\mathbf{G}}_0) \vec{\tilde{\psi}}_0 \qquad j = 1, \ldots, M, \tag{2.56}$$

where M is the number of additional intermediate time points in the solution. It is possible, with low computation cost, because expansion of function in the P_n basis has different coefficients $a_{n,j}$ but has the same $P_n(\widetilde{\mathbf{G}}_0)\vec{\tilde{\psi}}_0$ evaluations.

The Eq. 2.56 is in fact an expression for the evolution operator acting on the wavefunction, which for the purpose of the next section we will denote as:

$$\widetilde{\mathbf{U}}_0(\Delta t_j) \stackrel{\text{def}}{=} e^{\widetilde{\mathbf{G}}_0 \Delta t_j}. \tag{2.57}$$

Hence Eq. 2.56 can be also written as:

$$\vec{\tilde{\psi}}(t + \Delta t_j) = \widetilde{\mathbf{U}}_0(\Delta t_j) \vec{\tilde{\psi}}(t). \tag{2.58}$$

2.2.3 Source term with time dependence

On our way towards the full time dependence in Eq. 2.52, we will now add the source term with time dependence to Eq. 2.53:

$$\frac{\partial \vec{\tilde{\psi}}(t)}{\partial t} = \widetilde{\mathbf{G}}_0 \vec{\tilde{\psi}}(t) + \vec{\tilde{\mathbf{s}}}(t). \tag{2.59}$$

We can integrate this equation using Duhamel principle, which provides a way to go from the solution (Eq. 2.56) of the homogeneous Eq. 2.53 to the solution of the inhomogeneous Eq. 2.59 like this:

$$\begin{aligned}\vec{\tilde{\psi}}(t) &= \widetilde{\mathbf{U}}_0(t)\vec{\tilde{\psi}}_0 + \int_0^t \widetilde{\mathbf{U}}_0(t-\tau)\vec{\tilde{\mathbf{s}}}(\tau)d\tau \\ &= e^{\widetilde{\mathbf{G}}_0 t}\vec{\tilde{\psi}}_0 + \int_0^t e^{\widetilde{\mathbf{G}}_0 t}e^{-\widetilde{\mathbf{G}}_0 \tau}\vec{\tilde{\mathbf{s}}}(\tau)d\tau \\ &= e^{\widetilde{\mathbf{G}}_0 t}\vec{\tilde{\psi}}_0 + e^{\widetilde{\mathbf{G}}_0 t}\int_0^t e^{-\widetilde{\mathbf{G}}_0 \tau}\vec{\tilde{\mathbf{s}}}(\tau)d\tau\end{aligned} \tag{2.60}$$

Above we took the advantage of being able to extract from the Duhamel's integral the $\widetilde{\mathbf{U}}_0(t)$ part. So the only part which needs to be solved explicitly is the remaining integral. To do this we will assume that $\vec{\tilde{\mathbf{s}}}(\tau)$ (after discretization: $\vec{\tilde{\mathbf{s}}}(\Delta t_j)$) can be expressed as a polynomial function of time. It is a bit of a simplification, but later on we will be able to decide how many elements M (see Tab. 2.2) in the series the algorithm will use, thus being able to directly control the accuracy of the solution:

$$\vec{\tilde{\mathbf{s}}}(t) = \sum_{m=0}^{M-1} \frac{t^m}{m!}\vec{\tilde{\mathbf{s}}}_m. \tag{2.61}$$

Together with parameter K in Eq. 2.54 and desired error tolerance (to be introduced in Section 2.2.4) we have all the parameters which govern the accuracy of the solution. The meaning of these parameters is summarized in Section 2.2.8.

Let us now go back to calculating the integral present in Eq. 2.60. Plugging Eq. 2.61 into Eq. 2.60 yields:

$$\vec{\tilde{\psi}}(t) = e^{\widetilde{\mathbf{G}}_0 t}\vec{\tilde{\psi}}_0 + e^{\widetilde{\mathbf{G}}_0 t}\sum_{m=0}^{M-1}\frac{1}{m!}\int_0^t e^{-\widetilde{\mathbf{G}}_0 \tau}\,\tau^m d\tau\,\vec{\tilde{\mathbf{s}}}_m \tag{2.62}$$

Which with the following definitions of $f_m(\widetilde{\mathbf{G}}_0, t)$, $\vec{\tilde{\mathbf{w}}}_m$ and $\vec{\tilde{\mathbf{v}}}_j$:

$$f_m(z,t) \overset{\text{def}}{=} \begin{cases} \dfrac{1}{z^m}\left(e^{zt} - \displaystyle\sum_{j=0}^{m-1}\frac{(zt)^j}{j!}\right) & \text{for} \quad z \neq 0 \\ \dfrac{t^m}{m!} & \text{for} \quad z = 0 \end{cases} \tag{2.63}$$

$$\vec{\tilde{\mathbf{w}}}_m \overset{\text{def}}{=} \begin{cases} \vec{\tilde{\psi}}_0 & \text{for} \quad m = 0 \\ \vec{\tilde{\mathbf{s}}}_{m-1} & \text{for} \quad 0 < m \leq M \end{cases} \tag{2.64}$$

$$\vec{\tilde{\mathbf{v}}}_j \overset{\text{def}}{=} \sum_{m=0}^{j} \widetilde{\mathbf{G}}_0^{j-m}\vec{\tilde{\mathbf{w}}}_m \tag{2.65}$$

following the derivation in [41] the solution can be written as:

$$\vec{\tilde{\psi}}(t) = f_M(\widetilde{\mathbf{G}}_0, t)\vec{\tilde{\mathbf{v}}}_M + \sum_{j=0}^{M-1}\frac{t^j}{j!}\vec{\tilde{\mathbf{v}}}_j, \tag{2.66}$$

where the $f_M(\widetilde{\mathbf{G}}_0, t)$ is acting on $\vec{\tilde{\mathbf{v}}}_M$ and the calculations are actually performed in the discretized spectrum $z \in \sigma(\widetilde{\mathbf{G}}_0)$ (see Section 2.2.6). Now, since $\vec{\tilde{\mathbf{v}}}_j$ satisfy the recurrence relation:

$$\vec{\tilde{\mathbf{v}}}_j = \widetilde{\mathbf{G}}_0 \vec{\tilde{\mathbf{v}}}_{j-1} + \vec{\tilde{\mathbf{w}}}_j, \tag{2.67}$$

the overall computational cost of Eq. 2.66 is reduced to $M+K$ matrix vector multiplications.

2.2.4 Introducing time-dependent Hamiltonian

In the case of time-dependent Hamiltonian:

$$\frac{\partial \vec{\tilde{\psi}}(t)}{\partial t} = -\frac{i}{\hbar}\,\widetilde{\mathbf{H}}(t)\vec{\tilde{\psi}}(t) + \vec{\tilde{\mathbf{s}}}(t) \tag{2.68}$$

or rather:

$$\frac{\partial \vec{\tilde{\psi}}(t)}{\partial t} = \widetilde{\mathbf{G}}(t)\vec{\tilde{\psi}}(t) + \vec{\tilde{\mathbf{s}}}(t), \tag{2.69}$$

the Duhamel principle does not yield a closed form solution. Instead an iterative procedure can be used to obtain better and better approximations of the solution. First let us move the time dependence from $\widetilde{\mathbf{G}}(t)$ to $\vec{\tilde{\mathbf{s}}}$ by defining an extended source term $\vec{\tilde{\mathbf{s}}}_e$:

$$\vec{\tilde{\mathbf{s}}}_e(\vec{\tilde{\psi}}(t), t) \stackrel{\text{def}}{=} \vec{\tilde{\mathbf{s}}}(t) + \bar{\widetilde{\mathbf{G}}}(t)\vec{\tilde{\psi}}(t), \tag{2.70}$$

where $\bar{\widetilde{\mathbf{G}}}(t) \stackrel{\text{def}}{=} \widetilde{\mathbf{G}}(t) - \widetilde{\mathbf{G}}_{avg}$ and $\widetilde{\mathbf{G}}_{avg}$ is averaged time-independent component of $\widetilde{\mathbf{G}}$. The equation to be solved now has following form:

$$\frac{\partial \vec{\tilde{\psi}}(t)}{\partial t} = \widetilde{\mathbf{G}}_{avg}\vec{\tilde{\psi}}(t) + \vec{\tilde{\mathbf{s}}}_e(\vec{\tilde{\psi}}(t), t). \tag{2.71}$$

We can use the previous solution Eq. 2.66 to approximate the time evolution of $\vec{\tilde{\psi}}(t)$:

$$\vec{\tilde{\psi}}(t) \approx f_M(\widetilde{\mathbf{G}}_{avg}, t)\vec{\tilde{\mathbf{v}}}_M + \sum_{j=0}^{M-1} \frac{t^j}{j!}\vec{\tilde{\mathbf{v}}}_j, \tag{2.72}$$

where this time the $\vec{\tilde{\mathbf{v}}}_j$ are computed from $\vec{\tilde{\mathbf{s}}}_e$. It means that $\vec{\tilde{\mathbf{v}}}_j$ depend on $\vec{\tilde{\psi}}(t)$ which is still unknown, however the solution can be obtained via iterations. Upon first evaluation we either extrapolate from previous timestep Δt (by putting $t+\Delta t$ into Eq. 2.72) or when jump-starting the calculations we use $\vec{\tilde{\psi}}_0$. Next, in each successive evaluation we use the approximation from previous iteration within the timestep Δt (which spans M sub-timesteps). We repeat the iterative procedure until the convergence criterion at sub-step M is met:

$$\frac{||\vec{\tilde{\psi}}_{new} - \vec{\tilde{\psi}}_{prev}||}{||\vec{\tilde{\psi}}_{prev}||} < \varepsilon. \tag{2.73}$$

It means that this method has a radius of convergence which directly depends on the timestep Δt covered in the single iteration, and containing M sub-timesteps. Too large Δt will cause the successive iterations to diverge, this is the reason why this method is not a fully global method but a *semi-global* method. The useful result of this situation is that one can directly control the computation cost by setting an acceptable computation error ε. For reference solutions it can be set to ULP numerical precision, for faster calculations it can

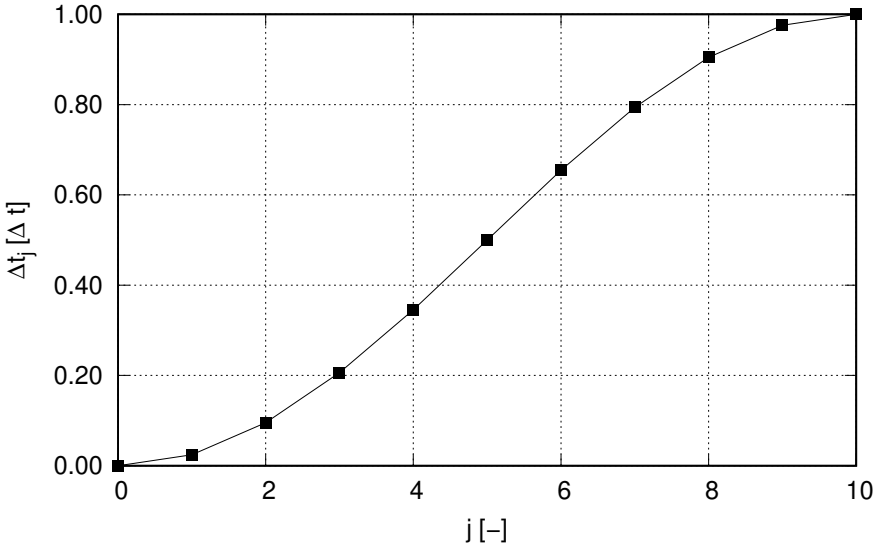


Figure 2.9: Chebyshev time points used for sub-timestep interpolation. They are equivalent to the axis projection of points equally spaced on a unit semicircle.

be a larger value. The novelty in this work is that it works also for higher precision types such as `long double` or `float128` with 33 decimal places (Table 2.1 and Table 3.2), thus enabling very accurate calculations.

Since we have put the dependence on $\vec{\tilde{\psi}}(t)$ into $\vec{\tilde{\mathbf{s}}}_e(\vec{\tilde{\psi}}(t),t)$ it is also computationally inexpensive to put this dependence into the Hamiltonian hence the method described above works also for Eq. 2.50. So putting it all together, the solution to Eq. 2.50:

$$\frac{\partial \vec{\tilde{\psi}}(t)}{\partial t} = -\frac{i}{\hbar}\,\widetilde{\mathbf{H}}(\vec{\tilde{\psi}}(t),t)\vec{\tilde{\psi}}(t) + \vec{\tilde{\mathbf{s}}}(t).$$

is following:

$$\vec{\tilde{\psi}}_{new}(t) \overset{\text{iterate}}{=} f_M(\widetilde{\mathbf{G}}_{avg},t)\vec{\tilde{\mathbf{v}}}_M(\vec{\tilde{\psi}}_{prev}) + \sum_{j=0}^{M-1} \frac{t^j}{j!}\vec{\tilde{\mathbf{v}}}_j(\vec{\tilde{\psi}}_{prev}), \tag{2.74}$$

where iterations are performed until the convergence condition Eq. 2.73 is met. Thanks to being able to extrapolate $\vec{\tilde{\psi}}_{prev}$ into the next timestep Δt by putting $t+\Delta t$ into above equation and with a good choice of M, K parameters usually one iteration is enough to achieve desired convergence.

2.2.5 Sub-timesteps in Chebyshev points and Newton interpolation

When interpolating a function (such as Eq. 2.56), an equidistant set of points is not a good choice: the closer to the boundary of the interpolation domain the less accurate is the interpolation. This effect is known as the Runge phenomenon. A much better set of points is with points becoming denser closer to the edges of the domain. Such a set of points, chosen with following formula:

$$\Delta t_j \overset{\text{def}}{=} -\frac{1}{2}\left(\cos\left(\frac{j\pi}{M-1}\right)+1\right)\Delta t \tag{2.75}$$

is known as Chebyshev points (see example for M=11 in Fig. 2.9). These points are used as the sub-timesteps Δt_j of Δt discussed in preceding sections.

A Newton interpolation of function $f(t)$ at points Δt_j is defined as:

$$f(t) \approx \sum_{n=0}^{N} a_n R_n(t) \tag{2.76}$$

where a_n are coefficients of the expansion and R_n are Newton basis polynomials defined as: $R_0(t) = 1$ and $R_n(t) = \prod_{j=0}^{n-1}(t - \Delta t_j)$. The a_n are calculated as the divided differences from the interpolated function. It is used in Eq. 2.54.

2.2.6 Arnoldi approach

Calculation of evolution operator in previous Section 2.1 requires the knowledge of spectral range of this operator. That method cannot be used when it is impossible to estimate the eigenvalue domain. The difficulty with such estimation grows when the eigenvalues are distributed on the complex plane, which is the case with absorbing boundary conditions (see Section 2.2.9). And almost all interesting use cases of time propagation (e.g. a multi level diatomic molecular system evolving under a laser impulse) require absorbing boundary conditions. In such, quite common, situation the Arnoldi approach comes to the rescue, because it works without required prior knowledge of the eigenvalue domain.

The Arnoldi approach calculates the $f_M(\widetilde{\mathbf{G}}_{avg}, t)\vec{\widetilde{\mathbf{v}}}_M$ in Eq. 2.74 in following manner:

(a) First construct an orthonormalized (via Gram-Schmidt process) reduced Krylov subspace $\vec{\widetilde{\mathbf{v}}}$, $A\vec{\widetilde{\mathbf{v}}}$, $A^2\vec{\widetilde{\mathbf{v}}}$, … , $A^K\vec{\widetilde{\mathbf{v}}}$ (the K parameter controls the accuracy, see Section 2.2.8 and $A = \widetilde{\mathbf{G}}_{avg}$).

(b) Construct the transformation matrix Υ from the reduced Krylov basis representation to the position representation of $\vec{\widetilde{\mathbf{v}}}_j$ vectors.

(c) During the process determine the eigenvalue domain and rescale the domain using method [82] by dividing by the *capacity of the domain* to reduce numerical errors.

(d) Perform the calculation in the reduced Krylov basis representation then transform the result back to original positional representation of $\vec{\widetilde{\mathbf{v}}}_j$ using the transformation matrix Υ.

2.2.7 Extension to coupled time-dependent Schrödinger equations

The critical observation to extend this semi-global algorithm to an arbitrary number of coupled Schrödinger equations is that this method is independent of the Hamiltonian used. There is no requirement that this Hamiltonian is a single level system or several coupled levels. The method presented in Section 2.1.5 applies exactly (including a corresponding version of Eq. 2.44). The method uses the iterative procedure to obtain the solution. The iterations are performed via invoking the $\hat{\mathbf{H}}(t)$ acting on ψ until a convergence criterion is met: the difference between the wavefunction from previous iteration and current iteration is smaller than predetermined tolerance ε. This iterative procedure is executed inside a single timestep Δt over the M sub-timesteps.

During this calculation all the expansion coefficients, both with respect to K (Eq. 2.54) and with respect to M (Eq. 2.61) are stored inside a matrix of size $N \times K$ and $N \times M$ respectively, where N is the number of grid points in the discretized wavefunction. The usual matrix algebra is used in the C++ implementation, hence preserving the original code of the implementation when moving to several coupled Schrödinger equations is desirable. And it is possible, because the algorithm itself is agnostic to the Hamiltonian: it only invokes the $\widetilde{\mathbf{H}}(t)\psi$ in the computation. The trick lies in the memory layout of the storage: the coupled wavefunctions are stored one after another inside a single column vector. A system of k coupled wavefunction uses kN grid points. The system simply becomes larger and the semi-global algorithm uses matrices of sizes $(kN) \times K$ and $(kN) \times M$ respectively without being informed how the information (stored in the column vector of size kN) is used by the Hamiltonian. The technical implementation details are discussed in Section 3.3.1. Additionally the same approach can be used when adapting semi-global algorithm to solving higher dimensional systems with more spatial directions, such as a system of three atoms using Jacobi coordinates and with coupled Schrödinger equations therein.

2.2.8 Summary of parameters used by the semi-global method

The semi-global time integration method uses following parameters: K introduced in Eq. 2.55 to compute the evolution operator, M introduced in Eq. 2.61 to help with the convergence of the iterative process, the error tolerance ε (Eq. 2.73) and the „global" timestep Δt (Eq. 2.75) over which the converging sub-iterations are being computed. The short summary of these parameters is listed in Tab. 2.2.

Table 2.2: Summary of parameters of semi-global time propagation algorithm

parameter	meaning	equation
K	The number of expansion terms used for the computation of the function of matrix (evolution operator).	Eq. 2.55
M	The number of interior time points in each timestep.	Eq. 2.61
ε	Tolerance: the largest acceptable computation error.	Eq. 2.73
Δt	The length of the timestep interval (it contains M sub-timesteps).	Eq. 2.75

2.2.9 Absorbing boundary conditions with a complex potential

The numerical damping presented in Section 2.1.6 cannot be applied to the semi-global time propagation algorithm presented in this section. The evolution operator is expanded in the Chebyshev time points using finite differences. The recurrence relation Eq. 2.46 no longer applies. Instead an optimized complex absorbing potential is used [41, 83, 84]. In the damping band a sequence of square complex barriers is placed, each of them having their own reflection and transmission amplitudes for a plane wave [84]. The parameters of each barrier are optimised with respect to the cumulated plane wave survival probability of all barriers. The typical characteristic of such barrier is that it damps momentum within a certain momentum range and this range pertains to the current problem being calculated. After the optimisation procedure is complete the complex potential is added to the potential used in

the time propagation. The Arnoldi approach works correctly with complex potentials (see Section 2.2.6) and this is why it is used by the semi-global time propagation method.

2.3 Calculations in higher numerical precision

There is a need for higher precision computations in quantum dynamics, especially for attosecond laser impulses where the simulated timespan is very short and the timestep is very small [1]. A very high temporal resolution is necessary to describe such system. It stems from the simple fact that with small timesteps (which are necessary to describe a quickly changing electromagnetic field) there are a lot of small contributions from each timestep. Upon adding these contributions many times, the errors will accumulate[13]. And only higher precision calculations can make the errors significantly smaller.

High-precision computation is an attractive solution in such situations, because even if a numerically better algorithm with smaller error or faster convergence is known for a given problem (e.g. Kahan summation [85] for avoiding accumulating errors[13]), it is often easier and more efficient to increase the precision of an existing algorithm rather than deriving and implementing a new one [86, 87]. However, switching to high-precision generally means longer run times [88, 89] as shown in the benchmarks in Section 3.5.

Nowadays, high-precision calculations find application in various different domains, such as long-term stability analysis of the solar system [86, 90, 91], supernova simulations [92], climate modeling [93], Coulomb n-body atomic simulations [94, 95], studies of the fine structure constant [96, 97], identification of constants in quantum field theory [98, 99], numerical integration in experimental mathematics [100, 101], three-dimensional incompressible Euler flows [102], fluid undergoing vortex sheet roll-up [99], integer relation detection [103], finding sinks in the Henon Map [104] and iterating the Lorenz attractor [105]. There are many more yet unsolved high-precision problems in electrodynamics [106]. In quantum mechanics the extended Hylleraas three electron integrals are calculated with 32 digits of precision in [107]. The long range asymptotics of exchange energy in a hydrogen molecule is calculated with 230 digits of precision in [108]. In quantum field theory calculations of massive 3-loop Feynman diagrams are done with 10000 decimal digits of precision in [98]. Moreover the experiments in CERN are being performed in higher and higher precision as discussed in the "Welcome to the precision era" article [109]. It brings focus to precision calculations and measurements which are performed to test the Standard Model as thoroughly as possible, since any kind of deviation will indicate a sign of new physics.

Consequently, I believe that in the future the high-precision calculations will become increasingly more popular and necessary to advance the science. I have implemented the algorithms presented here in high-precision. In Section 3.5 I show the performance benchmarks with additional details. See also my work [28] for high-precision classical dynamics.

[13] for n summands and ε Unit in Last Place (ULP) error, the error in regular summation is $n\varepsilon$, error in Kahan summation [85] is 2ε, while error with regular summation in twice higher precision is $n\varepsilon^2$. See proof of Theorem 8 in [85].

Chapter 3

Implementation and validation

In this chapter I will describe the implementation of both numerical methods from Chapter 2 and discuss the validation and verification of these implementations. As previously, I will start with the implementation of TDSE for time-independent Hamiltonian and verify its correctness. Next I will move onto TDSE for time-dependent Hamiltonian and verify it.

3.1 Implementation of TDSE for time-independent Hamiltonian

The main summation of the series (Eq. 2.43, Eq. 2.44) is implemented and shown in the Listing 3.1 of the function `State::propagateByDt`.

First, the R and G parameters are calculated in lines 3-4 using Eq. 2.36. They are declared to be of type `Real`. The numerical precision of the `Real` type and precision of all calculations is determined by the user at the stage of compiling the code (see Sections 2.1.1, 2.3 and 3.5 for details). The amount of elements in the series (variable `maxEl` declared in line 5) on the RHS of Eq. 2.43 (summed in the `for` loop in the line 21) is determined based on the value of R as recommended in [73]: the cutoff is happening at $1.3R$, but it is larger than 40 (to avoid the numerical error). This cutoff is valid, because as noted earlier the a_k coefficients are decaying exponentially after $k > R$ as shown in Fig. 2.1. For `double` type the Bessel function values are numerically negligible [73]. However when using higher precision types I am not using `maxEl`, but instead directly compare a_k with C++ `numeric_limits::epsilon` of selected `Real` type. For brevity this check is not shown in the code listing.

Next, in lines 7-10 the variables are declared to hold the data necessary to construct the wavefunction upon which the Hamiltonian is acting. The type `MultiVectorXcr` is a vector of wavefunctions[14] one per each coupled electronic state[15]. The `psi_0` and `psi_1` variables hold the wavefunctions representing respectively the but-last and last element in the series. Both of these elements are necessary (see Eq. A.2) to construct the next element in the series,

[14] Using plural *wavefunctions* might be confusing, so here's a clarification: on each coupled electronic state there is a wavefunction evolving on the electronic potential assigned to it. All of them together sit inside a C++ `std::vector` container. When there is no confusion in the context I am using *wavefunction* to refer to all coupled *wavefunctions*, otherwise I emphisize in text whether I mean a single coupled wavefunction or all wavefunctions.

[15] To be precise in C++ the following types are defined:

```
using VectorXcr = Eigen::Matrix<Complex, Eigen::Dynamic, 1>;
using MultiVectorXcr = std::vector<VectorXcr>;
```

Code listing 3.1: The evolution operator implemented in C++

```
1  void State::propagateByDt(Real dt)
2  {
3      Real R       = calcKosloffR(dt);
4      Real G       = calcKosloffG(dt);
5      Real maxEl   = max(Real(40), 1.3*R); // Kosloff recommendation
6
7      MultiVectorXcr psi_0 = wf;           // start calculations of wf
8      MultiVectorXcr psi_dt= zero(psi_0);  // will become ψ(t+dt)
9      MultiVectorXcr psi_1 = zero(psi_0);
10     MultiVectorXcr psi_2 = zero(psi_0);
11
12     // start the series
13     psi_1      = calc_Hnorm_psi(psi_0,dt,R,G); // ψ1 = H_norm ψ0
14     psi_1      = psi_1 * dampingTable;         // ψ1 = e^-γ ψ1
15     Complex ak0= calcAk(0, R, G);              // a0
16     Complex ak1= calcAk(1, R, G);              // a1
17     psi_dt     = ak0 * psi_0 + ak1 * psi_1;    // ψ(t+dt)=a0 ψ0 +a1 ψ1
18
19     // We have the first two summands, now calculate the rest
20     Complex ak(1);
21     for (int i = 2; i < maxEl; i++) {
22         psi_2 = calc_Hnorm_psi(psi_1,dt,R,G);  // ψ2 =H_norm ψ1
23         psi_0 = psi_0 * dampingTable;          // ψ0 =e^-γ ψ0
24         psi_2 = 2 * psi_2 - psi_0;             // ψ2 =2*H_norm ψ1 -ψ0
25         psi_2 = psi_2 * dampingTable;          // ψ2 =e^-γ ψ2
26         ak    = calcAk(i, R, G);               // ak
27         psi_dt= psi_dt + ak * psi_2;           // ψ(t+dt)=ψ+ak ψk
28         psi_0 = psi_1;                         // ψ0 ← ψ1
29         psi_1 = psi_2;                         // ψ1 ← ψ2
30     }
31     wf = psi_dt;                               // final result
32 };
```

which is called `psi_2`. The `psi_dt` holds the wavefunctions representing the final result of the series.

Next, in lines 12-17 the first two elements of the series are calculated to jump start the loop that follows. Hence, the construction of each next element (line 24) of the series is using the Chebyshev recurrence relation Eq. A.2 based on the two previous elements (line 28 and 29) in the loop.

The numerical damping boundary conditions [75] (Eq. 2.46) are applied in the lines 14, 23 and 25 as described in Section 2.1.6. If there are no boundary conditions used then the `dampingTable` equals 1 in all grid points and these three lines do not affect the wavefunction.

We will focus now on the function `State::calc_Hnorm_psi` called in lines 13 and 22 of Listing 3.1. It describes the single action of the normalized Hamiltonian $\widetilde{\mathbf{H}}_{norm}$ on the coupled electronic states. The implementation of this function is shown in Listing 3.2. To build the elements of the series (Eq. 2.44) the summation is split into three parts. In the first part (lines 6-9) the kinetic operator acts on each of the coupled electronic states by calling `State::Ekin_single`. Next in lines 11-14 the normalization $-(1+G/R)$ component (Eq. 2.43) is taken into account, and finally in lines 16-23 the potential matrix $\widetilde{V}_{jk}$ acts on each coupled wavefunction. It shall be noted that although we use the normalized Hamilto-

Code listing 3.2: The normalized Hamiltonian operator for coupled electronic states in C++

```
1  MultiVectorXcr State::calc_Hnorm_psi(
2          const MultiVectorXcr& psi_0, Real dt, Real R, Real G
3  ) {
4      MultiVectorXcr psi_ret = zero(psi_0);
5
6      // The kinetic operator, acting only on the diagonal
7      for (int j = 0; j < levels; j++) {
8          psi_ret[j] = Ekin_single(psi_0[j])*(dt/(PhysConst::hbar*R));
9      }
10
11     // The -(1+G/R) component in the normalized Hamiltonian
12     for (int j = 0; j < levels; j++) {
13         psi_ret[j] = psi_ret[j].array()-psi_0[j].array()*(1+G/R);
14     }
15
16     // The potential operator matrix acting on the wavefunction
17     for (int j = 0; j < levels; j++) {
18         for (int k = 0; k < levels; k++) {
19             psi_ret[j] = psi_ret[j].array()
20                          + potMatrix[j][k].array()*psi_0[k].array()
21                            *(dt/(PhysConst::hbar*R));
22         }
23     }
24     return psi_ret;
25 }
```

nian in `State::calc_Hnorm_psi`, the final result of the loop in the Listing 3.1 is the action of not normalized Hamiltonian $\widetilde{\mathbf{H}}$ on the full wavefunction, because the a_k factors contain the $e^{-i(R+G)}$ phase component[16], see Eq. 2.33, Eq. 2.34 and Eq. 2.36.

The Listing 3.3 shows the kinetic operator `State::Ekin_single` acting on a single electronic wavefunction. It follows the method detailed in Appendix B. The Fast Fourier Transform (FFT) is applied on the wavefunction (line 5), next in the momentum representation it is multiplied by $-\vec{k}^2$, next (line 7) the inverse FFT is used and then the $\frac{-\hbar^2}{2m}$ factor

[16] In fact this constant factor could be moved outside the loop (outside the Σ in Eq. 2.43), however at present I consider correctness to be more important than optimisation and leave this step for the later stage when the YADE unit tests for quantum dynamics will have a full coverage of present code.

Code listing 3.3: The kinetic operator using FFT implemented in C++

```
1  VectorXcr State::Ekin_single(const VectorXcr& psi_0)
2  {
3      VectorXcr  psi_1 = VectorXcr::Zero(commonConfig.points);
4      const Real mass  = commonConfig.mass;    // reduced mass
5      doFFT(psi_0, psi_1);                     // ψ1 =        F(ψ0)
6      psi_1 = psi_1.array()*minusKSqr.array();// ψ1 =     -k²F(ψ0)
7      doIFFT(psi_1, psi_1);                    // ψ1 =F⁻¹(-k²F(ψ0))=∇²ψ0
8      psi_1 *= (-PhysConst::hbarSqr/(2*mass));// ψ1 =(-ħ²(∇²ψ0)/(2m))
9      return psi_1;
10 }
```

is applied. The main limitation of current implementation of Schrödinger equation is baked into this function (line 4), namely that it describes the evolution of a single particle after the reduction of the coordinate system. Usually it is the reduced mass of a diatomic molecule, sometimes the mass of a single propagating particle. In the future this code can be improved to handle triatomic molecules (e.g. in the Jacobi coordinate system, see page 38 in [38]), in such case in this line there will appear multiple masses. Each of them will be corresponding to a separate degree of freedom and to a separate spatial direction in the numerical matrix holding the single coupled wavefunction state[17].

Finally, after enough (i.e. `maxEl`) elements of the series have been summed in Listing 3.1 (line 31), the calculated value of `psi_dt` is assigned to class variable `State::wf`. Thus the single propagation timestep has been completed.

3.2 Validation of time-independent Hamiltonian using standard benchmarks

To validate present implementation, in this section I will reproduce the solution of two problems for a system of coupled Schrödinger equations featuring nonadiabatic quantum dynamics. They were originally introduced in [110] and afterwards carefully analysed in [111]. Since then they are considered to be a standard benchmark for coupled systems. In these problems the diagonal of the diabatic potential energy surfaces $V_{11}(R)$ and $V_{22}(R)$ undergoes:

(a) a single crossing as in Fig. 3.1a and

(b) a dual crossing as in Fig. 3.4a.

In the crossing region there is a strong coupling $V_{12}(R)$ between the two levels. In adiabatic representation the potential energy surfaces $E_1(R)$ and $E_2(R)$ undergo respectively a single (Section 3.2.1 and Fig. 3.1b) and dual (Section 3.2.2 and Fig. 3.4b) avoided crossing. The nonadiabatic coupling matrix elements $D_{12}(R)$ between the two levels are relatively large (Fig. 3.1b and Fig. 3.4b).

The calculations are performed diabatically because the coupling elements are smaller and the Hamiltonian assumes a simpler form [39, 40][18]. To gain more insight from the wave packet dynamics on each of these levels the results are presented here in adiabatic representation. The two levels $E_1(R)$ and $E_2(R)$ do not cross and the evolution of wavefunction probability distributions (Figs. 3.2 and 3.5) on the lower and the higher electronic surface is easier to interpret. The transmission and reflection probabilities (Figs. 3.3 and 3.6) on $E_1(R)$ and $E_2(R)$ are calculated as integrals (on a discrete grid) of the single coupled wavefunction in the nuclear coordinate range satisfying $R > 0$ and $R < 0$ respectively. To obtain these

[17] Also in such case the type `using VectorXcr = Eigen::Matrix<Complex, Eigen::Dynamic, 1>;` from footnote 15, which is used to hold the data for a single coupled wavefunction, will have to be changed into a type which can hold a higher dimensional data. The `boost::ublas::tensor` from the Boost library [60] is a likely candidate.

[18] See Eqs. 2.12 and 2.100 in [40] for adiabatic TDSE: $i\hbar\frac{\partial\psi}{\partial t} = -\frac{\hbar^2}{2m}(\nabla+\tau)^2\psi$ (where τ is the nonadiabatic coupling matrix) and Eqs. 2.22 and 2.115 for diabatic TDSE: $i\hbar\frac{\partial\chi}{\partial t} = \left(-\frac{\hbar^2}{2m}\nabla^2 + V\right)\chi$; it can be seen that the position of τ is rather unfortunate in the adiabatic representation.

results the wavefunctions are converted from diabatic representation to adiabatic representation by first performing the diagonalization of the potential matrix $V(R)$ for each value of R. The obtained eigenvalues are the adiabatic potential energy surfaces $E_1(R)$ and $E_2(R)$ [111]. Next, the matrix of two eigenvectors $\phi_1(R)$ and $\phi_2(R)$:

$$P(R) = \begin{bmatrix} P_{11}(R) & P_{12}(R) \\ P_{21}(R) & P_{22}(R) \end{bmatrix} = [\, \phi_1(R) \;\; \phi_2(R) \,], \tag{3.1}$$

is used to compute the nonadiabatic coupling matrix elements $D_{12}(R)$ with:

$$D_{12}(R) = \phi_1^* \cdot \frac{\partial}{\partial R}\phi_2 = P_{11}^* \frac{dP_{12}}{dR} + P_{21}^* \frac{dP_{22}}{dR}. \tag{3.2}$$

Above the electronic integrals become a dot product or a sum in a two state basis. Finally to obtain the adiabatic wavefunction $\psi(R,t)$ from the diabatic one, $\chi(R,t)$, the following transformation is used:

$$\psi(R,t) = P(R)\chi(R,t) \tag{3.3}$$

or more specifically, since in these two examples we are dealing with a two level system[19]:

$$\begin{bmatrix} \psi_1(R,t) \\ \psi_2(R,t) \end{bmatrix} = \begin{bmatrix} P_{11}(R) & P_{12}(R) \\ P_{21}(R) & P_{22}(R) \end{bmatrix} \begin{bmatrix} \chi_1(R,t) \\ \chi_2(R,t) \end{bmatrix}. \tag{3.4}$$

The simulations begin with the single wavefunction placed on the lower energy surface $E_1(R)$ assuming shape of a a Gaussian wavepacket[20]:

$$\chi_{Gauss}(R) = \pi^{-\frac{1}{4}} a^{-\frac{1}{2}} e^{-\frac{(R-R_0)^2}{2a^2} - ik_0(R-R_0)}. \tag{3.5}$$

The initial values of parameters assumed in each simulation are listed in Tab. 3.1. They were chosen so as to best represent the undergoing evolution of the coupled system dynamics and to completely reproduce the results in [110, 111].

Table 3.1: Gauss wavepacket parameters used to reproduce the standard benchmark [110, 111].

parameter [a.u.]		single crossing		dual crossing	
		high k_0	low k_0	high k_0	low k_0
mass	m	2000	2000	2000	2000
wavenumber	k_0	15	8.5	52	30
packet width	a	0.75	0.8	0.7	0.7
start position	R_0	-4	-4.15	-8	-8

[19] Alternative method of obtaining the transformation matrix $P(R)$ is to use the Eq. 3.45 from [40]: $\beta(R) = -\frac{1}{2}\tan^{-1}\left(\frac{V_{12}(R)}{V_{11}(R)}\right)$ and then use the rotation matrix from in Eq. 3.44 [40].

[20] To be precise in the code the following formula is used: $\chi_{Gauss}(R) = \frac{\exp\left(-\frac{m(R-R_0)^2 + ia^2k_0(k_0\hbar(t-t_0) - 2m(R-R_0))}{2a^2m + 2i\hbar(t-t_0)}\right)}{\sqrt{\sqrt{\pi}\left(a + \frac{i\hbar(t-t_0)}{am}\right)}}$, as it is the solution of a free propagating Gaussian wave packet, in Eq. 3.5 it was assumed that $t_0 = 0$.

3.2.1 Single avoided crossing

In the first standard benchmark [110, 111] the potential matrix elements in diabatic representation $V_{ij}(R)$ are defined as follows (Fig. 3.1a):

$$\begin{aligned} V_{11}(R) &= \begin{cases} A\left(1-e^{-BR}\right) & \text{for} \quad R \geq 0 \\ -A\left(1-e^{BR}\right) & \text{for} \quad R < 0 \end{cases} \\ V_{22}(R) &= -V_{11}(R) \\ V_{12}(R) &= V_{21}(R) = Ce^{-DR^2} \end{aligned} \tag{3.6}$$

with the parameters assuming values: $A = 0.01$, $B = 1.6$, $C = 0.005$ and $D = 1.0$. In this example the diabatic surfaces cross at nuclear coordinate $R = 0$ and a Gaussian off-diagonal potential is assumed centered at this point. The adiabatic surfaces $E_1(R)$ and $E_2(R)$ (Fig. 3.1b) repel each other in the strong-coupling region and a large nonadiabatic coupling element D_{12} (Eq. 3.2) appears at the avoided crossing.

Figure 3.2 shows the time evolution of the probability distributions (squared amplitude of the wavefunction $|\psi(R,t)|^2$) on each of the adiabatic electronic surfaces. And Fig. 3.3 shows the transmission and reflection probabilities (integrals for $R > 0$ and $R < 0$) evolving over time.

In the high momentum case (Fig. 3.2a) the packet has enough energy to put about 32% of the population on the upper energy surface. Entering higher level $E_2(R)$ caused the wave packet to lose energy, it has smaller momentum and is propagating slower as can be seen by the time labels put beneath the center of each packet in the Fig. 3.2a. Figure 3.3a shows the transmission and reflection probabilities for high momentum case. It can be seen that whole packet passes through the crossing point and reflection vanishes over time. The final population is about twice higher on the lower electronic surface than on the upper one.

In the low momentum case (Fig. 3.2b) the packet doesn't have enough energy to populate the higher energy surface. Note the vertical scale on the upper level ρ_2 in Fig. 3.2b. After going up, the packet almost does not move forward and instead it is leaking back to the lower level in both directions. It can be seen in Fig. 3.3b that the reflection weakly increases over time while transmission on both electronic surfaces slowly decreases. Most (91%) of the final population resides on the lower surface.

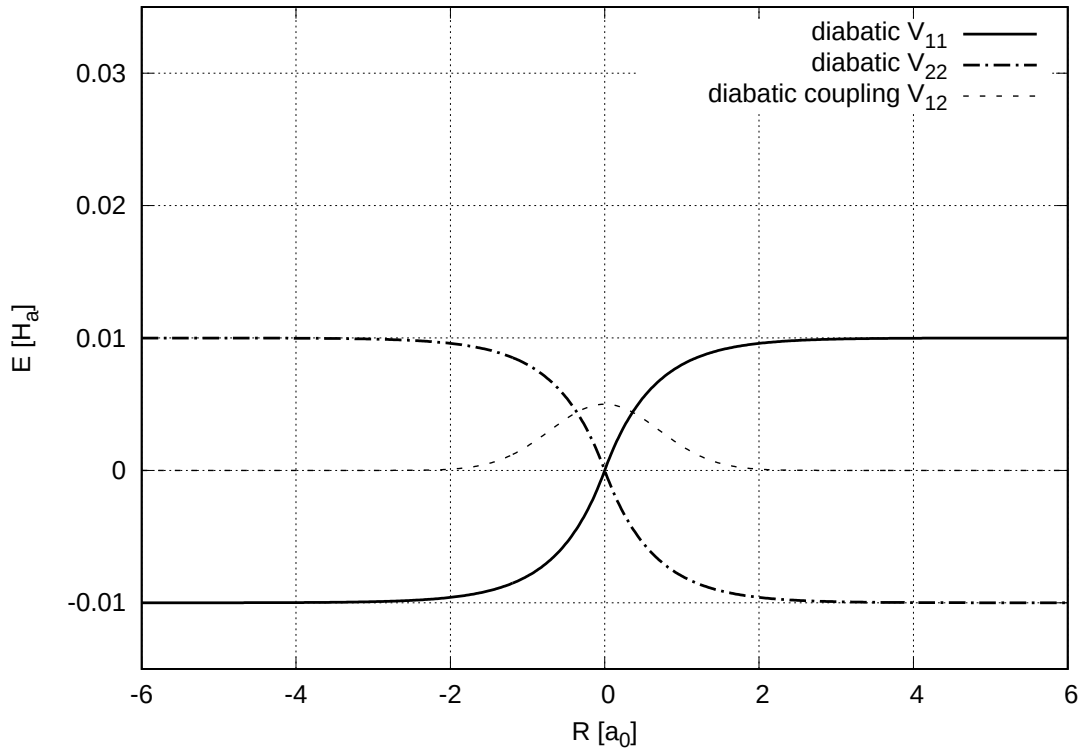
The obtained results are in very good agreement with [110, 111]. Please note that I obtained these results using a different, arguably more precise [64], time integration algorithm than the one used in [110, 111].

3.2.2 Dual avoided crossing

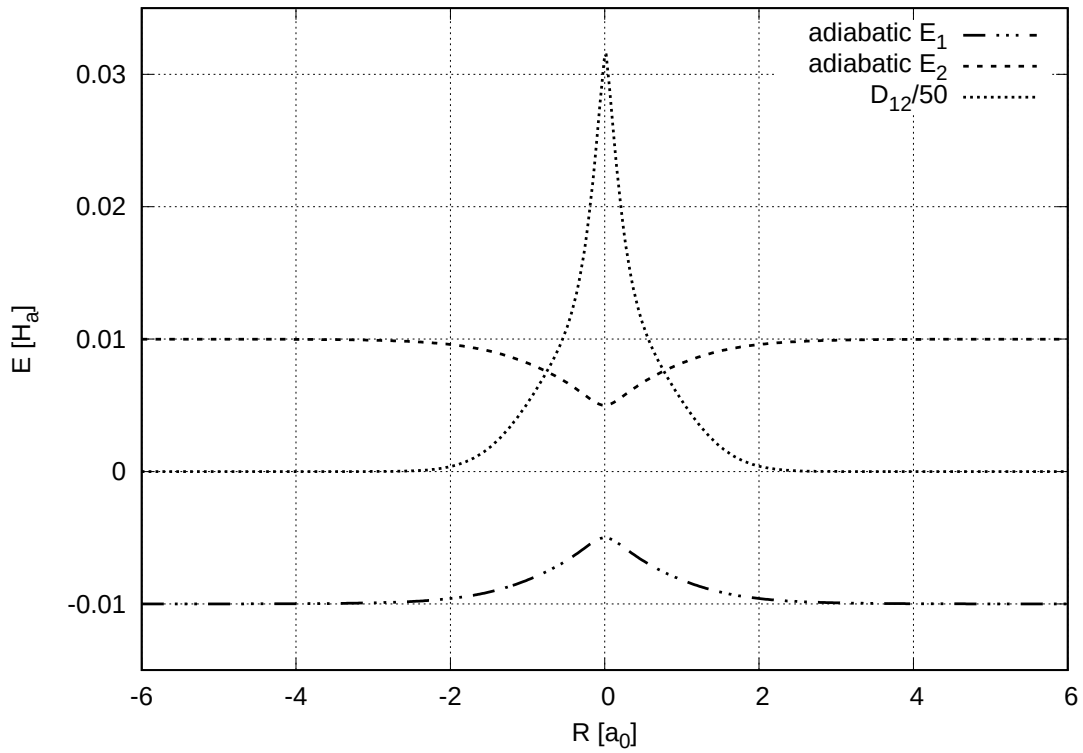
In the second standard benchmark [110, 111] the potential matrix elements in diabatic representation $V_{ij}(R)$ are defined as follows (Fig. 3.4a):

$$\begin{aligned} V_{11}(R) &= 0 \\ V_{22}(R) &= -Ae^{-BR^2} + E_0 \\ V_{12}(R) &= V_{21}(R) = Ce^{-DR^2} \end{aligned} \tag{3.7}$$

with the parameters assuming values: $A = 0.1$, $B = 0.28$, $C = 0.015$, $D = 0.06$ and $E_0 = 0.05$. In this example the diabatic potentials cross each other twice and a wide Gaussian off-

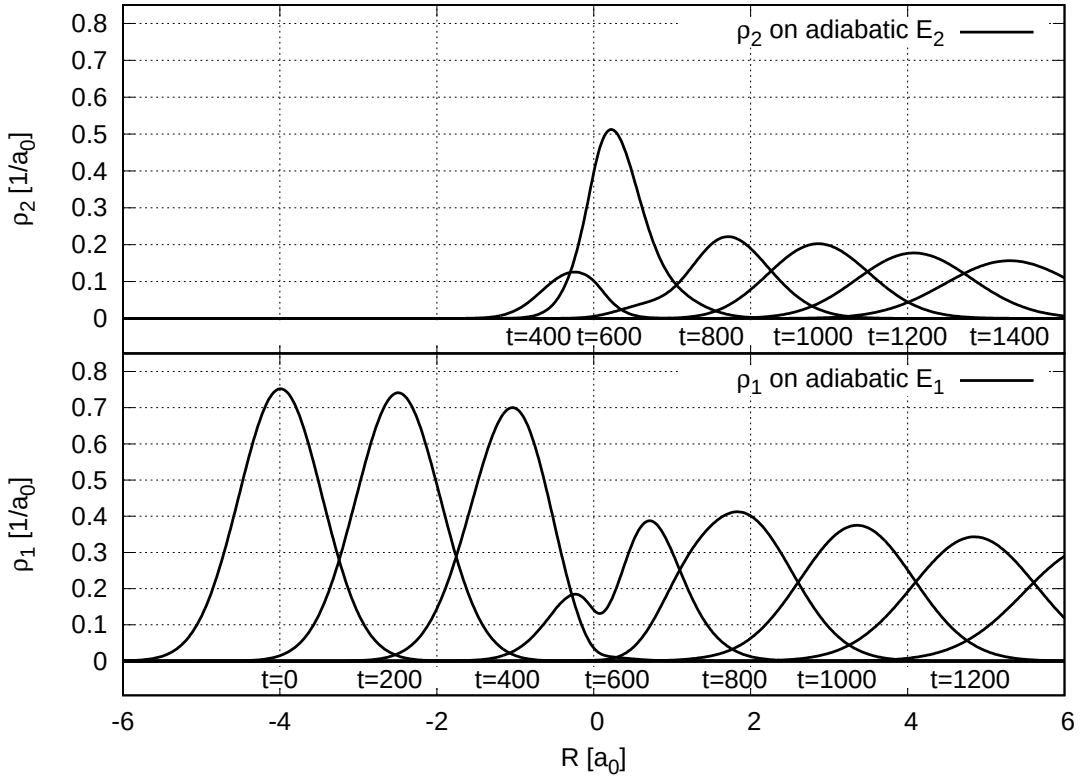


(a)

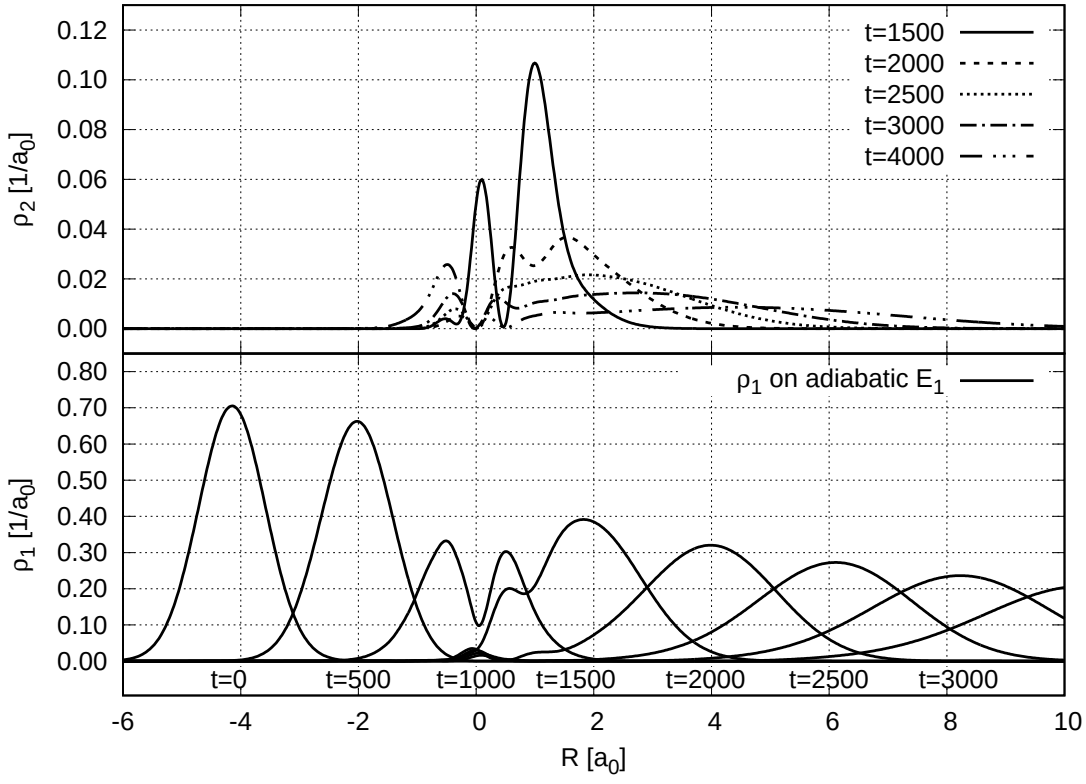


(b)

Figure 3.1: Model surfaces potential matrix of the simple avoided crossing example; (a) diabatic representation; (b) adiabatic representation, D_{12} is the nonadiabatic coupling element (it is drawn as divided by 50 because the coupling is large).

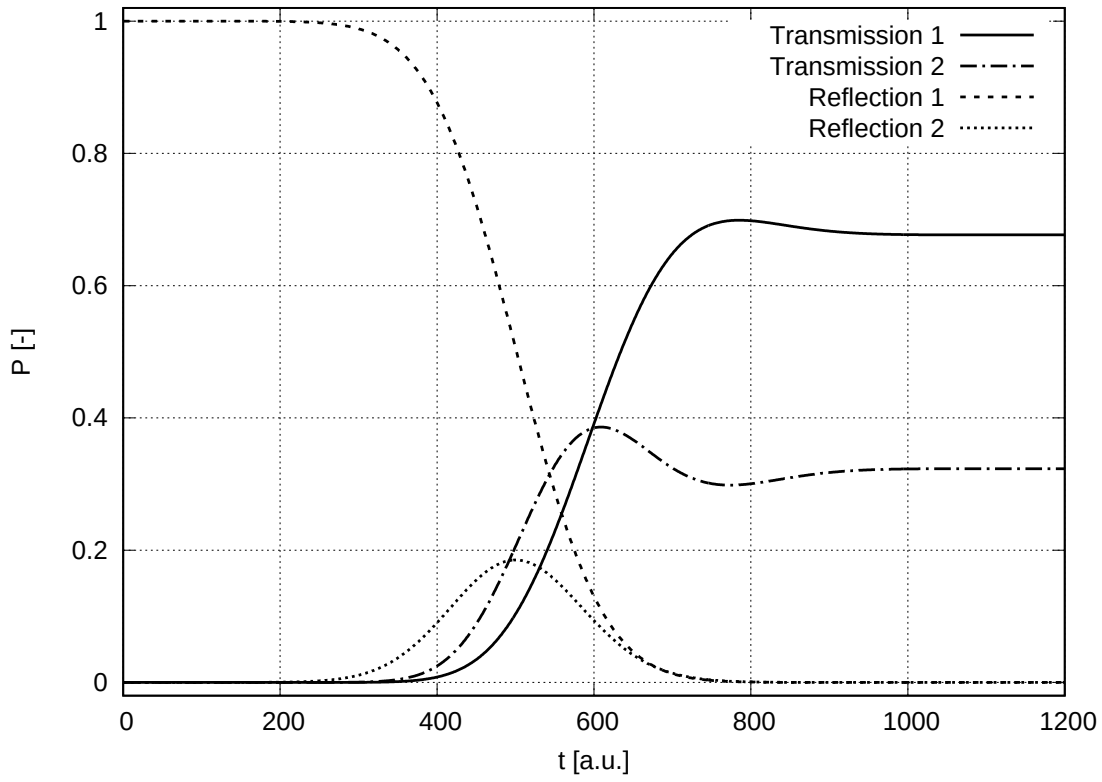


(a)

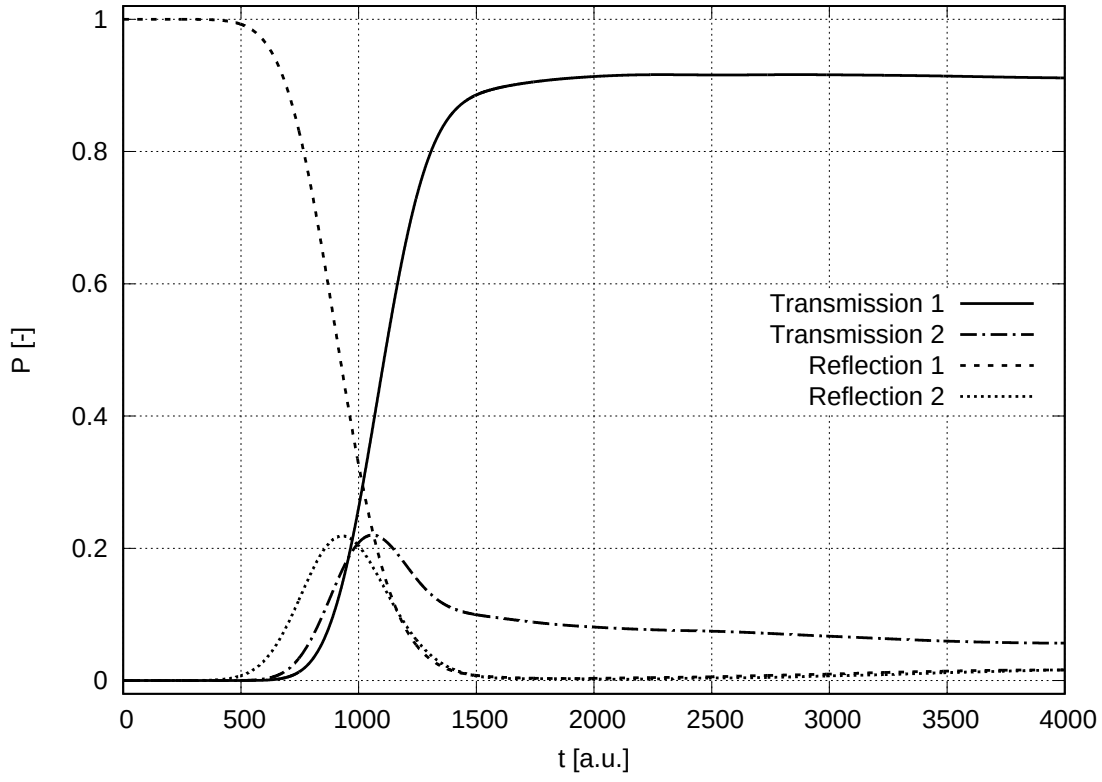


(b)

Figure 3.2: Time evolution of probability distributions on adiabatic energy surfaces, simple avoided crossing; (a) high momentum Gaussian wavepacket $k = 15$ a.u.; (b) low momentum Gaussian wavepacket $k = 8.5$ a.u.

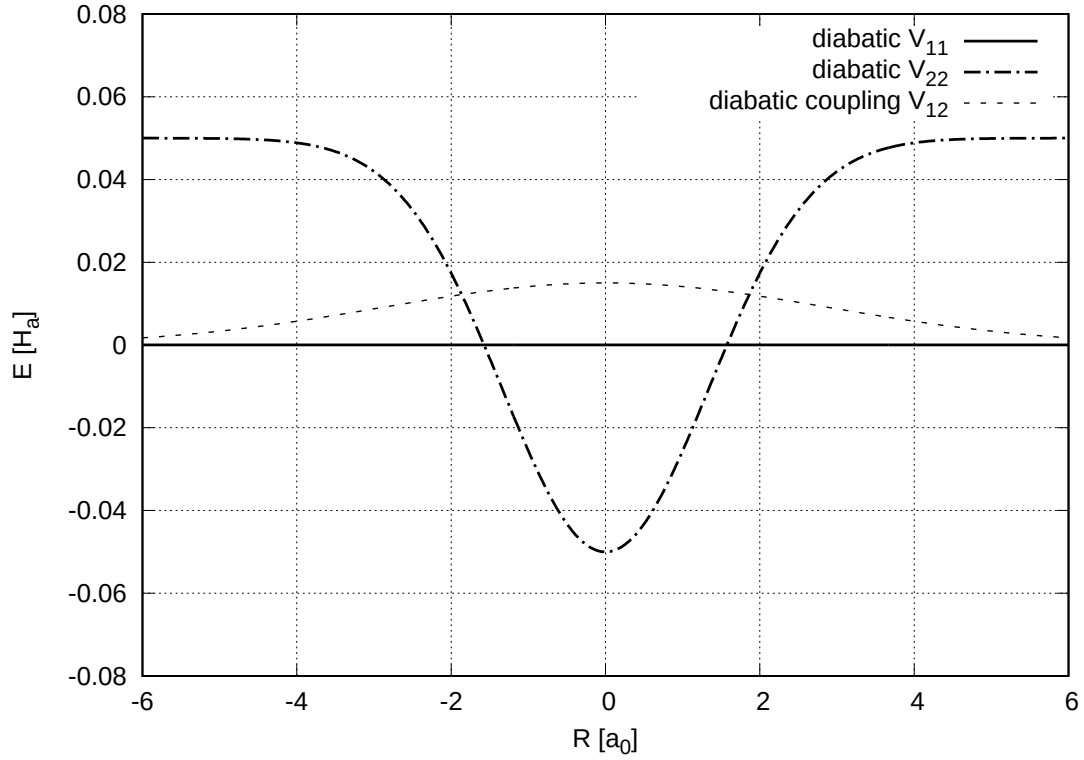


(a)

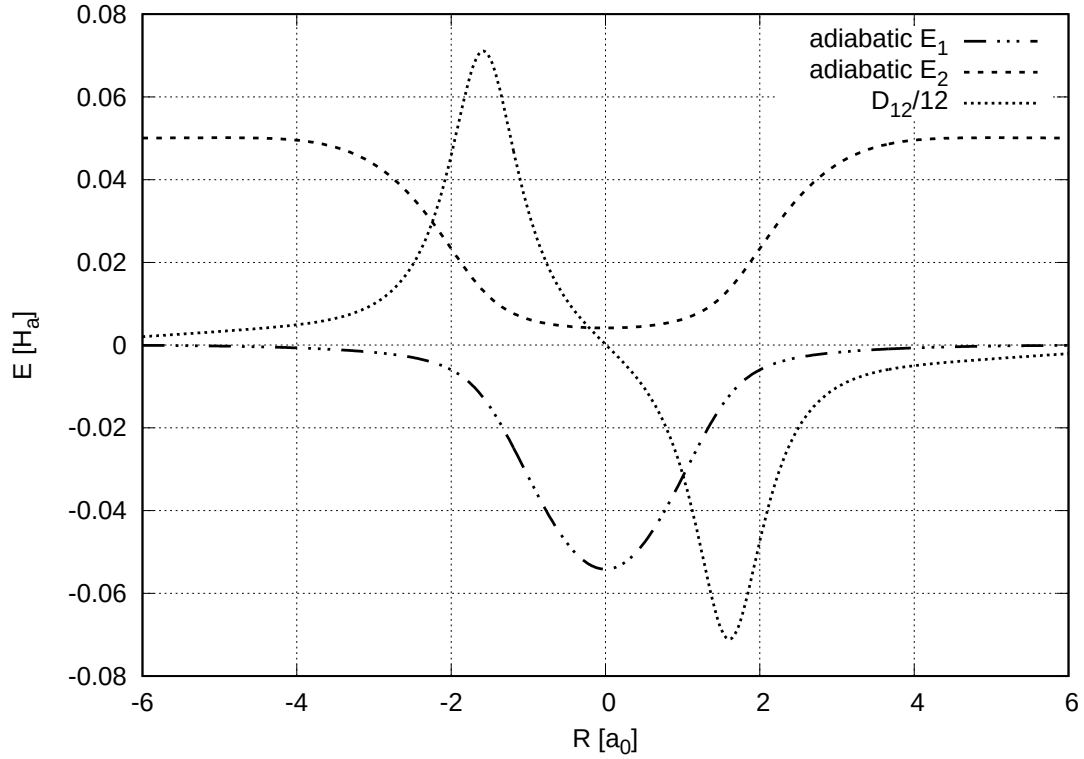


(b)

Figure 3.3: Transmission and reflection probabilities as function of time in simple avoided crossing; (a) high momentum Gaussian wavepacket $k = 15$ a.u.; (b) low momentum Gaussian wavepacket $k = 8.5$ a.u.

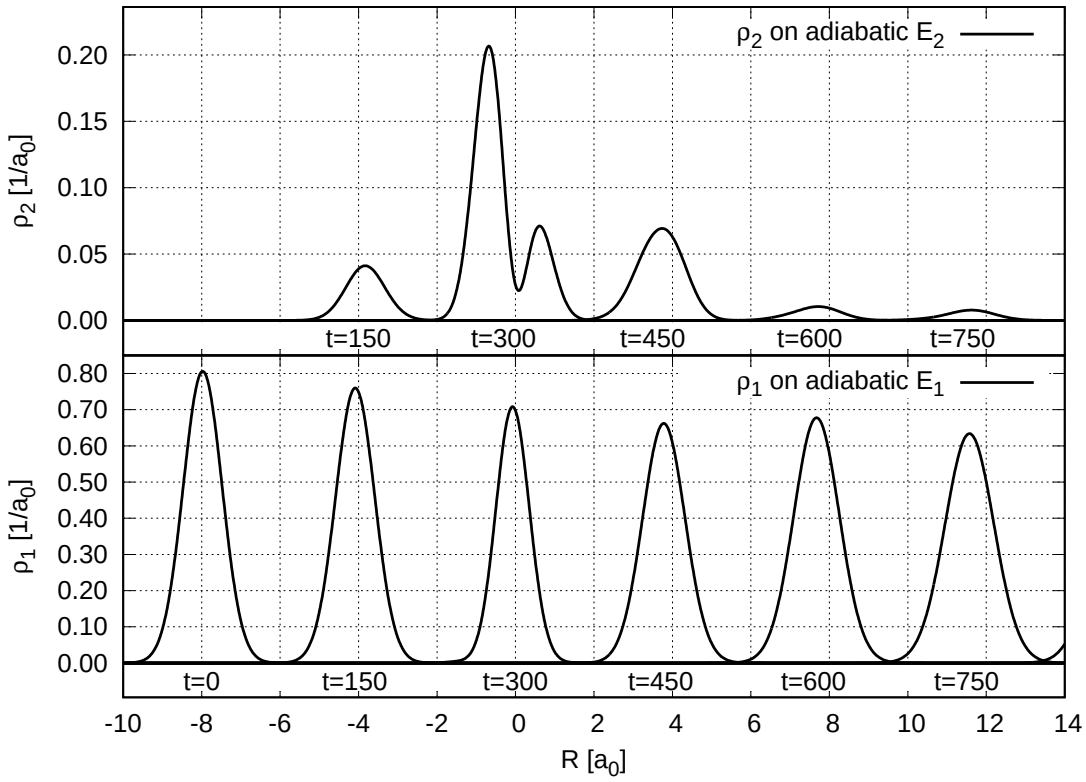


(a)

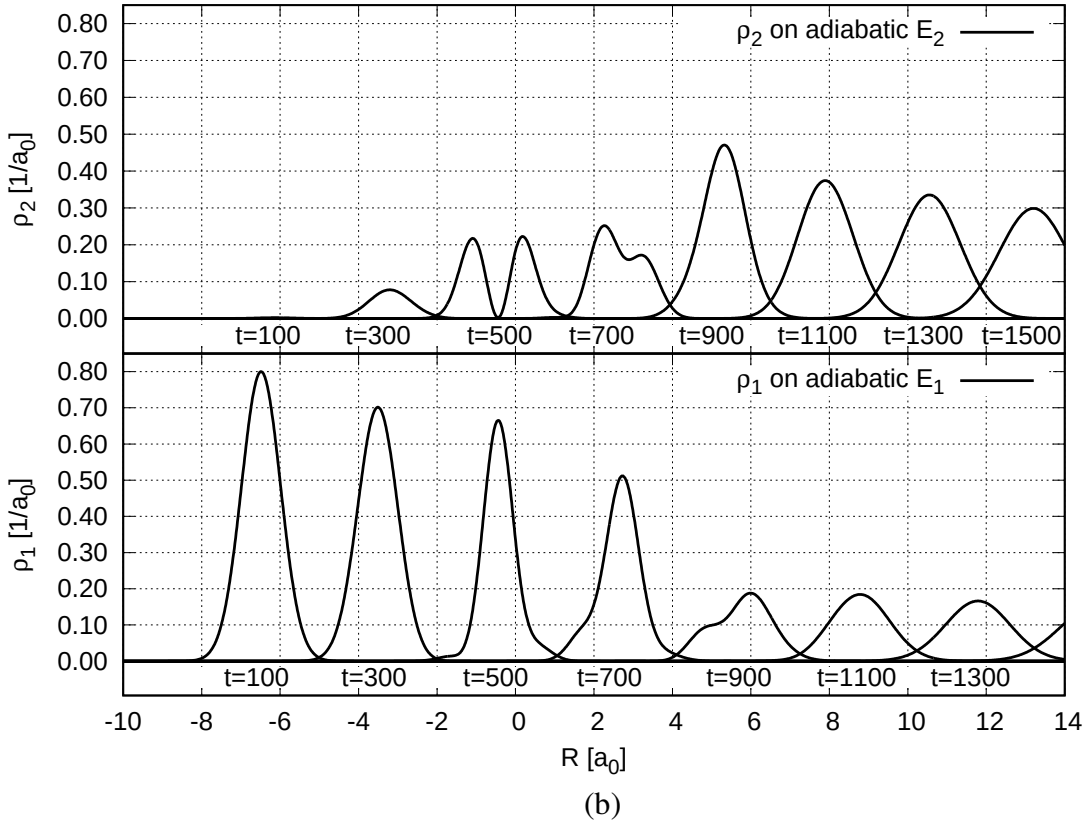


(b)

Figure 3.4: Model surfaces potential matrix of the dual avoided crossing example; (a) diabatic representation; (b) adiabatic representation, D_{12} is the nonadiabatic coupling element (it is drawn as divided by 12 because the coupling is large).

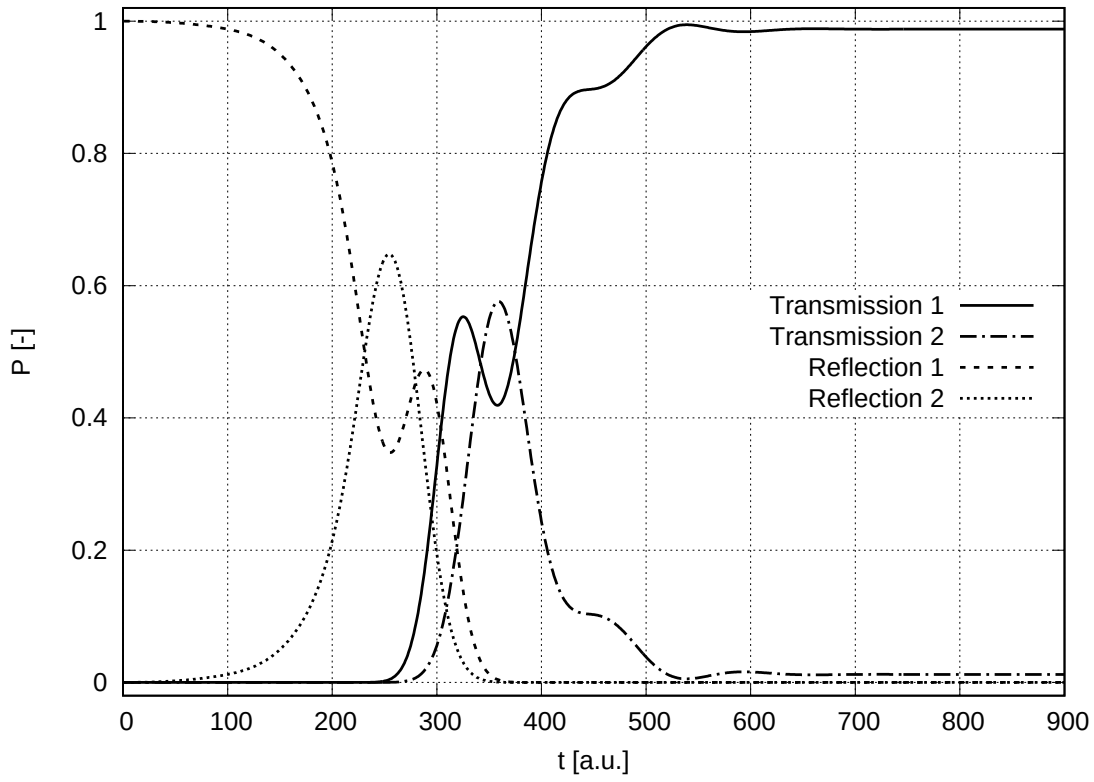


(a)

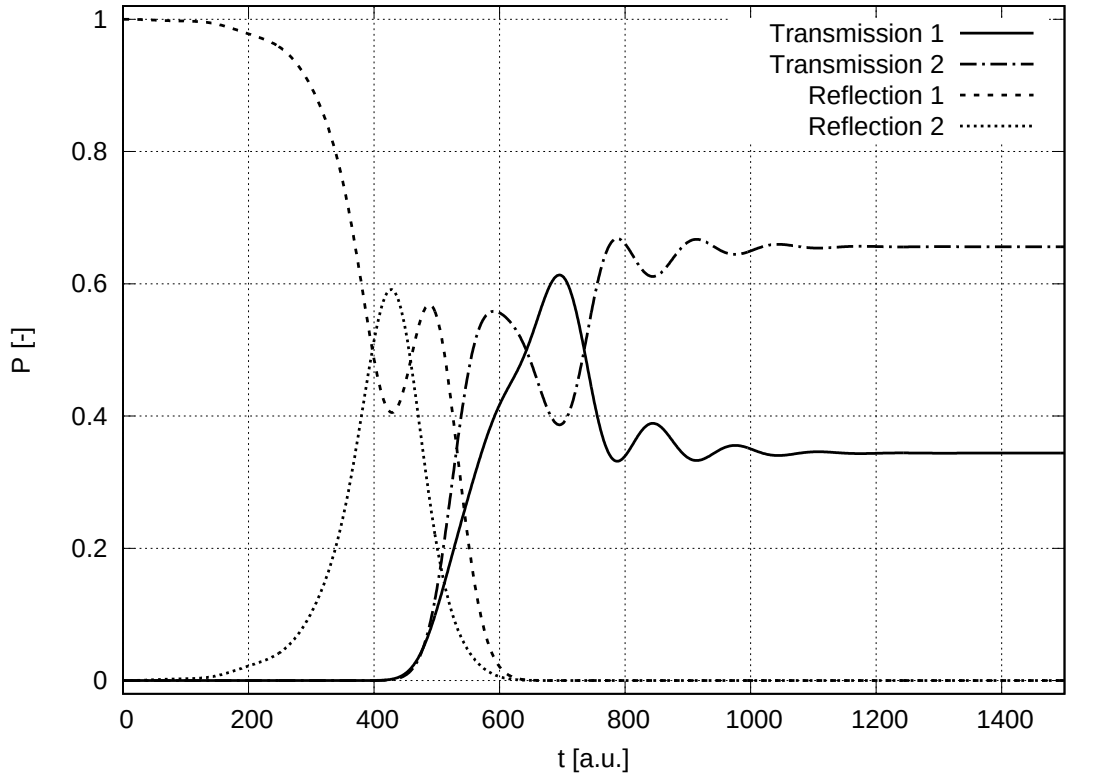


(b)

Figure 3.5: Time evolution of probability distributions on adiabatic energy surfaces, dual avoided crossing; (a) high momentum Gaussian wavepacket $k = 52$ a.u.; (b) low momentum Gaussian wavepacket $k = 30$ a.u.



(a)



(b)

Figure 3.6: Transmission and reflection probabilities as function of time in dual avoided crossing; (a) high momentum Gaussian wavepacket $k = 52$ a.u.; (b) low momentum Gaussian wavepacket $k = 30$ a.u.

diagonal potential is assumed. The adiabatic surfaces exhibit two avoided crossings (Fig. 3.4b) and the nonadiabatic coupling element D_{12} (Eq. 3.2) has two pronounced peaks.

The time evolution of probability distributions is shown on Fig. 3.5 and respective transmission and reflection probabilities are on Fig. 3.6.

The high momentum case in Fig. 3.5a demonstrates the effect of destructive interference between the first and second crossing. The wave packet populates the higher level around $t = 150$ (ρ_2 in Fig. 3.5a) then peaks at around $t = 350$ (transmission 2 in Fig. 3.6a) and arrives to second crossing at about the same phase at which it entered the higher level, but this time the nonadiabatic coupling element $D_{12}(R)$ (Fig. 3.4b) has negative sign. Same phase of packet in conjunction with negative sign of $D_{12}(R)$ causes the wave packet to leave the higher electronic surface almost completely, despite having high momentum. This effect can be seen in Fig. 3.6a, where in the end about 98% of population resides on the lower energy surface.

The low momentum case (Fig. 3.5b) shows constructive interference. First at around $t = 500$ about 32% of the population enters the higher energy surface (ρ_2 in Fig. 3.5b), then at second crossing additional 32% enters ρ_2. At $t = 900$ there is a significant population increase. Figure 3.6b shows that about 64% of the population was transmitted to the higher energy surface. This phenomenon is also known as Stükelberg oscillations and occurs when the time spent by the wave packet between two coupling regions is an integer or half integer multiple of mean wavepacket oscillations.

Again, the obtained results are in very good agreement with [110, 111].

3.3 Implementation of TDSE for time-dependent Hamiltonian

The C++ implementation of the iterative process in Equation 2.74 is shown in the Listing 3.4 of the function `SemiGlobalODE::propagateByDtSemiGlobal`.

First the simulation parameters (see Tab. 2.2) are assigned to local variables. A matrix `Unew` (variable $\vec{\tilde{\psi}}_{new}$ in Eq. 2.74) in line 10 is created to store M wavefunctions, one for each sub-timestep (Figure 2.9). Additionally one extra $M+1^{\text{th}}$ wavefunction is created for the purposes of tracking and estimating the calculation error of the function of the matrix (the K parameter) and the error of the sub-timestep interpolation (the M parameter). The self convergent iteration process (Eq. 2.74) spans whole Δt time period divided into M sub steps (Figure 2.9). The guess wavefunction `d.Ulast` in line 11 (variable $\vec{\tilde{\psi}}_{prev}$ in Equation 2.74) takes value from the extrapolated `Uguess` which was prepared at the end of the previous timestep (line 49)[21]. In case when there was no previous iteration the $\vec{\tilde{\psi}}(t=0)$ is used for all M sub timestep wavefunctions, this causes the main convergence `while` loop to be executed about 2 times more. The value $\vec{\tilde{\psi}}_{new}(\Delta t_0)$ at first sub-timestep is the final value from previous timestep (line 12). The $\vec{\tilde{\mathbf{v}}}_j$ vectors (Eq. 2.65) are created in line 13, their first component is just the wavefunction $\vec{\tilde{\psi}}_0$ (Eq. 2.64). In the main convergence loop they are calculated in line 20. Remaining preparation lines concern counting the number of iterations it took Eq. 2.74 to converge and tracking and estimating the calculation error. The `tol+1` in line 15 is to ensure that the first execution of the `while` loop always takes place. In next executions of this loop the value from Eq. 2.73 is used and compared against ε tolerance.

In the main `while` loop (line 18, Eq. 2.74), first the extended inhomogeneous source term $\vec{\tilde{\mathbf{s}}}_e(\vec{\tilde{\psi}}(t),t)$ from Eq. 2.70 is calculated in line 19 taking into account all of the time dependence of the Hamiltonian. Next the $\vec{\tilde{\mathbf{v}}}_j$ vectors are calculated for all M. The Newton interpolation polynomial (Section 2.2.5) is used and the divided differences calculations are performed in the process. This is followed by a check for numerical divergence (line 21). Next a lambda function for the $\widetilde{\mathbf{G}}_{avg}$ (line 22) is created to be used in the calculation of the first term $f_M(\widetilde{\mathbf{G}}_{avg})$ in Eq. 2.74. In line 26 the orthogonalized Krylov space vectors are stored in an upper triangular Hessenberg matrix and its eigenvalues are found (line 28). This is the place in the Arnoldi algorithm which finds the range of the energy spectrum of the Hamiltonian and makes it possible to calculate using complex potential which renders the Eq. 2.26 obsolete and simplifies a lot the usage of this algorithm. One additional point in the spectrum named `avgp` (line 30) is used in order to track the calculation error and compute the energy spectrum capacity [82] (line 34). Next the expansion vectors for the Newton approximation in the reduced Krylov space are calculated (line 36) and then all $M+1$ wavefunctions $\vec{\tilde{\psi}}_{new}$ are calculated (Eq. 2.74, line 39). It is this line that the conversion between Krylov space and position representation is performed using the transformation matrix Υ (c.f. point (d) in Section 2.2.6). This follows by estimating current convergence Eq. 2.73 and assigning $\vec{\tilde{\psi}}_{prev}$ to $\vec{\tilde{\psi}}_{new}$.

When the iterative process is complete the total number of iterations is stored (line 46) in order to track the computational cost. Next, the solution is stored in a class variable

[21] When jump starting the calculations `Uguess` equals the initial wavefunction (see Eq. 2.64). The assignment to `Uguess` is performed in the class constructor, hence it is not shown in the Listing 3.4

Code listing 3.4: The main propagation loop for the time-dependent Hamiltonian

```
1  void SemiGlobalODE::propagateByDtSemiGlobal(const int& iteration)
2  {
3      const int&  M   = cpar.M_Nt_ts;  // M param interior time points
4      const int&  K   = cpar.K_Nfm;    // K param function of matrix
5      const Real& tol = cpar.tol;      // tolerance parameter
6      const auto& N   = cpar.Nu;       // number of grid points
7      LocalData   d(*this, iteration); // local variables Eqs.3.61,3.65
8      // Unew is the new wavefunction in all M sub-timesteps
9      // The extra M+1 is used for estimating errors
10     Unew            = MatrixXcr::Zero(N, M + 1);
11     d.Ulast         = Uguess; // first guess for Eq.3.74
12     Unew.col(0)     = d.Ulast.col(0);
13     d.v_vecs.col(0) = d.Ulast.col(0); // Eq.3.64 and 3.65
14     niter           = 0; // count iterations in Eq.3.74
15     currentErrors   = EstimatedErrors(0, tol + 1); // error tracking
16
17     // main iteration loop in Eq.3.74
18     while (not d.converged(currentErrors, tol)) { // Eq.3.73
19         d.calcSExtended(); // calculate Eq. 3.70, inhomogeneos term
20         d.calcVVectors();  // calculate Eq. 3.64 and 3.65
21         d.checkAllFinite(currentErrors); // check for divergence
22         auto Gop_avg = [=](const VectorXcr& v) -> VectorXcr {
23             return Gop(d.Ulast.col(cpar.tmidi), d.t(cpar.tmidi), v);
24         };
25         // Create the orthogonalized Krylov space, Section 3.2.6
26         const MatrixXcr Hessenberg
27             = createKrop(Gop_avg, d.v_vecs.col(M), K, d.Upsilon);
28         const VectorXcr eigval
29             = eigenValues(Hessenberg.block(0, 0, K, K));
30         const Complex avgp = eigval.sum() / Real(K);
31         // sampled energy spectrum and extra point to estimate error
32         d.samplingp.resize(eigval.rows() + 1);
33         d.samplingp << eigval, avgp;
34         d.capacity = get_capacity(eigval, avgp);
35         // Obtain the expansion vectors for Newton approximation
36         d.RvKr = getRv(
37             K, d.v_vecs, M, Hessenberg, d.samplingp, d.capacity);
38         // use Arnoldi to approximate new iteration of Eq.3.74
39         Unew.block(0, 1, Unew.rows(), Unew.cols() - 1)
40             = Arnoldi(cpar.timeMts, M, tol, K, d, currentErrors);
41         d.estimateErrors(currentErrors);
42         d.Ulast = Unew;
43         niter   = niter + 1;
44     }
45     if (iteration == 0) niter0th = niter;
46     allniter      = allniter + niter;
47     fiSolution    = Unew.col(M - 1); // store the result
48     Uguess.col(0) = Unew.col(M - 1); // extrapolate for next timestep
49     Uguess.block(0, 1, Uguess.rows(), M)
50         = Arnoldi(cpar.timeMnext, M, tol, K, d);
51     if (there_is_ih) { sNext = d.s.col(M - 1); }
52     d.checkErr(currentErrors, maxErrors); // check estimated errors
53 }
```

Code listing 3.5: The time-dependent Hamiltonian operator for coupled electronic states in C++

```
1  MultiVectorXcr
2  State::calc_Hpsi(const MultiVectorXcr& psi_0, const Real& t)
3  {
4      MultiVectorXcr psi_ret = getZeroMultiVectorXcr(psi_0);
5      // The kinetic operator, acting only on the diagonal
6      for (int j = 0; j < levels; j++) {
7          psi_ret[j] = Ekin_single(psi_0[j]);
8      }
9      // The potential operator matrix acting on the wavefunction
10     for (int j = 0; j < levels; j++) {
11         for (int k = 0; k < levels; k++) {
12             // if present obtain the time dependent potential
13             if (hasTimeDependence(j, k)) {
14                 psi_ret[j] = psi_ret[j].array()
15                     + timeDependentPotential(j, k, t).array()
16                         * psi_0[k].array();
17             } else {
18                 psi_ret[j] = psi_ret[j].array()
19                     + potentialMatrix[j][k].array()*psi_0[k].array();
20             }
21         }
22     }
23     return psi_ret;
24 }
```

Code listing 3.6: The C++ wrapper for handling arbitrary number of coupled electronic states

```
1  VectorXcr State::minus_i_Hpsi__MultiVectorFlattened(
2      const VectorXcr& /*u*/, // wf from this step, e.g. Bose-Einstein
3      const Real&      t,     // time
4      const VectorXcr& v      // wavefunction wf to propagate
5  )
6  {
7      MultiVectorXcr psi_work = zero(wf);
8      decompres(v, psi_work);
9      psi_work = calc_Hpsi(psi_work, t);
10     return -Mathr::I * flatten(psi_work);
11 };
```

`fiSolution` (line 47). Finally the wavefunction is extrapolated for the next timestep Δt (line 49) and the estimated errors are checked and stored (line 52).

The time-dependent Hamiltonian $\widetilde{\mathbf{H}}(t)$ is called in lines 19 and 23, when invoking the `calcSExtended` and `Gop` functions. It is presented in Listing 3.5. The differences between $\widetilde{\mathbf{H}}(t)$ and the time-independent counterpart $\widetilde{\mathbf{H}} \neq \widetilde{\mathbf{H}}(t)$ (Listing 3.2) are small but significant. First it is not the normalized Hamiltonian $\widetilde{\mathbf{H}}_{norm}$ (like it was used in the time-independent case), second it has encoded time dependence in line 12 of Listing 3.5, third it deals with coupled electronic states (Section 3.3.1). The time-dependent potential is used in line 15 when acting on the wavefunction in line 16. The time-dependent Hamiltonian uses the same function `Ekin_single` (Listing 3.3) in line 7 (Listing 3.5) to calculate the kinetic energy operator although if necessary (if for example weak interactions are planned to be simulated) it also could become time-dependent with only a small change in the code.

In here, the numerical damping absorbing boundary conditions Section 2.2.9 are encoded inside the complex potential (lines 15 and 19 in Listing 3.5) during the calculations.

3.3.1 Coupled time-dependent Schrödinger equations

In Section 2.2.7 I explained how it is possible to adapt this algorithm to arbitrary amount of coupled electronic states by storing all levels inside a single table.

To deal with each k^{th} level the `Ekin_single` loops over all `levels` (Listing 3.5 line 6). Similarly when dealing with the off diagonal elements of Eq. 2.2 the loop on `levels` is done in lines 10 and 11. The function `calc_Hpsi` in Listing 3.5 as the wavefunction argument takes the `MultiVectorXcr` which contains all wavefunctions as separate elements of `std::vector`[22]. But the semi-global algorithm deals with a single „flattened" `VectorXcr`. Hence a conversion discussed in Section 2.2.7 has to be performed.

This conversion is shown in Listing 3.6. The `minus_i_Hpsi__MultiVectorFlattened` is the function which is provided to the semi-global algorithm as the function $\widetilde{\mathbf{G}}(\vec{\widetilde{\psi}}(t),t)$ (Eq. 2.51, Listing 3.4 line 23). It is called with wavefunction stored inside a single argument `VectorXcr`. This data is decompressed in line 8 (Listing 3.6), then the time-dependent Hamiltonian `calc_Hpsi` is called on it (Listing 3.5) and then the data is flattened again into a single `VectorXcr` and multiplied by negative imaginary unit (Eq. 2.51, line 10 in Listing 3.6) (atomic units are used here). The semi-global algorithm can also work when the wavefunction from present timestep is used (e.g. a Bose-Einstein condensate trap) and provides this argument for the Hamiltonian in line 2 of Listing 3.6. The wavefunction from present timestep is the `d.Ulast.col(cpar.tmidi)` (line 23 in Listing 3.4) where `cpar.tmidi` is the time coordinate of the averaged Hamiltonian $\widetilde{\mathbf{G}}_{avg}$[23].

3.4 Validation of time-dependent Hamiltonian using an atom in an intense laser field

To validate the implementation of the semi-global time propagator I have run following tests and reproduced the results both from [41] and from the previous Section 3.2:

(a) Atom in an intense laser field [41].

(b) Single avoided crossing, Section 3.2.1.

(c) Dual avoided crossing, Section 3.2.2.

(d) Gaussian packet in a forced harmonic oscillator, supplementary materials of [41].

(e) Forced harmonic oscillator with an arbitrary inhomogeneous source term, supplementary materials of [41].

The single and dual avoided crossing are the same tests as in the previous Section 3.2, because an algorithm for time-dependent Hamiltonian has to work the same with time-independent Hamiltonian. And I have reproduced these results exactly using the time-dependent Hamiltonian. The tests with a Gaussian packet in a forced harmonic oscillator and an oscillator with an inhomogeneous source term are example simulations found in the

[22] See footnote[15] on page 29 for details.

[23] In Listing 3.6 the variable `/*u*/` is commented out because in this chapter I am not dealing with the Bose-Einstein condensate. It is confirmed to work by comparing with examples provided in [41].

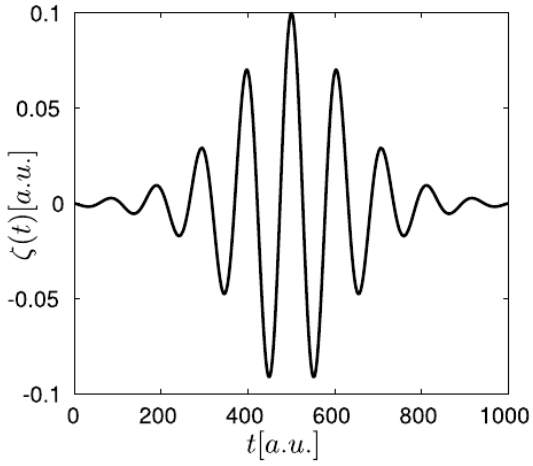


Figure 3.7: The electric field of the laser impulse [41].

supplementary materials of [41] and also I have reproduced them exactly. Here I will only present the reproduction of results of a model atom in an intense laser field which was presented in [41]. I shall note that these are not new results, since validation of an algorithm has to be run on an example for which the results are already known and verified. In this case I am comparing the electronic wavefunction of an atom in a laser field after evolution for 1000 a.u. (a time of about 24.2 fs), with the reference result found in the supplementary materials of [41]. In the calculations I have used the following parameters (Tab. 2.2): $K = 9$, $M = 9$, $\varepsilon = 2 \times 10^{-16}$ and $\Delta t = 0.025$ a.u.

In this test the central potential is represented by a simplified Coulomb potential without the singularity (hence it is a model one dimensional atom):

$$V_{atom}(x) = 1 - \frac{1}{\sqrt{x^2+1}}. \tag{3.8}$$

This simple model is for example used in the context of intense laser atomic physics. The electric field of the laser impulse used is following (Fig. 3.7):

$$\zeta(t) = 0.1\mathrm{sech}^2\left(\frac{t-500}{170}\right)\cos(0.06(t-500)). \tag{3.9}$$

This laser impulse with $\omega = 0.06$ a.u. is similar to the wavelength of a Titanium-Sapphire laser which is $\lambda = 760$ nm. The $\mathrm{sech}^2 = 1/\cosh^2$ envelope is similar to the actual envelope found in the laser pulses. The maximum aplitude $\zeta_{max} = 0.1$ a.u. represents the intensity of about $I_{max} = 3.52 \times 10^{14}$ W/cm^2. This term, together with the dipole approximation $x\zeta(t)$ is added to the Hamiltonian which reads:

$$\hat{\mathbf{H}}(t) = \frac{-\hbar^2}{2m_e}\nabla^2 + 1 - \frac{1}{\sqrt{x^2+1}} - x\zeta(t), \tag{3.10}$$

where the mass of electron $m_e = 1$ a.u. For the purposes of numerical simulation this Hamiltonian is modified by adding complex absorbing boundary conditions (Section 2.2.9). Their effect can be seen on the left and right edges of Fig. 3.8 where the wavefunction is diminished. The initial wavefunction is the ground state of the model atom.

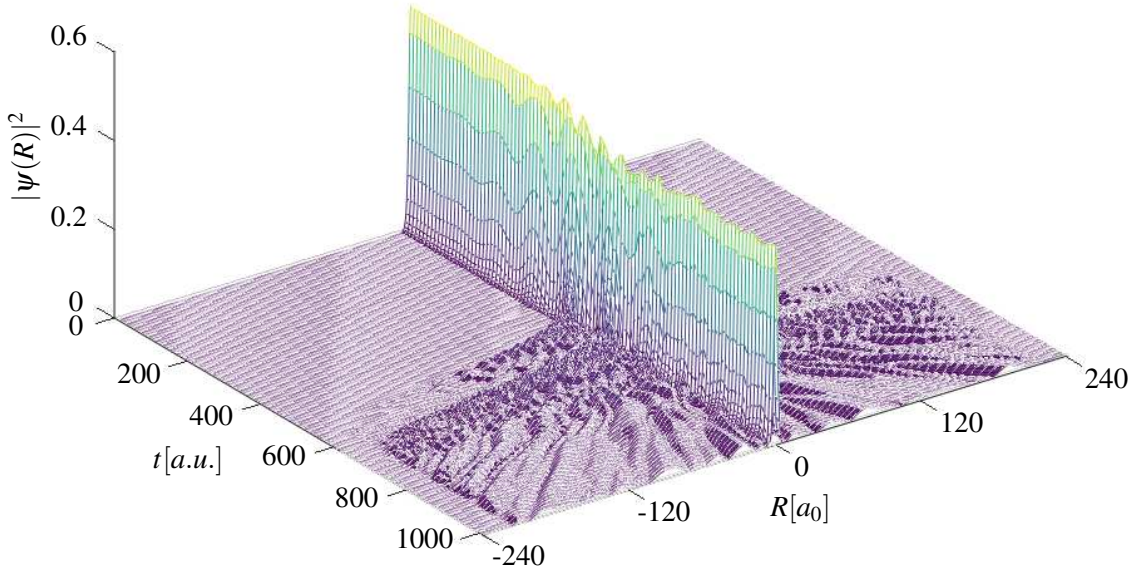


Figure 3.8: Model atom in an intense laser field, the time evolution of $|\psi(R)|^2$ during 1000 a.u. The wavefunction at 1000 a.u. agrees with reference solution in [41] with error $< 8 \times 10^{-15}$.

Figure 3.8 shows the time evolution of the wavefunction over the time of 1000 a.u. The maximum intensity of the laser impulse is centered on 500 a.u. (Fig. 3.7) and it can be seen on Fig. 3.8 that this is when the wavefunction starts to undergo a rapid change and some parts of the wavefunction start the process of dissociation. When the calculations reach the 1000 a.u. point in time I compare the results with the reference solution found in the supplementary materials in [41] and the maximum difference is smaller than 8×10^{-15}, well within the range of the numerical ULP error[25]. It means that my implementation of the semi-global algorithm completely reproduces the reference results.

3.5 Benchmarks of high precision quantum dynamics

As mentioned in Section 2.3, I would like to emphasize that this algorithm is implemented in C++ for arbitrary floating point precision types, specified during compilation. It works just as well for types with 15, 18 or 33 decimal places. This is the reason why `Real` type instead of `double` is used e.g. in line 5 in Listing 3.4. The `Real` type is set during compilation to one of the following types: `double`, `long double` or `float128`. The list of all high precision types, the number of decimal places and their speed relative to `double` is given in Table 3.2. It follows my work on high precision in YADE, a software for classical dynamics calculations [28]. However the arbitrary precision types `boost mpfr` and `boost cpp_bin_float` are currently not available in the quantum dynamics code because the Fast Fourier Transform (FFT) routines[24] for them are currently unavailable in the Boost libraries [60]. To resolve this problem I took part in the Google Summer of Code 2021 [62] as a mentor and now the high precision FFT code is in preparations to be included in the Boost libraries. The report from high precision FFT implementation in Boost is available here [61]. After that work is complete all types listed in Table 3.2 will be available for quantum dynamics calculations.

[24] used in Listing 3.3 on page 31.

Table 3.2: The high-precision benchmark: comparison between classical dynamics [28] and quantum dynamics. The speed is shown as relative to speed at `double` precision

Type	Decimal places	Classical dynamics speed w.r.t `double`	Quantum dynamics preliminary benchmark‡
`float`	6	1.01× faster	
`double`	15	—	—, $\varepsilon = 2 \times 10^{-16}$
`long double`	18	1.4× slower	3.5× slower, $\varepsilon = 1 \times 10^{-19}$
`boost float128`	33	4.7× slower	50× slower, $\varepsilon = 2 \times 10^{-16}$
`boost float128`	33		170× slower, $\varepsilon = 2 \times 10^{-34}$
`boost mpfr`†	62	13.5× slower	
`boost mpfr`	150	19.1× slower	
`boost cpp_bin_float`	62	24.2× slower	

† for future comparison with `libqd-dev` library.
‡ assuming ε=ULP error[25] and not changing other parameters (Equation 2.73 and Table 2.2).

Table 3.2 shows the speed comparison between different high-precision types, relative to `double`, separately for classical dynamics and quantum dynamics. The classical dynamics are reproduced from YADE benchmark [28] to serve as a reference. I have done the quantum dynamics benchmark using the example of atom in an intense laser field from previous Section 3.4. I used the same parameters with the exception of changing ε (Eq. 2.73) to match the ULP error[25] of selected precision as provided in table.

The `long double` precision in quantum dynamics is about 3.5× slower and in exchange provides about 2000 times greater precision. This might be useful in some situations for quick verification of results.

For the `float128` type I did the benchmark twice, first for the ε same as for `double` type, then for the significantly smaller ε matching the `float128` type. We can see that using $\varepsilon = 2 \times 10^{-16}$ from `double` for the `float128` makes it about 50× slower[26]. When using full `float128` precision then the decreased error tolerance ε forces more iterations in Eq. 2.74 consequently making it 170× slower. If one wished to calculate with larger tolerance, say $\varepsilon = 1 \times 10^{-25}$, then still `float128` has to be used, and it will have speed somewhere between the two values in Tab. 3.2.

I would like to mention that from my experience, it is better to increase Δt and allow more sub-iterations in Equation 2.74 as it speeds up calculation more than changing the K and M parameters. For example this atom in the laser field calculation took 30 seconds with `double` and $\Delta t = 0.1$ a.u. and 10 minutes with `double` and $\Delta t = 0.0041(6)$ a.u. both having the same $\varepsilon = 2 \times 10^{-16}$ error tolerance.

Moreover, the original code [41], which deals with a single Schrödinger equation, was written in Matlab [112] and the same simulation which took me 30 seconds in C++, took about 5 minutes in Octave[27]. So the 10× speed gain due to migrating from Octave to C++ can now be wisely spent on higher precision calculations or on simulating larger systems.

[25] see footnote[7] on page 6 for details.
[26] Not 4.7× slower like in classical dynamics, because the semi-global algorithm is more matrix oriented than classical dynamics calculations and uses FFT. It is however performing the same amount of mathematical operations as the `double` type.
[27] Octave is an open-source version of Matlab [112]. I have no access to commercial Matlab software. Single core was used in both cases, C++ and Octave.

Chapter 4

Results

In this chapter I use the numerical methods described in previous chapters to analyze the quantum dynamics of KLi and NaRb dimers and compare them with available experiments. It is divided into two main parts.

In the first part (Section 4.1, published in [29]) I simulate the tunneling dissociation and exponential decay in the KLi dimer using a time-independent Hamiltonian. Hence I use the work presented in Section 2.1 and Section 3.1 which deals with the time-independent Hamiltonian.

In the second part (Section 4.2, presented on a poster in [30] and in preparations to publish) I simulate the power-law decay of the NaRb dimer subject to the influence of a femtosecond impulse using the time-dependent Hamiltonian with three coupled electronic states as described in Sections 2.2 and 3.3.

Since both of those parts deal with a very different molecular system each of them has its own conclusion subsection.

4.1 Dissociation in KLi

The very good agreement between the calculated [29] and the experimental [113, 114] curve of the $2^1\Pi$ state[28] allows for a reliable description of the dissociation process through a small (~ 20 cm^{-1} for $J = 0$) potential energy barrier, which is present in the $2^1\Pi$ state. The barrier supports several rovibrational quasi-bound states and explicit time evolution of these states via the time-dependent nuclear Schrödinger equation shows that the state populations decay exponentially in time. I precisely describe the time-dependent dissociation process of several rovibrational levels and show that the calculated spectrum matches very well with the assigned experimental spectrum. Moreover, my approach is able to predict the positions of previously unassigned lines, particularly in the case of their low intensity.

Studies of polar alkali dimers provide valuable insight into several basic phenomena, such as perturbations in excited states, potential curve crossings and avoided crossings, photodissociation, photoassociation, and new quantum matter, namely the Bose-Einstein condensate and ultracold two-fermionic species [115]. The development of spectroscopic methods allows for very accurate measurements, providing valuable data for the ground and excited states of these molecules. From the theoretical point of view heteronuclear alkali metal dimers are very attractive objects due to their simple electronic structure and the possibility

[28] The atomic asymptote of $2^1\Pi$ correlates for the large internuclear distance with the K(4s) + Li(2p) state.

of treating them as effective two-electron systems with separated atomic cores. Pseudopotential methods with longtail core polarization model potentials are well suited to treat such systems [116, 117].

For years, the KLi dimer was an object of spectroscopic experiments [118, 119, 120, 121, 122, 123] with the notable series of studies by the Warsaw group [114, 124, 125, 126, 127, 128] which provided, using Doppler-free polarization labeling technique, molecular constants and potential energy curves for the ground and several excited singlet states up to $11^1\Pi$ lying 33,000 cm^{-1} above the minimum of the ground state. Some potential energy curves, like in the case of the $1^3\Sigma^+$ state [126], were determined experimentally despite limited data sets of vibrational energies. The KLi dimer was also studied in ultracold atomic and molecular gases with strongly interacting two-fermionic species consisting of ^{40}K and ^{6}Li atoms [129, 130, 131].

Already in 1984, Müller and Meyer [132] performed extensive all-electron SCF calculations on KLi along with a careful treatment of intershell effects. In 1999, Rousseau et al. [133] performed CI calculations with nonempirical one-electron pseudopotentials and appropriate polarization potentials by means of the CIPSI program package [134] obtaining 58 electronic states of KLi. Recently extensive theoretical ab initio studies were performed, including the electronic structure, transition dipole moments [135], and effect of inner-shell electrons on the molecular properties [136]. Additionally the low-lying potential energy curves, transition dipole moments and Franck-Condon factors were calculated in order to show effective schemes of photoassociation reactions [137]. Also the electronic structure of KLi has been investigated by treating the K and Li atoms with the non-empirical relativistic effective core potentials [138]. Very recently, the multireference coupled cluster method (MRCC) has been used to calculate 10 low-lying states of KLi [139].

In this section, I address the problem of direct bound-to-free simulations, particularly for quasi-bound rovibrational levels lying just below the potential barrier. This is the energy region in which the good agreement between semi-classical and fully quantum results is questionable (the difference between them is discussed below, Fig. 4.2). I study the validity of the quantum bound-to-free approach near the potential barrier where quasi-bound levels are broad and their low intensity makes an additional challenge for experimentalists. Here I present the lines of partial cross section for predissociation along with time-dependent populations. Initial quasi-bound wavepackets with precisely assigned rotational and vibrational quantum numbers allow to calculate the time-dependent population of rovibrational levels. I shall note here that using the eigenfunctions is the only approach to obtain population decay curves specific for a given rovibrational level. In comparison using an initial Gaussian wavepacket would not be effective since it usually covers many interfering vibrational levels, each of them with different population decay constant. To perform any time-dependent (dynamical) calculations it is preferable to have potential energy curves of high accuracy, this is essential in description of the predissociation process. Thus the potential energy curve for the $2^1\Pi$ state from [29] is used. This allows to get a new insight into the dynamics of the rovibrational predissociation process. Details of this process showing the role of shapes of initial wavepackets in forming the spectral lines is shown step by step in the molecular movies available in [140]. To that end I show explicitly that the quasi-bound state populations decay exponentially in time. Moreover, I present here a systematic study of energies and widths of quasi-bound levels, which provides a new valuable data for finding effective photoassociation schemes leading to Bose-Einstein condensates. Whenever it is possible I carefully compare the results with the available experimental data to get a better insight into

the nature of the predissociation process.

The application of Section 2.1 to the problem at hand is explained in the next section putting emphasis on the time-dependent method and boundary conditions. In Section 4.1.2, I present the results for the $2^1\Pi$ state and compare them with the experimental data together with the spectrum and the population evolution. Conclusions are given in the Section 4.1.3.

4.1.1 Computation method

The time-dependent approach (Section 2.1) can be regarded as a complementary tool to the time-independent method and is often used in studying photodissociation processes. Here, it serves as an alternative and quite illustrative method for comparing with the structural calculation results obtained via semi-classical time-independent method in program LEVEL [141].

Let us consider the time-dependent Schrödinger equation in the following form:

$$\imath\hbar\frac{\partial}{\partial t}\Phi(R,t) = H_J^{nuc}\Phi(R,t), \tag{4.1}$$

where $\Phi(R,t)$ is the time-dependent wavepacket moving on the effective potential energy curve $U_J(R)$ and the nuclear Hamiltonian is taken as $H_J^{nuc} = -\frac{\hbar^2}{2\mu}\frac{\partial^2}{\partial R^2} + U_J(R)$. The adiabatic potential energy curve $U_J(R)$ is the $2^1\Pi$ electronic state from [29].

The evolving wavepacket $\Phi(R,t)$ is a solution of Eq. 4.1. As the initial wavefunction, $\Phi(R,t=0) = \Psi_{E,J}(R)$, I select a quasi-bound eigenstate which can be calculated very accurately by the methods implemented in the program LEVEL [141]. This approach allows to calculate the population for the particular state labeled by the vibrational and rotational quantum numbers (v,J). The program LEVEL calculates the state only up to the R coordinate equal to the outermost classical turning point. To use it in the quantum dynamic calculations I needed to extrapolate the tail of the provided wavefunction using exponential decay. This extrapolated part of the wavefunction only serves the purpose to ensure that the first and second derivatives of the wavefunction are smooth. Apart from that they bear very small influence on the calculation results, since the wavefunction in this extrapolated region is quickly filled with larger parts of the wavefunction which are tunneling in the predissociation process.

The wavepacket is tunneling away from its starting position with the main amplitude located inside the potential energy barrier. The time-dependent population $P(t)$ is calculated in the range from $R = 0$ till R_{max}, where R_{max} is chosen sufficiently to the right of the outermost classical turning point such that $P(t=0) = 1$. The integration of population excludes the ending region in which the absorbing boundary conditions are in effect (see Section 2.1.6)[29]. Hence the population is calculated as

$$P(t) = \int_0^{R_{max}} |\Phi(R,t)|^2 dR. \tag{4.2}$$

The process is also described by the time-dependent autocorrelation function

$$S(t) = \int \Phi^\star(R,t=0)\,\Phi(R,t)\,dR. \tag{4.3}$$

[29] As mentioned earlier, the calculations in this section are done with a time-independent Hamiltonian, a method from Section 2.1

In this case the autocorrelation function describes the evolution of the initial nuclear eigenfunction in the excited electronic state.

The spectrum is determined by the inverse Fourier transform of $S(t)$ [38] as follows

$$\sigma(E(v,J)) = \int_{-\infty}^{\infty} \mathrm{e}^{iE(v,J)t/\hbar} S(t)\, dt. \tag{4.4}$$

I am estimating the above integral over the time range $\langle 0, T \rangle$ (where $T = 1$ ns) using the fast Fourier transform (FFT) routines [142].

I have implemented the code for calculating the quantum dynamics from Section 3.1 [63, 73, 143] into an unmerged[30] branch of YADE computational framework which I developed [28, 33, 34, 35]. In this approach the Chebyshev polynomial recurrence relation formula [63, 72] is modified by multiplying it by the absorbing boundary conditions term $e^{-\gamma(R)}$ (Section 2.1.6 [74, 75]). The resulting absorbing potential acts as the commonly used imaginary absorbing potential (see Section 2.2.9 for time-dependent Hamiltonian), such that the calculations are numerically stable in the Kosloff calculation method.

In the calculations of the autocorrelation function (Eq. 4.3), the propagation time is 1 ns which is sufficient for estimation of the integral in Eq. 4.4. In Equation 4.2, I use the value of R_{max} equal to 57 a_0 (30 Å). There are 8192 points in the whole grid and 4669 points in the integration grid (since integration excludes the region of the absorbing boundary conditions). In order to avoid the interference between the outgoing and incoming waves on the periodic grid an absorbing potential is placed in the range from 57 to 100 a_0 (30 Å to 53 Å). This potential smoothly absorbs the wavepacket (Section 2.1.6) [74].

The obtained population curve $P(t)$ is roughly constant in the first 5 ps (see Fig. 4.1A), which is the time for the wavepacket to reach the R_{max} point. After this the population (Eq. 4.2) starts to decay. It is seen that the population follows an exponential decay $e^{-t/\tau}$, with a decay constant τ. Hence here I demonstrate the exponential decay of population based on explicit time-propagation of a quasi-bound state. Often the exponential decay of such states is described by introducing an imaginary part in the state energies [38, 144]. An exponential decay in population leads by using Eq. 4.4 to Lorentzian line shapes with full width at half maximum (FWHM) equal to $\Gamma = \hbar/\tau$ [38, 144].

The time t_0 for the best determination of the decay constant τ (and correspondingly Γ) is determined by performing a parametric least squares fitting, where the t_0 is adjusted in order to maximize the Pearson's correlation coefficient $r_{t_0,\Gamma}$ value of the fit (see Fig. 4.1B). The dependence of Γ on t_0 (at which the least squares fitting starts) is shown on Fig. 4.1C. It can be seen that a "perfect" fit of Γ is found in a very narrow range (see inset of Fig. 4.1C) in the order of 0.02 cm^{-1}. When scanning the entire time range $t_0 \in$[5 ps,1 ns], the least squares fitting had a tendency to produce "perfect" fits $r_{t_0,\Gamma} = 1.0$ at very large t_0 (of the order of 0.1 ns), hence to force inclusion of the beginning of the population decay (which starts around 5 ps) the fitting is constrained to start before $t_0 < 15$ ps.

This method allows to calculate the FWHM spectral peak widths Γ values for vibrational and rotational quantum numbers shown in Fig. 4.2. For each of these levels I am fitting the population decay curve $e^{-t/\tau}$ obtained from dynamical evolution of the eigenfunctions, and then convert it with the formula for $\Gamma = \hbar/\tau$ [38, 144] to the peak width. The comparison with results from program LEVEL [141] in Fig. 4.2 is discussed in the next section.

[30] See also the footnotes on page 1.

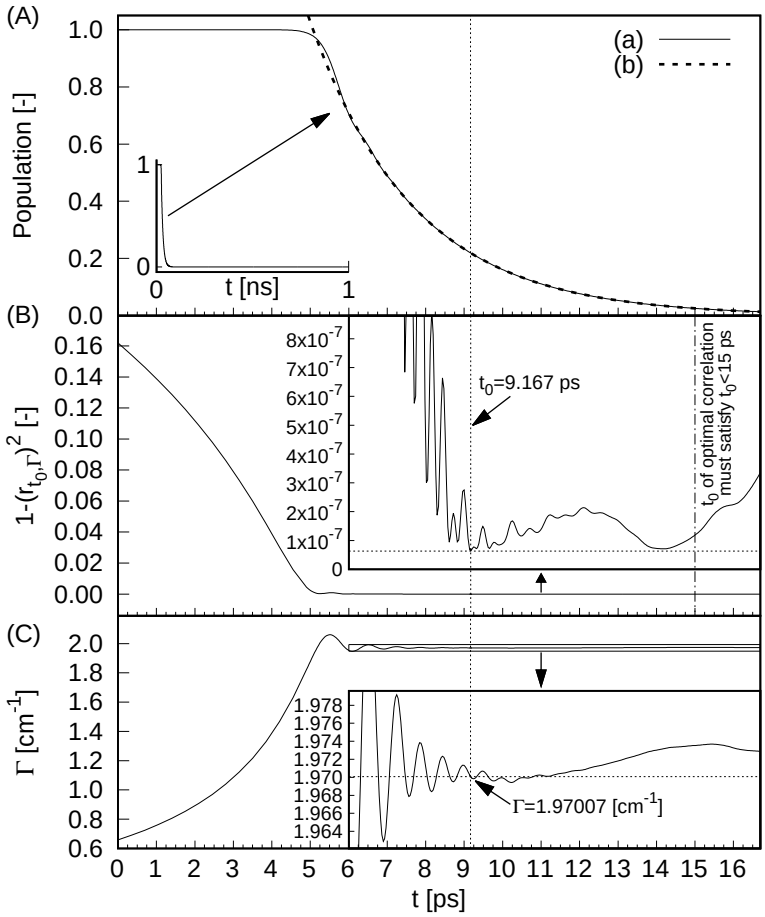


Figure 4.1: The procedure of finding optimal FWHM Γ [cm^{-1}] shown for the level $J = 37$ and $v = 16$ of the $2^1\Pi$ state. (Aa) Population of time–evolving wavepacket is calculated up to 1 ns, but only the first 17 ps are shown here for clarity. (Ab) The obtained best fit population decay function $e^{-t/\tau}$, the population values to the left of t_0 are not used in the fit. (B) The value of Pearson's correlation coefficient $r_{t_0,\Gamma}$ (plotted as $1-(r_{t_0,\Gamma})^2$) as a function of time t_0 where the least squares fitting starts. (C) The value of FWHM Γ [cm^{-1}] obtained from least squares fitting starting at each t_0.

4.1.2 Results and discussion

Using the computer code for solving the time-dependent Schrödinger equation with the time-independent Hamiltonian (Section 2.1 and Section 3.1, [28, 33, 34, 35, 63, 73, 143]), I present several interesting features of the spectra and where it is possible I compare them with the other theoretical results and experimental data. Figure 4.2 shows comparison of my results of the level widths Γ (e.g. the full width at half maximum, FWHM) with those calculated from the program LEVEL [141]. My line widths are calculated directly from the exponential decay of population. The extracted decay constant is used to estimate the width $\Gamma = \hbar/\tau$. The program LEVEL calculates the line width from a uniform semi-classical treatment which assumes that at the outermost classical turning point the wavefunction behaves like the Airy function of the second kind. It can be seen that the semi-classical treatment (crosses in Fig. 4.2) overestimates the line widths for the broadest levels. Thus, the differences between the results become significant when the energy of quasi-bound levels approaches the top of the barrier. It is known that in this energy regime, the semi-classical approach does give less reliable results [141].

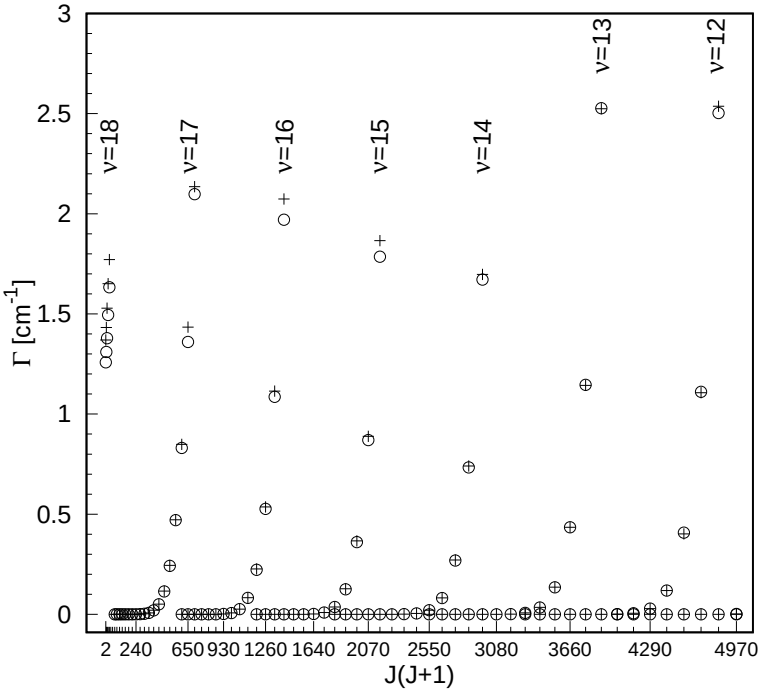


Figure 4.2: Comparison of the widths of the bound ($\Gamma = 0$) and quasi-bound levels. The results from bound-continuum time-dependent simulations are given by the open circles, while results obtained from the program LEVEL are given by crosses [141].

The distribution of bound and quasi-bound levels for high vibrational numbers v exhibits several similarities to the experimental data (Fig. 4.3). For v =13, 14 and 15 the values of J which constitutes a boundary between bound and quasi-bound levels differs only by one. For v =16, because of strong perturbation with other states the comparison is impossible, while for $v = 17$, there is a perfect agreement. At last for v=18, no experimental data is available. The long broken vertical line placed at $J = 8$ crosses bound level with $v = 16$ and quasi-bound level with $v = 17$, and clearly illustrating that the last line with $v = 18$ in Fig. 4.4 comes from the virtual state lying above the potential barrier.

The part of the discrete spectrum calculated from Eq. 4.4 for the absorption transitions from the ground state level v" $= 3$, J" $= 8$ to the levels $v' = 16 - 18$, $J' = 8$ of $2^1\Pi$ is compared with the experimental spectrum in Fig. 4.4. The remaining peaks in the spectrum are as yet not assigned by experimentalists. I shall note here that I use the Condon approximation, i.e. a constant transition-dipole moment, as a result the relative intensities of the peaks may change when this approximation is abandoned. The calculated spectrum is shifted by 744.835 cm^{-1} with respect to the data in Tab. 4.1 (on page 58). This shift is calculated in the following manner: the experimental energy 726.485 cm^{-1} [145] of the level v" $= 3$, J" $= 0$ in the ground state was incremented by 18.350 cm^{-1} which is the energy difference between the ground state levels v" $= 3$, J" $= 8$ and v" $= 3$, J" $= 0$. The agreement between obtained spectra and the experimental spectra is very satisfactory showing the usability of those methods in assigning particular transitions to quasi-bound levels. The process of quantum tunneling and forming the spectral lines can be viewed in the molecular movies [140].

An overview of the calculated term energies are shown in Fig. 4.5. The present results of term values from $v = 8$ up to $v = 18$ show a characteristic pattern. The numerical values of level energies and widths of the $2^1\Pi$ state are shown in Tab. 4.1. Here only levels with

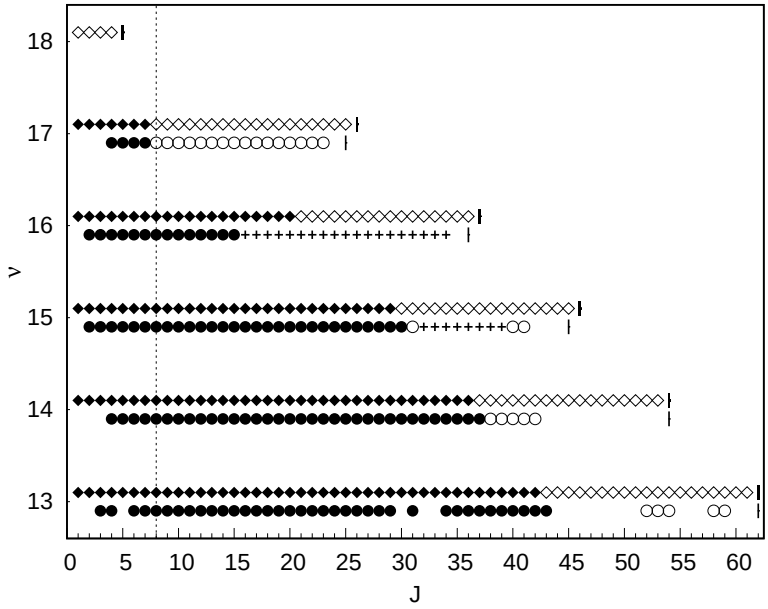


Figure 4.3: Distribution of the present theoretical results (diamonds in the upper rows) and experimental data [113] (circles in the lower rows) for the high rovibrational levels in the $2^1\Pi$ state of KLi. Solid figures represent bound levels, open figures - quasi-bound levels, crosses - perturbed levels. Short vertical bars denote the last rotational level for each v.
The long broken vertical line placed at $J = 8$ cuts v levels involved in transitions showed in Fig. 4.4.

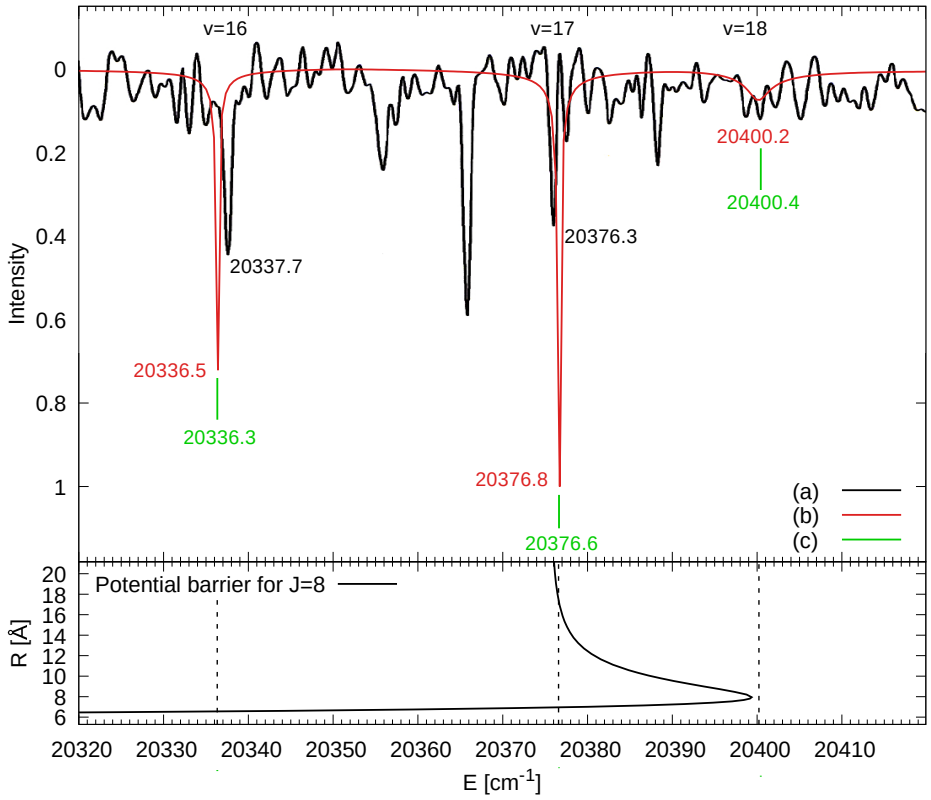


Figure 4.4: The black color shows the experimental spectrum of KLi observed with a linearly polarized pump laser beam [114, 146]. The assigned progression corresponds to transitions from the ground state levels v" = 3, J" = 8 to the levels $v' = 16$, 17, and 18, for $J' = 8$ of $2^1\Pi$. The red color is the spectrum calculated from Eq. 4.4 using potential curve of $2^1\Pi$ from [29]. The green color is the spectrum calculated from the program LEVEL.

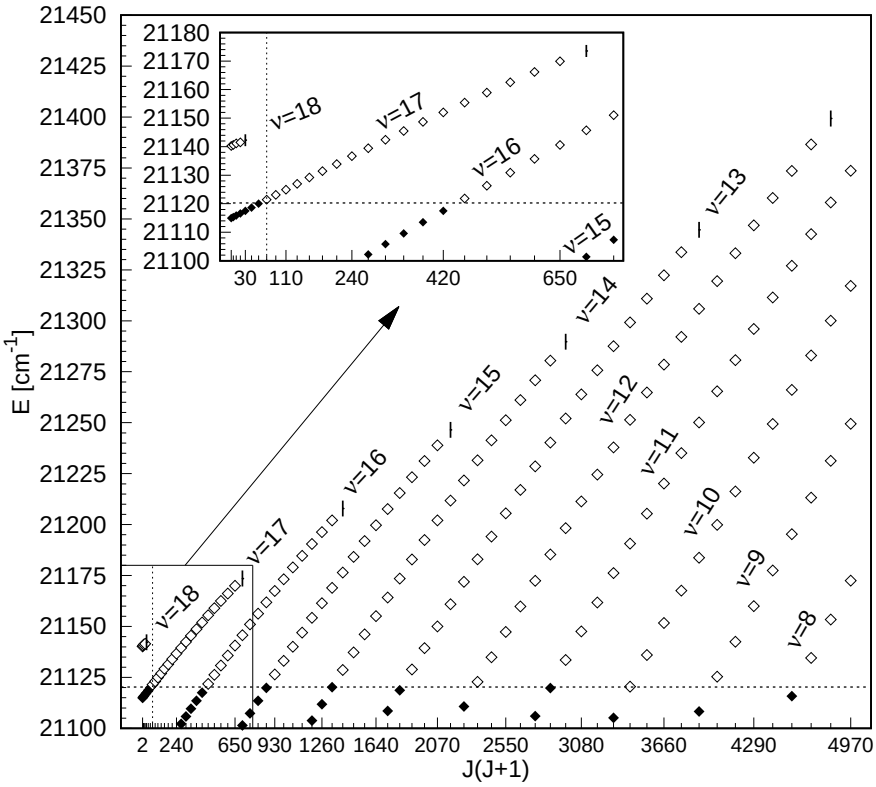


Figure 4.5: Energies of term values for v = 8-18 obtained from time-dependent calculations. Full diamonds indicate bound levels, empty diamonds - quasi-bound levels, short vertical bars - the last rotational level for a chosen v.

widths broader then 10^{-3} cm^{-1} are listed. As expected the biggest differences between theoretical results occur for the quasi-bound states lying just below the barrier's maximum, where the semi-classical approach becomes insufficient. The differences between theoretical results and available experimental data do not exceed the experimental resolution. It means that these set of results may be useful in further assigning of experimental spectrum. The energies, widths and lifetimes of all calculated quasi-bound levels are in [140].

4.1.3 Conclusions for KLi

In order to describe the rovibrational predissociation process of the KLi molecule I used the low lying adiabatic potential energy curve of the $2^1\Pi$ state from [29]. A small barrier to dissociation (~ 20 cm^{-1} for $J = 0$) is present in this state. Using the potential curve of $2^1\Pi$ state I calculated the rovibrational levels. The differences between the successive levels are compared with those derived from experimental data. The agreement is very good.

Using the complementary time-dependent approach I solved the time-dependent nuclear Schrödinger equation. The solutions show the evolving wavepacket originally placed on the effective potential curve. The spectrum is calculated as a Fourier transform of the autocorrelation function. The differences between successive peaks in the spectrum are compared with the experimental spectrum. In my calculations of the time-dependent population of the rovibrational (v, J) levels, I focus on the initial wavepackets chosen as eigenfunctions of quasi-bound states calculated with classical turning points. This approach allows for the exact description of the rovibrational predissociation mechanism of the KLi molecule. I show

explicitly that the population of a quasi-bound state decays exponentially in time. This approach can be easily used for other diatomic and even polyatomic molecules.

Next, I also describe the detailed procedure of calculating widths of quasi-bound vibrational levels with the high accuracy. The present method for solving TDSE (Section 2.1) is often used as a benchmark for testing other numerical methods [64]. It is especially important in the view of ultracold experiments as quasi-bond could possibly be explored for new ways of cooling molecules. Extensive Table 4.1 (on page 58) with calculated level energies, widths and lifetimes is presented. Those results are of considerable relevance to the design of experiments and the development of approximate computational methods. For available experimental data the comparison with my results gives very good agreement. Certainly, I hope that the results may be helpful in assigning transitions to quasi-bound levels.

Table 4.1: Energies and widths of chosen quasi-bound levels from the $2^1\Pi$ state. Present results obtained by time-dependent calculations (Present,TDSE) and by means of the program LEVEL (Present) are compared with those measured in the experiment [113]. Only levels with the f rotational symmetry are listed. The double column ΔE gives differences between Experiment and respectively Present,TDSE and Present. $\Delta\Gamma$ gives the relative difference between level widths calculated as (Present - Present,TDSE)/Present,TDSE.

		Energy [cm^{-1}]				Width Γ [cm^{-1}]		
J	v	Present,TDSE	Present	Experiment [113]	ΔE	Present,TDSE	Present	$\Delta\Gamma$
1	18	21140.338	21140.449			1.2580	1.3698	0.089
2	18	21140.605	21140.720			1.3096	1.4321	0.094
3	18	21141.006	21141.122			1.3787	1.5282	0.108
4	18	21141.507	21141.643			1.4948	1.6508	0.104
5	18	21142.175	21142.238			1.6331	1.7715	0.085
17	17	21142.408	21142.423	21142.417	0.009, -0.006	2.4877×10^{-3}	2.4938×10^{-3}	0.002
18	17	21145.514	21145.509	21145.434	-0.080, -0.075	7.3878×10^{-3}	7.4499×10^{-3}	0.008
19	17	21148.720	21148.719	21148.553	-0.167, -0.166	2.0125×10^{-2}	2.0341×10^{-2}	0.011
20	17	21152.026	21152.037	21151.714	-0.312, -0.323	4.9872×10^{-2}	5.0865×10^{-2}	0.020
21	17	21155.433	21155.451	21155.066	-0.367, -0.385	1.1437×10^{-1}	1.1653×10^{-1}	0.019
22	17	21158.939	21158.948	21158.400	-0.539, -0.548	2.4177×10^{-1}	2.4461×10^{-1}	0.012
23	17	21162.513	21162.521	21161.830	-0.683, -0.691	4.7034×10^{-1}	4.7253×10^{-1}	0.005
24	17	21166.153	21166.174			8.3086×10^{-1}	8.4896×10^{-1}	0.022
25	17	21169.860	21169.917			1.3596	1.4341	0.055
26	17	21173.594	21173.538			2.0983	2.1353	0.018
30	16	21167.455	21167.465	21167.380	-0.075, -0.085	1.5023×10^{-3}	1.5443×10^{-3}	0.028
31	16	21173.133	21173.152	21172.888	-0.245, -0.264	6.7315×10^{-3}	6.8733×10^{-3}	0.021
32	16	21178.877	21178.906	21178.406	-0.471, -0.500	2.5554×10^{-2}	2.5844×10^{-2}	0.011
33	16	21184.688	21184.702	21183.965	-0.723, -0.737	8.1813×10^{-2}	8.2820×10^{-2}	0.012
34	16	21190.499	21190.509	21189.630	-0.869, -0.879	2.2249×10^{-1}	2.2701×10^{-1}	0.020
35	16	21196.309	21196.311			5.2662×10^{-1}	5.3651×10^{-1}	0.019
36	16	21202.087	21202.120			1.0863	1.1147	0.026
37	16	21207.864	21207.938			1.9701	2.0734	0.052
40	15	21199.716	21199.713	21199.333	-0.383, -0.380	1.7736×10^{-3}	1.7881×10^{-3}	0.008
41	15	21207.564	21207.593	21206.890	-0.674, -0.703	8.7548×10^{-3}	8.9113×10^{-3}	0.018
42	15	21215.479	21215.488			3.6380×10^{-2}	3.6912×10^{-2}	0.015
43	15	21223.327	21223.358			1.2539×10^{-1}	1.2745×10^{-1}	0.016
44	15	21231.141	21231.163			3.6025×10^{-1}	3.6692×10^{-1}	0.019
45	15	21238.856	21238.891			8.7055×10^{-1}	8.8924×10^{-1}	0.021
46	15	21246.504	21246.573			1.7858	1.8662	0.045
49	14	21241.327	21241.345			4.2147×10^{-3}	4.2525×10^{-3}	0.009
50	14	21251.246	21251.231			2.0288×10^{-2}	2.0543×10^{-2}	0.013
51	14	21261.064	21261.066			8.1186×10^{-2}	8.2309×10^{-2}	0.014
52	14	21270.782	21270.793			2.6828×10^{-1}	2.7182×10^{-1}	0.013
53	14	21280.367	21280.376			7.3356×10^{-1}	7.4032×10^{-1}	0.009
54	14	21289.785	21289.840			1.6714	1.6977	0.016
56	13	21275.625	21275.668			1.2668×10^{-3}	1.2967×10^{-3}	0.024
57	13	21287.447	21287.484			7.1898×10^{-3}	7.3604×10^{-3}	0.024
58	13	21299.236	21299.248			3.3983×10^{-2}	3.4759×10^{-2}	0.023
59	13	21310.891	21310.900	21309.310	-1.581, -1.590	1.3478×10^{-1}	1.3582×10^{-1}	0.008
60	13	21322.379	21322.373			4.3449×10^{-1}	4.3478×10^{-1}	0.001
61	13	21333.634	21333.644			1.1463	1.1435	-0.002
62	13	21344.721	21344.746			2.5271	2.5240	-0.001

4.2 Power-law decay in the model NaRb dimer

In this section I use the quantum dynamics code from Sections 2.2 and 3.3 that allows to study the dynamics of multiple coupled states under the influence of an arbitrary time-dependent external field[31]. The goal of this section is to examine the capabilities of the new software by simulating an excitation in a model NaRb dimer. The model used here is relatively simple: without multiphoton excitations, without ionization channel, with classical electromagnetic field and with rotational quantum number fixed at $J = 0$. Three electronic levels are used here: $1^3\Sigma^+$, $2^1\Sigma^+$ and $1^3\Pi$ together with the spin-orbit coupling (SOC) between $2^1\Sigma^+$ and $1^3\Pi$. In this model a strong femtosecond impulse excitation causes association and the forbidden transition in the $^{23}Na^{87}Rb$ dimer. In this numerical study, the dimer is excited from the ground triplet state $1^3\Sigma^+$ to the $1^3\Pi$ state using the femtosecond impulse and next, the spin-orbit coupling between the $1^3\Pi$ and $2^1\Sigma^+$ states results in the singlet-triplet transition. The impulse parameters are optimised to obtain maximum yield in electronic levels correlating with the first excited atomic asymptote. As a result I obtain the detailed population statistics and power-law decay of these states. Finally, the analysis of the population oscillations allows for the determination of the optimal time delay for dumping the molecule to its absolute ground state.

The aim in this section is to show the time-dependent descriptions of the photoassociation and photodissociation processes in the polar alkali diatomic molecules and the quantum properties of these reactions in their characteristic time regimes. I propose the femtosecond impulse parameters' optimization procedure providing the maximization of the population in the coupled complex of excited electronic states. Dynamics investigations of this system of molecular states allows, in the case of an association, to establish the optimal time delays for the femtosecond train of impulses enabling the formation of molecules in the deeply bound ground state, as well as present the general modified power-law decay allowing for the proper description of the dissociation process of coupled states. The dissociation through the barrier of the potential (tunneling) is well described by the exponential decay (see previous section and [29, 39, 147]), while the dissociation of the system of coupled excited electronic states should be rather described by other forms of decays. Several different models for these kinds of decays were investigated, such as product decay [148], dephasing [149], and the power-law decay [150, 151, 152]. The fit on the data obtained here will show that it is a power-law decay.

All considerations here are based on an exemplary numerical model of a polar alkaline dimer, which is the NaRb molecule. The sodium rubidium molecular system has been studied both experimentally [153, 154, 155, 156, 157] and theoretically [116, 158, 159, 160]. It is worth underlining that in a recent experiment [161, 162] Guo et al. reported successful production of an ultracold sample of the absolute ground state of $^{23}Na^{87}Rb$ molecules.

Here I am simulating the impulse-driven dynamics of photoassociation of the NaRb dimer that takes the spin-orbit coupling into consideration. The dynamics on multiple coupled electronic levels are determined by solutions to the coupled time-dependent Schrödinger equations using the *semi-global* method [41] (Sections 2.2 and 3.3). The effective Hamiltonian for describing nuclear kinetics in coupled multiple adiabatic or diabatic electronic states plays a key role. The newly-developed quantum dynamics code allows to study the dynamics of multiple coupled states under the influence of an arbitrary time-dependent external field,

[31] This section is in preparations to publish and was presented on a poster in [30].

Table 4.2: The schematic view of the Hamiltonian matrix of the system used in the calculations. The transition dipole moment functions (TDMF) between electronic levels μ multiplied by the impulse electric field $\varepsilon(t)$ are marked with '+'. The lack of coupling between levels is marked with '∘'. The spin-orbit coupling is marked with SOC. The potential energy curves (PEC) are on the matrix diagonal.

	$1^3\Sigma^+$	$2^1\Sigma^+$	$1^3\Pi$
$1^3\Sigma^+$	PEC	∘	+
$2^1\Sigma^+$	∘	PEC	SOC
$1^3\Pi$	+	SOC	PEC

to investigate the femtosecond impulse-driven process of association and the forbidden transition in the $^{23}Na^{87}Rb$ dimer. The Born-Oppenheimer potential energy curves (PEC) used here are from [160].

In this analysis, I use the lowest electronic states of the NaRb molecule, appropriate transition dipole moment functions (TDMF), and spin-orbit coupling (SOC) matrix elements. Additionally the femtosecond impulse parameters are optimized to obtain maximum yield in electronic levels correlating with the first excited atomic asymptote. I present a map of the optimization space of the pulse parameters and the detailed population statistics. Inspired by [150] I propose a modified power-law to describe the population decay. Additionally, the analysis of the oscillations of population allows to determine the optimal time delay for dumping molecule to its absolute ground state.

4.2.1 Computation method

The time propagation of a system of coupled time dependent nuclear Schrödinger equations (TDSE) for multiple electronic levels and time-dependent Hamiltonian follows the *semi-global* method [41] (Sections 2.2 and 3.3). The couplings between the electronic levels are time-dependent. The TDSE has the following general form:

$$i\hbar\frac{\partial}{\partial t}\begin{pmatrix}\psi_1\\ \psi_2\\ \psi_3\end{pmatrix} = \begin{bmatrix}\hat{\mathbf{H}}_1(t) & \hat{\mathbf{V}}_{1,2}(t) & \hat{\mathbf{V}}_{1,3}(t)\\ \hat{\mathbf{V}}_{2,1}(t) & \hat{\mathbf{H}}_2(t) & \hat{\mathbf{V}}_{2,3}(t)\\ \hat{\mathbf{V}}_{3,1}(t) & \hat{\mathbf{V}}_{3,2}(t) & \hat{\mathbf{H}}_3(t)\end{bmatrix}\begin{pmatrix}\psi_1\\ \psi_2\\ \psi_3\end{pmatrix}, \tag{4.5}$$

where the $\hat{\mathbf{H}}_m(t)$ corresponds to the Hamiltonian for the respective electronic level in the Born-Oppenheimer approximation and $\hat{\mathbf{V}}_{m,n}(t)$ are either: (1) the respective transition dipole moment functions μ multiplied[32] by the impulse electric field $\varepsilon(t)$ or (2) the spin orbit potential ξ; depending on the nature of the coupling between the two electronic levels. The Hamiltonian for three electronic levels in this simulation is summarized in Tab. 4.2.

The gist of the *semi-global* method [41] (Sections 2.2 and 3.3) is that the evolution operator is expanded, using K terms (see Tab. 2.2 on page 26), in the Krylov space into two parts: the time-dependent part of the Hamiltonian and the Hamiltonian in the middle of the timestep $t+\frac{\Delta t}{2}$, for which the time-dependent part serves as a correction calculated in the

[32] Following Tannor [39] (page 396) I neglect the vector character of μ.

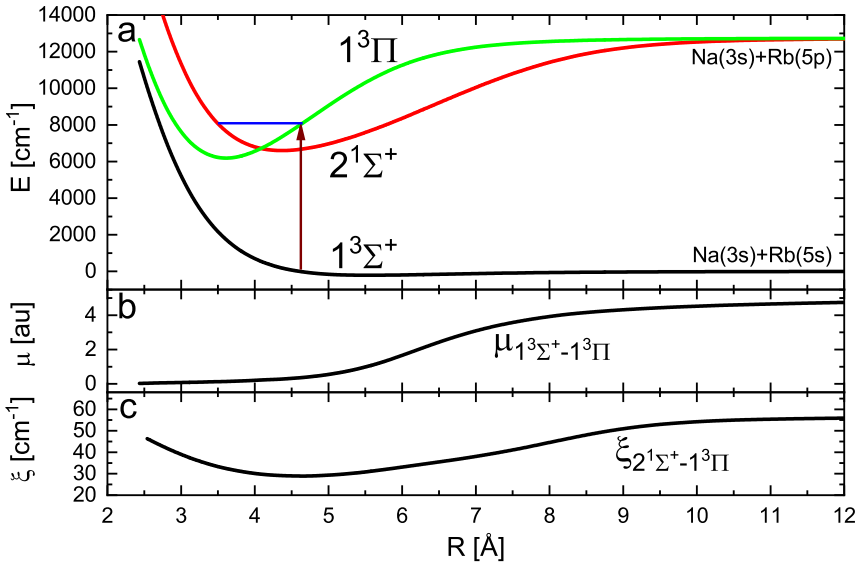


Figure 4.6: (a) the electronic states of the NaRb molecule with the schematic arrow representing transition imposed by the impulse; (b) the transition dipole moment function μ and (c) the spin-orbit coupling ξ. See Tab. 4.2 for the schematic view of the Hamiltonian in which they are used.

M interior sub-timesteps. The calculation of solutions for M interior sub-timesteps is iterated several times (usually two to five) within a single global timestep Δt until the solution converges with the requested error tolerance ε. Then the calculation moves on to the next timestep.

The *semi-global* method is implemented in C++ (Section 3.3), thus allowing calculations $10\times$ faster than the original Matlab code (see Section 3.5 and [41]) and extended with the ability to handle an arbitrary number of electronic levels (a feature not present in [41], see Section 2.2.7 and Listing 3.4) such as in Eq. 4.5. It is then used with the following parameters (Table 2.2 on page 26): the timestep $\Delta t = 1$ a.u., the number of interior Chebyshev time points $M = 5$ and the number of expansion terms used for the computation of the function of a matrix $K = 7$. The error tolerance is set to $\varepsilon = 10^{-10}$. The discretization grid has 6144 points and the absorbing boundary condition (see Section 2.2.9) is placed at the distance $R_{max} = 70\ a_0$.

4.2.2 Results and discussion

The previously calculated PECs from [160], TDMF $\mu_{1^3\Sigma^+-1^3\Pi}$ and SOC $\xi_{2^1\Sigma^+-1^3\Pi}$ along with the schematic photoassociation transitions are shown in Fig. 4.6.

As the initial condition, the $v = 20$ eigenfunction is placed on the $1^3\Sigma^+$ state, because it is one of the highest populated levels obtained in the experiment [161, 162]. Then the following impulse is used:

$$\varepsilon(t) = I\ \mathrm{sech}^2\left((t - t_p)/\tau_p\right)\cos(\omega t), \tag{4.6}$$

where the frequency ω corresponds to a commercially available laser with the wavelength

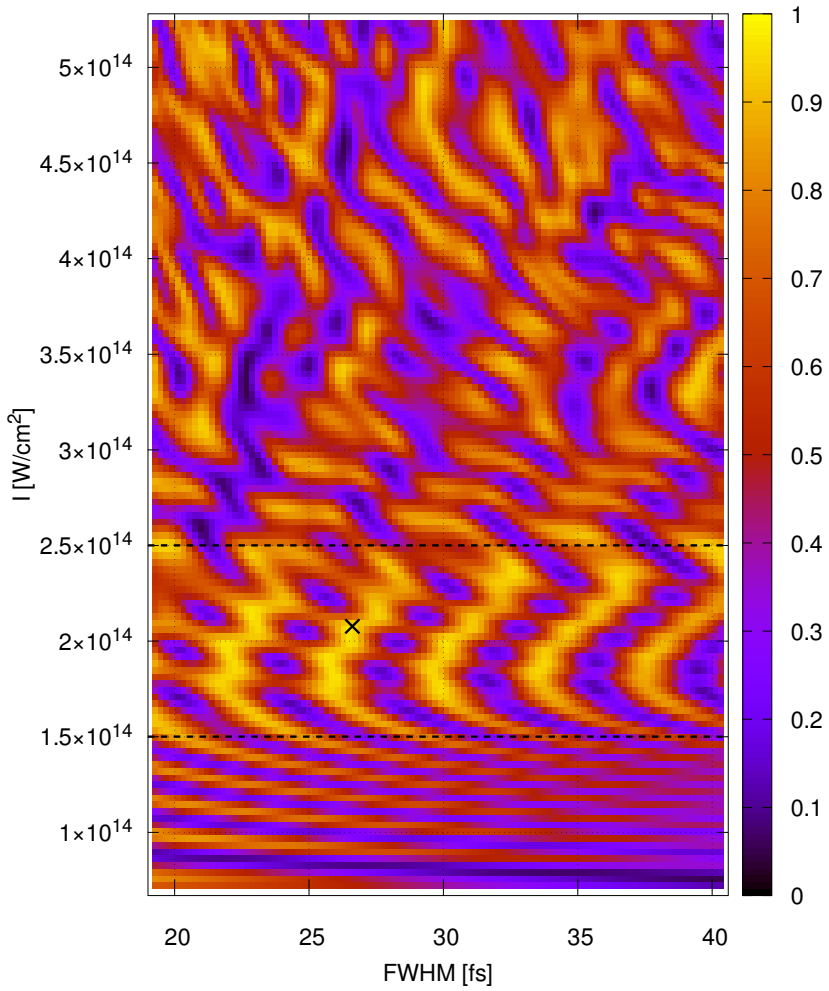


Figure 4.7: Sum of the population on $2^1\Sigma^+$ and $1^3\Pi$ depending on the full width at half maximum (FWHM) of the impulse and its intensity I. The peak population $P_0 = 97.2\%$ at FWHM=26.6 fs, $I = 2.0768 \times 10^{14}$ [W/cm^2] is marked with a cross. The horizontal bands separated by dashed lines are discussed in the text.

$\lambda = 1560$ nm, the impulse center $t_p = 111.26$ fs (4600 a.u., also marked in Fig. 4.8 with an arrow), while the intensity I and the full width at half maximum (FWHM) are optimized to maximize population on the excited state (the FWHM of the impulse equals to $1.76\tau_p$). The optimization is performed by scanning a range of these two parameters.

The scan of the sum of the population on two excited states $2^1\Sigma^+$ and $1^3\Pi$ is shown in Fig. 4.7. It is interesting to note the existence of three characteristic horizontal bands. The first band occurs for the intensity $I < 1.5 \times 10^{14}$ [W/cm^2]. Here the horizontal lines of the higher population indicate that the FWHM energy spread of the impulse does not affect the population and it mostly depends on the impulse intensity. The second band is in the middle range of I between 1.5×10^{14} and 2.5×10^{14} [W/cm^2]. Here a semi-diagonal pattern emerges where the FWHM energy spread dependence plays a major role. Finally in the third range for $I > 2.5 \times 10^{14}$ [W/cm^2] a nonlinear behavior takes precedence and a chaotic pattern emerges. These findings indicate that there is possible an interplay between the intensity and the impulse duration with the eigenenergies and the vibration frequencies of the NaRb molecule occurring on the target electronic state $1^3\Pi$.

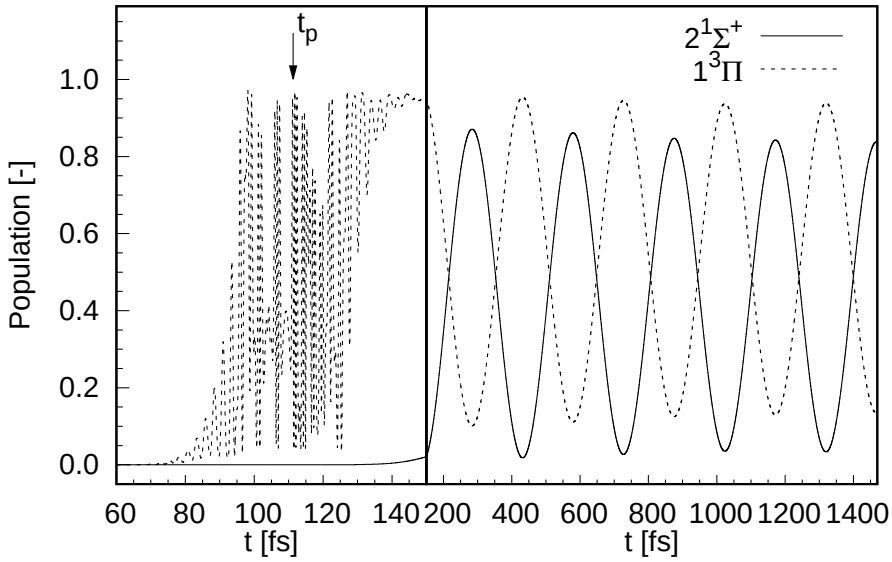


Figure 4.8: The evolution of the population on the two excited states ($2^1\Sigma^+$, $1^3\Pi$). *The impulse FWHM is 26.6 fs and the intensity is* 2.0768×10^{14} *[W/cm*^{2}*]. The impulse parameters correspond to the peak population found in the scan in Fig. 4.7. The period of oscillations between the two levels is 296.35 fs. The first peak of* $2^1\Sigma^+$ *is 172.49 fs after* t_p *(see Eq. 4.6). In the figure the first 60 fs, when the impulse is weak and population is near zero, are not shown for brevity.*

Next, using the optimal parameters I and τ_p (marked with a cross in Fig. 4.7) corresponding to the maximum population, the time evolution of the population on the excited electronic levels is performed. Fig. 4.8 shows results for obtaining the maximum population found, which is $P_0 = 97.2\%$.

The effect of the spin-orbit coupling is visible in Fig. 4.8 as the population exchange between levels $2^1\Sigma^+$ and $1^3\Pi$. The period of the oscillation between the two levels is 296.35 fs. The first maximum on $2^1\Sigma^+$ occurs 172.49 fs after the center of the impulse t_p. This is the optimal time delay for the next impulse in this numerical study to dump the molecule to the absolute ground state using a pulse train with a period of 296.35 fs. It means that the peaks of the pulse train should correspond to the peaks in the $2^1\Sigma^+$ state.

The population between the electronic levels is constantly shifted due to the spin-orbit coupling. Figure 4.9 shows that there occurs a population decay. The higher frequencies in the wavepacket which are above the first excited atomic asymptote are undergoing a dissociation. This process is slowed down due to the interaction of the wavepacket with the coupled potential on two electronic levels. This causes a population decay over a larger timescale and only a fraction of the original population remains at the end, as shown in Fig. 4.9. As shown below, this is not an exponential decay as in the predissociation process (Section 4.1 and [29]). It is instead a distribution of the power-law, as it happens by the coincidence of population exchange between two levels [150, 163, 164, 165].

Following the idea in [150] that the population decay in the case of two coupled levels should follow a power-law decay, in this model I analyze the population decay on the two excited levels $2^1\Sigma^+$ and $1^3\Pi$. The calculated population from Fig. 4.9 is fitted respectively to the modified power-law decay formula:

$$P(t) = P_0 \left((1 - P_{f,pow}) \sqrt{\frac{\tau_{pow}}{t + \tau_{pow} - t_0}} + P_{f,pow} \right) \tag{4.7}$$

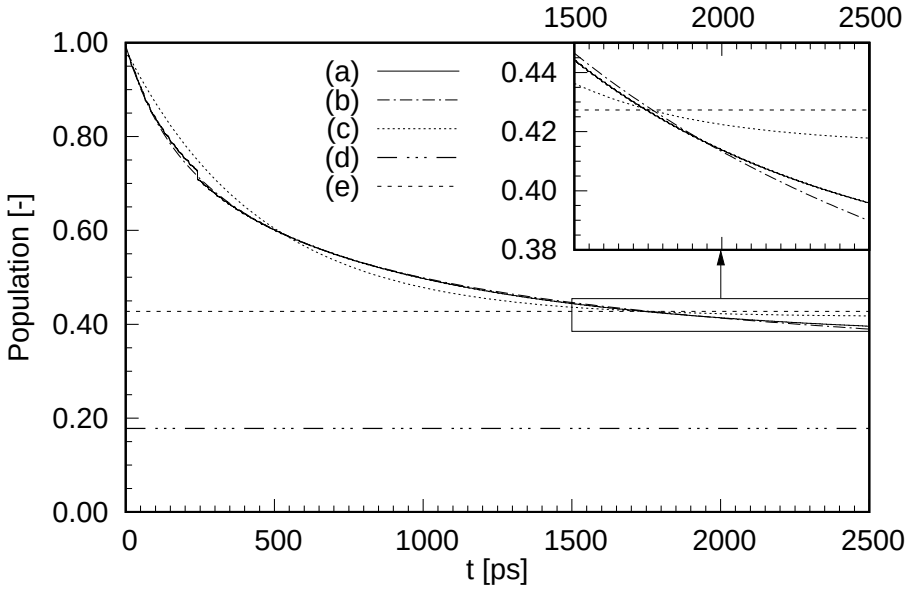


Figure 4.9: Sum of population on $2^1\Sigma^+$ and $1^3\Pi$ decaying during the first 2500 ps. The decay starts at $t_0 = 10$ ps. (a) numerical result; (b) best fit using the power-law in Eq. 4.7; (c) best fit using the exponential decay law in Eq. 4.8; (d) The best fit value for the final population $P_{f,pow}$ in Eq. 4.7; (e) The best fit value of the final population $P_{f,exp}$ in Eq. 4.8.

and to the exponential decay formula:

$$P(t) = P_0 \left((1 - P_{f,exp}) \exp\left(-\frac{t - t_0}{\tau_{exp}} \right) + P_{f,exp} \right), \tag{4.8}$$

where $P_0 = 97.2\%$ is the initial population on both levels, $t_0 = 10$ ps is the start of the decay, τ_{pow}, τ_{exp} are the fitting decay parameters, and $P_{f,pow}$ and $P_{f,exp}$ are the fitting parameters describing the final non-zero population. I modified the $\sqrt{\bullet}$ part of Eq. 4.7 from [150] with an extra τ_{pow} in the denominator in order to shift the function to the left so that the initial population is not infinite when $t = t_0$ (as is the case in [150]), but instead equals P_0. Also in both equations for decay I added a modification of type: $(1 - P_f) \bullet + P_f$ to allow extra fitting parameters $P_{f,pow}$ and $P_{f,exp}$ for the final non-zero population. It shall be noted that compared to [150] the situation here is also different: both potentials $2^1\Sigma^+$ and $1^3\Pi$ are bonding potentials and neither of them is dissociative (see Fig. 4.6). Therefore, it is worth seeing how the modified power-law decay fits the obtained numerical results.

The fit results in Fig. 4.9 are as follows: $\tau_{pow} = 196.03 \pm 0.04$ ps, $P_{f,pow} = 0.1801 \pm 0.0001$ and $\tau_{exp} = 459.7 \pm 0.3$ ps, $P_{f,exp} = 0.4246 \pm 0.0001$. The asymptotic standard error percentage of the fit [166] for the power-law is 0.02% and for the exponential law is 0.06%. The best fit for exponential law produces a nonphysical result, because the fitted final population $P_{f,exp}$ (Fig. 4.9e) has value greater than the steadily decreasing population (a) or (c). The fit for the power-law (Fig. 4.9b) is significantly better, since it almost overlaps the numerical result (Fig. 4.9a), with error 0.02%. The final population (Fig. 4.9d) is $P_{f,pow} = 18\%$. Overall the modified power-law is a better description of the decay occurring in this situation.

4.2.3 Conclusions for NaRb

In this section, I show the time evolution of three coupled potential energy curves of an exemplary model diatomic NaRb system. The necessary potential energy curves and dipole transition moments were taken from earlier ab initio calculations [160] and supplemented with a calculated spin-orbit coupling between the upper singlet and triplet states [30]. In this approach, the spin-orbit coupling is a function depending on the distance between the nuclei. The considered system is driven by a femtosecond impulse. The two parameters describing the femtosecond impulse, i.e. intensity and half-width, are independently optimized. The aim of optimization is to obtain the largest possible population of molecular states correlated with the first excited atomic asymptote. I identify the period of population oscillation between the two excited levels $2^1\Sigma^+$ and $1^3\Pi$. This allows to design the optimal sequence of impulses to dump the molecule to the ground state. The optimal time delay after a pump pulse to dump the molecule to the ground state is 172.49 fs using a pulse train of 296.35 fs. I am also examining the combined population decay from these two levels and fit this to a modified power law. Then I show that this fit is better than that of an ordinary exponential decay. I would like to emphasize that these numerical results can be important for a thorough insight into quantum processes where time plays a key role. This approach is possible to apply to even more complex systems where only one soft bond (e.g. between two fragments in a molecule as in [7]) is most important to describe a time-dependent process. Also I confirm the finding in [7] that impulse control can be applied to the spin-orbit coupled states.

The next step for the calculations presented in this section is to replace the femtosecond excitation with a femtosecond laser impulse with intensity and FWHM commonly used in experiments as well as taking into account the permanent dipole moment and light polarisation. Furthermore an optimisation of the second femtosecond laser impulse to obtain an ultracold NaRb dimer can be performed.

All dynamic results are obtained from a new computer code which uses a *semi-global* method [41] (Sections 2.2 and 3.3) to expand the time evolution operator. The method is improved to work with multiple electronic levels and time dependent couplings. It can be a valuable tool for scientists studying quantum dynamics and planning future experiments.

Chapter 5

Conclusions and future work

This dissertation is designed to solve the problem of performing high-precision quantum dynamics calculations for a system of coupled Schrödinger equations, for any amount of coupled electronic states. And also to apply it to dissociation in the KLi dimer and power-law in the NaRb dimer, as well as compare it with experiments and reference solutions found in literature.

In Chapter 2 I have shown two time integration algorithms designed to propagate in time the Schrödinger equation. First algorithm is suitable for time-independent Hamiltonian and is known for being remarkably precise, as it is used as a reference precision benchmark for other time integration algorithms [64]. The second semi-global algorithm is the best for situations where a time-dependent Hamiltonian is used, such as ultrafast spectroscopy, high harmonic generation, coherent control problems or attosecond laser impulses.

In Chapter 3 I have discussed the details of the numerical implementation of both of these algorithms. Also I have performed a careful validation of these implementations comparing my results with numerous reference results: single avoided crossing, dual avoided crossing, Gaussian packet in a harmonic oscillator and a forced harmonic oscillator with an inhomogeneous source term. Also I have completely reproduced the calculation of a model atom in an intense laser field [41]. This allowed me to confidently progress further knowing that my implementations are correct. Based on my earlier work [28] I have implemented these algorithms in high precision. It is possible to perform calculations using 15, 18 or 33 decimal places. And I have plans to expand the precision to arbitrary number of decimal places after the work presented in [61, 62] is complete.

Next, in Chapter 4 I have shown that the implemented algorithms, which are the main subject of this thesis, for both time-dependent and time-independent Hamiltonian, work well and agree with the experiment.

In Section 4.1 I have calculated the rovibrational predissociation in the KLi molecule with the time-independent Hamiltonian algorithm from Section 2.1. Using the potential curve of $2^1\Pi$ state, which has a small barier to dissociation (~ 20 cm^{-1} for $J = 0$), I calculated the rovibrational levels and obtained a very good agreement with experimental results. Also I have shown that the quasi-bound state decays exponentially in time and I have described a robust method to calculating the widths of quasi-bound vibrational levels with high accuracy. I hope that those results are of considerable relevance to the design of experiments.

In Section 4.2 I have calculated the power-law decay in the model NaRb dimer. I used the time-dependent Hamiltonian semi-global algorithm from Section 2.2. I performed the calculations with three coupled electronic states ($1^3\Sigma^+$, $2^1\Sigma^+$, $1^3\Pi$) and solved the corresponding

three coupled Schrödinger equations. Upon excitation with an optimised femtosecond impulse the wavepacket started to move between two upper electronic levels ($2^1\Sigma^+$, $1^3\Pi$). This caused the combined population to decay from these two levels in the presented model. I have performed a fit of this decay to a modified power-law and I have shown that this fit is better than a fit to the exponential decay. Also I have provided the impulse parameters which in this model may be used to perform photoassociation to obtain an ultracold NaRb model molecule in the ground state. I would like to emphasize that these numerical results can be important for a thorough insight into quantum processes where time plays a key role.

Results from Section 4.1 are already published in [29], results from Section 4.2 are in preparations to publish and were presented on a poster in [30].

In the future I plan to perform more calculations of the NaRb dimer to use the proposed second laser impulse to bring it down to the ground level electronic state. Moreover I plan to do similar calculations in the Hund-C case, which takes into account the strong spin-orbit coupling, because in Section 4.2 I have been dealing with Hund-A case only. The Hund-C case calculations will involve seven electronic states. Next, I plan to perform similar Hund-A vs. Hund-C case comparison in the KRb dimer and find out what femtosecond laser impulses are necessary to obtain an ultracold KRb molecule in the electronic ground state level.

All this work, however, is a stage in my scientific program. I already have a fairly good knowledge of classical dynamics (c.f. publications from my earlier career on page 88). And in this dissertation I have obtained knowledge of quantum dynamics [29, 30, 31, 32]. Next I plan to better understand the quantum field theory, how the Feynman diagrams are calculated with the electroweak force and quantum chromodynamics, as well as the lattice approaches. Following this I plan to improve my understanding of the general relativity, how the curvature of spacetime is calculated from the stress energy tensor and how to solve the Einstein's field equations. Finally I am planning to understand and implement a code for the loop quantum gravity, as I currently believe it is the best shot at the unification theory.

Appendix A

Chebyshev polynomials

In the Kosloff approach [63] the Chebyshev polynomials are used. The Chebyshev polynomials T_k of the first kind [72] (page 776, Eq.22.3.15) are defined as:

$$T_k(x) = \cos(k \arccos(x)), \quad \text{for} \quad x \in [-1,1], k \in \mathbb{N}_0 \,. \tag{A.1}$$

And they can be also constructed using following recurrence relation:

$$\begin{cases} T_0(x) = 1 \\ T_1(x) = x \\ T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x) \,. \end{cases} \tag{A.2}$$

The Chebyshev polynomials form an orthogonal basis in the range $x \in [-1,1]$ with the weight $\frac{1}{\sqrt{1-x^2}}$:

$$\int_{-1}^{1} \frac{T_m(x)T_k(x)}{\sqrt{1-x^2}} dx = \begin{cases} \pi\, \delta_{m,k} & \text{for} \quad m = k = 0 \\ \frac{\pi}{2}\, \delta_{m,k} & \text{for} \quad m \neq 0, k \neq 0 \,. \end{cases} \tag{A.3}$$

The above relation can also be written using Kronecker's deltas, after dividing both sides by π and multiplying both sides by $(2 - \delta_{m,0}\delta_{k,0})$:

$$\frac{2 - \delta_{m,0}\delta_{k,0}}{\pi} \int_{-1}^{1} \frac{T_m(x)T_k(x)}{\sqrt{1-x^2}} dx = \delta_{m,k} \,, \tag{A.4}$$

please note however that the $\delta_{m,0}$ is redundant with the integral $\int_{-1}^{1} \frac{T_m(x)T_k(x)}{\sqrt{1-x^2}} dx$ because this integral is equal to zero if $m \neq k$, hence $\delta_{m,0}$ will be omitted when this relation will be used.

Also the following identity [63, 72] between Chebyshev polynomials and Bessel functions of the first kind will be useful for some $R \in \mathbb{R}$:

$$\frac{2 - \delta_{k,0}}{\pi} \int_{-1}^{1} \frac{T_k(x) e^{ixR}}{\sqrt{1-x^2}} dx = \left(2 - \delta_{k,0}\right) i^k J_k(R) \,. \tag{A.5}$$

Figure A.1, on the next page, shows the first seven Chebyshev polynomials.

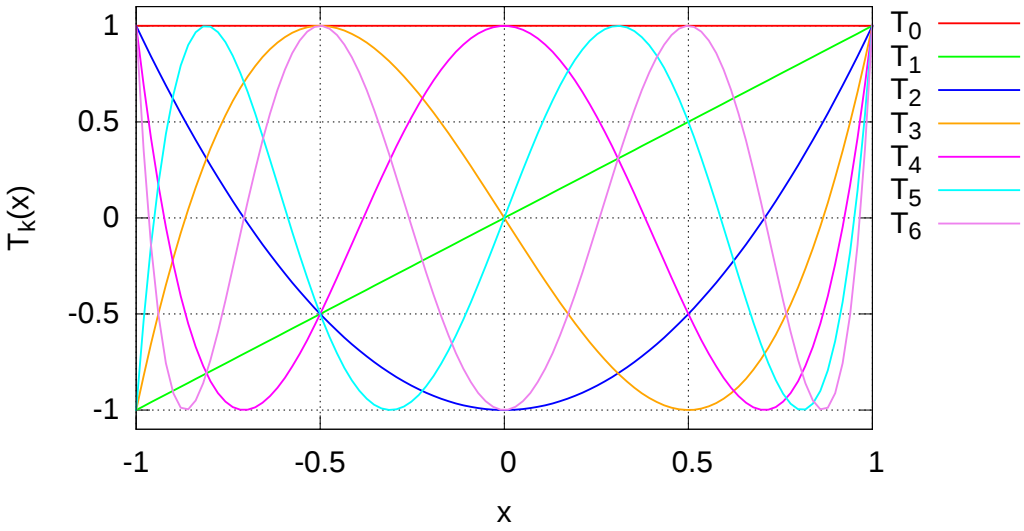


Figure A.1: Chebyshev polynomials from $T_0(x)$ to $T_6(x)$, $x \in [-1,1]$.

Appendix B

Numerical approximation of ∇^2

The fact that the second derivative $\frac{\partial^2\psi(\vec{r},t)}{\partial x^2}$ satisfies following relation to Fourier transform $\mathcal{F}$:

$$\begin{aligned}\frac{\partial^2\psi(\vec{r},t)}{\partial x^2} &= \frac{\partial^2}{\partial x^2}\underbrace{\left(\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}\psi(\vec{k},t)e^{i\vec{k}\cdot\vec{r}}\,d\vec{k}\right)}_{=\psi(\vec{r},t)=\mathcal{F}^{-1}(\mathcal{F}(\psi(\vec{r},t)))} \\ &= \int_{-\infty}^{\infty}-{k_x}^2\frac{1}{2\pi}\int_{-\infty}^{\infty}\psi(\vec{x},t)e^{-i\vec{k}\cdot\vec{r}}\,d\vec{r}\,e^{i\vec{k}\cdot\vec{r}}\,d\vec{k} = \mathcal{F}^{-1}\left(-{k_x}^2\mathcal{F}\left(\psi(\vec{r},t)\right)\right),\end{aligned} \tag{B.1}$$

can be used to approximate ∇^2 numerically as $\widetilde{\nabla}^2$ by using the Discrete Fourier Transform $\widetilde{\mathcal{F}}$. The derivatives in more spatial dimensions are independent hence:

$$\widetilde{\nabla}^2\widetilde{\psi}(\vec{r},t) = \widetilde{\mathcal{F}}^{-1}\left(-\vec{k}^2\widetilde{\mathcal{F}}\left(\widetilde{\psi}(\vec{r},t)\right)\right), \tag{B.2}$$

where in one dimensional case:

$\mathcal{F}$ — Fourier transform: $$\mathcal{F}(\psi(x,t)) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}\psi(x,t)e^{-ikx}\,dx = \psi(k,t)$$

$\mathcal{F}^{-1}$ — inverse Fourier transform: $$\mathcal{F}^{-1}(\psi(k,t)) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}\psi(k,t)e^{ikx}\,dk = \psi(x,t)$$

$\widetilde{\mathcal{F}}$ — discrete Fourier transform: $$\widetilde{\mathcal{F}}(\widetilde{\psi}(x_n,t)) = \frac{1}{\sqrt{N}}\sum_{m=0}^{N-1}\widetilde{\psi}(x_m,t)e^{\frac{-i2\pi nm}{N}} = \widetilde{\psi}(k_n,t)$$

$\widetilde{\mathcal{F}}^{-1}$ — inverse discrete Fourier transform: $$\widetilde{\mathcal{F}}^{-1}(\widetilde{\psi}(k_n,t)) = \frac{1}{\sqrt{N}}\sum_{m=0}^{N-1}\widetilde{\psi}(k_m,t)e^{\frac{i2\pi nm}{N}} = \widetilde{\psi}(x_n,t).$$

Discretized position is $x_n = x_{min} + n\Delta x$; discretized wavenumber is $k_n = -k_{max} + \frac{n}{N}2k_{max}$. Any derivative can be obtained via Eq. B.1 by putting the desired derivative order as ξ[33]:

$$\widetilde{\nabla}^{\xi}\widetilde{\psi}(\vec{r},t) = \widetilde{\mathcal{F}}^{-1}\left((i\vec{k})^{\xi}\widetilde{\mathcal{F}}\left(\widetilde{\psi}(\vec{r},t)\right)\right), \tag{B.3}$$

[33]This should work even for fractional derivatives, an interesting avenue. But it is different than the Fractional Fourier Transform (FRFT) which operates in any intermediately rotated domain between time and frequency or position and momentum.

Bibliography

[1] A. Palacios and F. Martín. "The quantum chemistry of attosecond molecular science". en. In: *WIREs Computational Molecular Science* 10.1 (2019). DOI: 10.1002/wcms.1430 (cit. on pp. 1, 27).

[2] A. Zewail. "Femtochemistry: Atomic–Scale Dynamics of the Chemical Bond". In: *J. Phys. Chem. A* 104 (2000), pp. 5660–5694 (cit. on p. 1).

[3] P. V. Kamat and G. C. Schatz. "Advancing the Frontiers of Physical Chemistry". en. In: *The Journal of Physical Chemistry Letters* 3.1 (2012), pp. 38–39. DOI: 10.1021/jz201567f (cit. on p. 1).

[4] K.-J. Yuan, C.-C. Shu, D. Dong, and A. D. Bandrauk. "Attosecond Dynamics of Molecular Electronic Ring Currents". en. In: *The Journal of Physical Chemistry Letters* 8.10 (2017), pp. 2229–2235. DOI: 10.1021/acs.jpclett.7b00877 (cit. on p. 1).

[5] L. E. Rosaleny et al. "Peptides as Versatile Platforms for Quantum Computing". en. In: *The Journal of Physical Chemistry Letters* 9.16 (2018), pp. 4522–4526. DOI: 10.1021/acs.jpclett.8b01813 (cit. on p. 1).

[6] S. Jahangiri, J. M. Arrazola, and A. Delgado. "Quantum Algorithm for Simulating Single-Molecule Electron Transport". en. In: *The Journal of Physical Chemistry Letters* 12.4 (2021), pp. 1256–1261. DOI: 10.1021/acs.jpclett.0c03724 (cit. on p. 1).

[7] Z. Wei, J. Li, S. T. See, and Z.-H. Loh. "Spin–Orbit State-Selective C–I Dissociation Dynamics of the CH_3I^+ $\tilde{X}$ Electronic State Induced by Intense Few-Cycle Laser Fields". en. In: *The Journal of Physical Chemistry Letters* 8.24 (2017), pp. 6067–6072. DOI: 10.1021/acs.jpclett.7b03022 (cit. on pp. 1, 65).

[8] I. S. Vinklárek et al. "Energy partitioning and spin–orbit effects in the photodissociation of higher chloroalkanes". en. In: *Physical Chemistry Chemical Physics* 23.26 (2021), pp. 14340–14351. DOI: 10.1039/d1cp01371h (cit. on p. 1).

[9] Y. Chang et al. "Three body photodissociation of the water molecule and its implications for prebiotic oxygen production". en. In: *Nature Communications* 12.1 (2021). DOI: 10.1038/s41467-021-22824-7 (cit. on p. 1).

[10] I. A. Ramphal, M. Shapero, and D. M. Neumark. "Photodissociation Dynamics of the Cyclohexyl Radical from the 3p Rydberg State at 248 nm". en. In: *The Journal of Physical Chemistry A* 125.18 (2021), pp. 3900–3911. DOI: 10.1021/acs.jpca.1c02393 (cit. on p. 1).

[11] Y. Zhao et al. “Rotational and nuclear-spin level dependent photodissociation dynamics of H_2S”. en. In: *Nature Communications* 12.1 (2021). DOI: 10.1038/s41467-021-24782-6 (cit. on p. 1).

[12] M. L. Murillo-Sánchez et al. “Femtosecond XUV–IR induced photodynamics in the methyl iodide cation”. In: *New Journal of Physics* 23.7 (2021), p. 073023. DOI: 10.1088/1367-2630/ac0c9b (cit. on p. 1).

[13] C. P. Koch and M. Shapiro. “Coherent Control of Ultracold Photoassociation”. en. In: *Chemical Reviews* 112.9 (2012), pp. 4928–4948. DOI: 10.1021/cr2003882 (cit. on p. 1).

[14] J. Ulmanis, J. Deiglmayr, M. Repp, R. Wester, and M. Weidemüller. “Ultracold Molecules Formed by Photoassociation: Heteronuclear Dimers, Inelastic Collisions, and Interactions with Ultrashort Laser Pulses”. en. In: *Chemical Reviews* 112.9 (2012), pp. 4890–4927. DOI: 10.1021/cr300215h (cit. on p. 1).

[15] J. L. Carini, S. Kallush, R. Kosloff, and P. L. Gould. “Efficient Formation of Ultracold Molecules with Chirped Nanosecond Pulses”. en. In: *The Journal of Physical Chemistry A* 120.19 (2015), pp. 3032–3041. DOI: 10.1021/acs.jpca.5b10088 (cit. on p. 1).

[16] M. Tomza et al. “Cold hybrid ion-atom systems”. en. In: *Reviews of Modern Physics* 91.3 (2019). DOI: 10.1103/revmodphys.91.035001 (cit. on p. 1).

[17] A. Giussani and G. A. Worth. “On the Intrinsically Low Quantum Yields of Pyrimidine DNA Photodamages: Evaluating the Reactivity of the Corresponding Minimum Energy Crossing Points”. en. In: *The Journal of Physical Chemistry Letters* 11.13 (2020), pp. 4984–4989. DOI: 10.1021/acs.jpclett.0c01264 (cit. on p. 1).

[18] M. Weyland, S. S. Szigeti, R. A. B. Hobbs, P. Ruksasakchai, L. Sanchez, and M. F. Andersen. “Pair Correlations and Photoassociation Dynamics of Two Atoms in an Optical Tweezer”. en. In: *Physical Review Letters* 126.8 (2021). DOI: 10.1103/physrevlett.126.083401 (cit. on p. 1).

[19] D. Borsalino, B. Londoño-Florèz, R. Vexiau, O. Dulieu, N. Bouloufa-Maafa, and E. Luc-Koenig. “Efficient optical schemes to create ultracold KRb molecules in their rovibronic ground state”. en. In: *Physical Review A* 90.3 (2014). DOI: 10.1103/physreva.90.033413 (cit. on p. 1).

[20] D. Borsalino, R. Vexiau, M. Aymar, E. Luc-Koenig, O. Dulieu, and N. Bouloufa-Maafa. “Prospects for the formation of ultracold polar ground state KCs molecules via an optical process”. In: *Journal of Physics B: Atomic, Molecular and Optical Physics* 49.5 (2016), p. 055301. DOI: 10.1088/0953-4075/49/5/055301 (cit. on p. 1).

[21] N. Balakrishnan. “Perspective: Ultracold molecules and the dawn of cold controlled chemistry”. en. In: *The Journal of Chemical Physics* 145.15 (2016), p. 150901. DOI: 10.1063/1.4964096 (cit. on p. 1).

[22] A. Devolder, M. Desouter-Lecomte, O. Atabek, E. Luc-Koenig, and O. Dulieu. “Laser control of ultracold molecule formation: The case of RbSr”. In: *Phys. Rev. A* 103 (3 2021), p. 033301. DOI: 10.1103/PhysRevA.103.033301 (cit. on p. 1).

[23] K. H. Leung, E. Tiberi, B. Iritani, I. Majewska, R. Moszynski, and T. Zelevinsky. "Ultracold $^{88}Sr_2$ molecules in the absolute ground state". In: *New Journal of Physics* 23.11 (2021), p. 115002. DOI: 10.1088/1367-2630/ac2dac (cit. on p. 1).

[24] A. B. Alekseyev et al. "Spin–orbit effects in photodissociation of sodium iodide". en. In: *The Journal of Chemical Physics* 113.4 (2000), pp. 1514–1523. DOI: 10.1063/1.481938 (cit. on p. 1).

[25] E. Kokkonen et al. "Spin–orbit interaction mediated molecular dissociation". en. In: *The Journal of Chemical Physics* 140.18 (2014), p. 184304. DOI: 10.1063/1.4873718 (cit. on p. 1).

[26] M. McDonald et al. "Photodissociation of ultracold diatomic strontium molecules with quantum state control". en. In: *Nature* 535.7610 (2016), pp. 122–126. DOI: 10.1038/nature18314 (cit. on p. 1).

[27] A. Tóth, A. Csehi, G. J. Halász, and Á. Vibók. "Control of photodissociation with the dynamic Stark effect induced by THz pulses". In: *Phys. Rev. Research* 2 (1 2020), p. 013338. DOI: 10.1103/PhysRevResearch.2.013338 (cit. on p. 1).

[28] J. Kozicki, A. Gladky, and K. Thoeni. "Implementation of high-precision computation capabilities into the open-source dynamic simulation framework YADE". In: *Computer Physics Communications* 270 (2022), p. 108167. DOI: 10.1016/j.cpc.2021.108167 (cit. on pp. 1, 2, 8, 27, 47, 48, 52, 53, 67).

[29] P. Jasik, J. Kozicki, T. Kilich, J. E. Sienkiewicz, and N. E. Henriksen. "Electronic structure and rovibrational predissociation of the $2^1\Pi$ state in KLi". en. In: *Physical Chemistry Chemical Physics* 20.27 (2018), pp. 18663–18670. DOI: 10.1039/c8cp02551g (cit. on pp. 1, 49–51, 55, 56, 59, 63, 68).

[30] J. Kozicki and P. Jasik. "Optimization of the Femtosecond Laser Impulse for Photoassociation and the Power-Law Decay for the Spin-Orbit Mediated Dissociation in the NaRb Dimer". In: https://attochem-prague.troja.mff.cuni.cz/index.php. Annual Meeting of the COST action CA18222, Prague, 19-21 September (poster). 2022 (cit. on pp. 1, 2, 49, 59, 65, 68).

[31] P. Jasik, T. Kilich, J. Kozicki, and J. E. Sienkiewicz. "Potential energy surfaces of the low-lying electronic states of the Li+LiCs system". In: *Chemical Physics Letters* 695 (2018), pp. 119–124. ISSN: 0009-2614. DOI: 10.1016/j.cplett.2018.02.005 (cit. on pp. 1, 2, 68).

[32] P. Jasik, J. Franz, D. Kędziera, T. Kilich, J. Kozicki, and J. E. Sienkiewicz. "Spontaneous electron emission vs dissociation in internally hot silver dimer anions". en. In: *The Journal of Chemical Physics* 154.16 (2021), p. 164301. DOI: 10.1063/5.0046060 (cit. on pp. 1, 2, 68).

[33] J. Kozicki and F. Donzé. "A new open–source software developed for numerical simulations using discrete modeling methods". In: *Computer Methods in Applied Mechanics and Engineering* 197 (2008), pp. 4429–4443 (cit. on pp. 1, 52, 53).

[34] J. Kozicki and F. Donzé. "Yade–open DEM: an open–source software using a discrete element method to simulate granular material". In: *Engineering Computations* 26.7 (2009), pp. 786–805 (cit. on pp. 1, 52, 53).

[35] V. Smilauer et al. *Yade Documentation 3rd ed.* http://yade-dem.org/doc/. The Yade Project, 2021. DOI: 10.5281/zenodo.5705394 (cit. on pp. 1, 52, 53).

[36] F. Martin et al. *Memorandum of Understanding for the implementation of the COST Action "Attosecond Chemistry" (AttoChem) CA18222*. https://e-services.cost.eu/files/domain_files/CA/Action_CA18222/mou/CA18222-e.pdf. 2019 (cit. on p. 2).

[37] F. Martin, A. Palacios, and W. Rodríguez. *School on New Computational Methods for Attosecond Molecular Processes*. https://www.cecam.org/workshop-details/1058. 2021 (cit. on p. 2).

[38] R. Schinke. *Photodissociation dynamics*. Cambridge: Cambridge monographs on atomic, molecular and chemical physics, 1995 (cit. on pp. 2, 32, 52).

[39] D. J. Tannor. *Introduction to quantum mechanics: a time-dependent perspective*. University Science Books, Sausalito, 2007 (cit. on pp. 2, 32, 59, 60).

[40] M. Baer. *Beyond Born-Oppenheimer, Electronic Nonadiabatic Coupling Terms and Conical Intersections*. John Wiley & Sons, 2006 (cit. on pp. 2, 32, 33).

[41] I. Schaefer, H. Tal-Ezer, and R. Kosloff. "Semi-global approach for propagation of the time-dependent Schrödinger equation for time-dependent and nonlinear problems". In: *Journal of Computational Physics* 343 (2017), pp. 368–413. DOI: 10.1016/j.jcp.2017.04.017 (cit. on pp. 2, 5, 14, 19, 22, 26, 45–48, 59–61, 65, 67).

[42] E. Runge and E. K. U. Gross. "Density-Functional Theory for Time-Dependent Systems". en. In: *Physical Review Letters* 52.12 (1984), pp. 997–1000. DOI: 10.1103/physrevlett.52.997 (cit. on pp. 2, 19).

[43] A. Castro et al. "octopus: a tool for the application of time-dependent density functional theory". en. In: *physica status solidi (b)* 243.11 (2006), pp. 2465–2488. DOI: 10.1002/pssb.200642067 (cit. on pp. 2, 19).

[44] A. D. Becke. "Perspective: Fifty years of density-functional theory in chemical physics". en. In: *The Journal of Chemical Physics* 140.18 (2014), 18A301. DOI: 10.1063/1.4869598 (cit. on pp. 2, 19).

[45] E. Gross and W. Kohn. "Time-Dependent Density-Functional Theory". In: *Advances in Quantum Chemistry* (1990), pp. 255–291. DOI: 10.1016/s0065-3276(08)60600-0 (cit. on pp. 2, 19).

[46] N. Balakrishnan, C. Kalyanaraman, and N. Sathyamurthy. "Time–dependent quantum mechanical approach to reactive scattering and related processes". In: *Physics Reports* 280 (1997), pp. 79–144 (cit. on pp. 2, 19).

[47] M. Beck. "The multiconfiguration time-dependent Hartree (MCTDH) method: a highly efficient algorithm for propagating wavepackets". In: *Physics Reports* 324.1 (2000), pp. 1–105. DOI: 10.1016/s0370-1573(99)00047-2 (cit. on pp. 2, 19).

[48] K. C. Kulander. "Time-dependent Hartree-Fock theory of multiphoton ionization: Helium". In: *Physical Review A* 36.6 (1987), pp. 2726–2738. DOI: 10.1103/physreva.36.2726 (cit. on pp. 2, 19).

[49] U. Manthe. "The multi-configurational time-dependent Hartree approach revisited". In: *The Journal of Chemical Physics* 142.24 (2015), p. 244109. DOI: 10.1063/1.4922889 (cit. on pp. 2, 19).

[50] J. D. Durrant and J. A. McCammon. "Molecular dynamics simulations and drug discovery". en. In: *BMC Biology* 9.1 (2011). DOI: 10.1186/1741-7007-9-71 (cit. on p. 2).

[51] B. Schmidt and U. Lorenz. "WavePacket: A Matlab package for numerical quantum dynamics. I: Closed quantum systems and discrete variable representations". en. In: *Computer Physics Communications* 213 (2017), pp. 223–234. DOI: 10.1016/j.cpc.2016.12.007 (cit. on p. 2).

[52] B. Schmidt and C. Hartmann. "WavePacket: A Matlab package for numerical quantum dynamics.II: Open quantum systems, optimal control, and model reduction". en. In: *Computer Physics Communications* 228 (2018), pp. 229–244. DOI: 10.1016/j.cpc.2018.02.022 (cit. on p. 2).

[53] B. Schmidt, R. Klein, and L. Cancissu Araujo. "WavePacket: A Matlab package for numerical quantum dynamics. III. Quantum-classical simulations and surface hopping trajectories". en. In: *Journal of Computational Chemistry* 40.30 (2019), pp. 2677–2688. DOI: 10.1002/jcc.26045 (cit. on p. 2).

[54] H.-J. Werner et al. *MOLPRO, version 2012.1, a package of ab initio programs*. see http://www.molpro.net (cit. on p. 2).

[55] A. Scrinzi. "Infinite-range exterior complex scaling as a perfect absorber in time-dependent problems". In: *Physical Review A* 81.5 (2010). DOI: 10.1103/physreva.81.053845 (cit. on p. 2).

[56] A. Scrinzi. "t-SURFF: fully differential two-electron photo-emission spectra". In: *New Journal of Physics* 14.8 (2012), p. 085008. DOI: 10.1088/1367-2630/14/8/085008 (cit. on p. 2).

[57] A. Scrinzi. "tRecX – an environment for solving time-dependent Schrödinger-like problems". In: *Computer Physics Communications* (2021). (under review in Computer Physics Communications) (cit. on p. 2).

[58] V. J. Borràs, J. González-Vázquez, L. Argenti, and F. Martín. *Molecular-Frame Photoelectron Angular Distributions of CO in the Vicinity of Feshbach Resonances: An XCHEM Approach*. en. 2021. DOI: 10.1021/acs.jctc.1c00480 (cit. on p. 2).

[59] G. Guennebaud, B. Jacob, et al. *Eigen v3*. http://eigen.tuxfamily.org. 2010 (cit. on p. 3).

[60] B. Dawes et al. *Boost C++ Libraries*. https://www.boost.org/. 2020 (cit. on pp. 3, 32, 47).

[61] E. Q. Miranda, C. Kormanyos, and J. Kozicki. *Fast Fourier Transform in Boost libraries, GSoC Report*. https://github.com/BoostGSoC21/math-fft-report/releases/download/v1.1/gsoc-report.pdf. 2021 (cit. on pp. 3, 47, 67).

[62] E. Q. Miranda, C. Kormanyos, and J. Kozicki. *Boost Google Summer of Code 2021*. https://summerofcode.withgoogle.com/archive/2021/organizations/5123901926932480. 2021 (cit. on pp. 3, 47, 67).

[63] H. Tal-Ezer and R. Kosloff. "An accurate and efficient scheme for propagating the time dependent Schrödinger equation". en. In: *The Journal of Chemical Physics* 81.9 (1984), pp. 3967–3971. DOI: 10.1063/1.448136 (cit. on pp. 5, 52, 53, 69).

[64] C. Leforestier et al. “A comparison of different propagation schemes for the time dependent Schrödinger equation”. In: *Journal of Computational Physics* 94.1 (1991), pp. 59–80. DOI: 10.1016/0021-9991(91)90137-a (cit. on pp. 5, 10, 34, 57, 67).

[65] W. Kahan. *How Futile are Mindless Assessments of Roundoff in Floating-Point Computation?* https://people.eecs.berkeley.edu/~wkahan/Mindless.pdf. 2006 (cit. on pp. 6, 8).

[66] D. G. Hough, M. Cowlishaw, and et al. “IEEE Standard for Floating-Point Arithmetic”. In: *IEEE Std 754-2019 (Revision of IEEE 754-2008)* (2019), pp. 1–84. DOI: 10.1109/IEEESTD.2019.8766229 (cit. on p. 7).

[67] M. Blair, S. Obenski, and P. Bridickas. *Patriot missile defense software problem led to system failure at Dhahran, Saudi Arabia*. Tech. rep. https://www.gao.gov/assets/220/215614.pdf. United States General Accounting Office, 1992 (cit. on p. 8).

[68] J. L. Lions et al. *Ariane 5 flight 501 failure*. Tech. rep. https://esamultimedia.esa.int/docs/esa-x-1819eng.pdf. Report by the inquiry board, 1996 (cit. on p. 8).

[69] G. L. Lann. *The Ariane 5 Flight 501 Failure - A Case Study in System Engineering for Computing Systems*. Tech. rep. https://hal.inria.fr/inria-00073613. Research Report, RR-3079, INRIA. 1996. inria-00073613, 2006 (cit. on p. 8).

[70] J.-M. Jézéquel and B. Meyer. “Design by contract: The lessons of Ariane”. In: *Computer* 30 (1 1997). http://se.ethz.ch/~meyer/publications/computer/ariane.pdf, pp. 129–130. ISSN: 0018-9162. DOI: 10.1109/2.562936 (cit. on p. 8).

[71] M. Ben-Ari. “The Bug That Destroyed a Rocket”. In: *SIGCSE Bull.* 33.2 (2001), pp. 58–59. ISSN: 0097-8418. DOI: 10.1145/571922.571958 (cit. on p. 8).

[72] M. Abramowitz and I. Stegun. *Handbook of mathematical functions*. Dover publications, inc. New York (reprint), 2013 (cit. on pp. 10, 52, 69).

[73] R. Kosloff. “Quantum molecular dynamics on grids”. In: *Department of Physical Chemistry and the Fritz Haber Research Center* (1997) (cit. on pp. 16, 29, 52, 53).

[74] V. A. Mandelshtam and H. S. Taylor. “Spectral projection approach to the quantum scattering calculations”. In: *The Journal of Chemical Physics* 102.19 (1995), pp. 7390–7399. DOI: 10.1063/1.469051 (cit. on pp. 18, 52).

[75] V. A. Mandelshtam and H. S. Taylor. “A simple recursion polynomial expansion of the Green’s function with absorbing boundary conditions. Application to the reactive scattering”. In: *The Journal of Chemical Physics* 103.8 (1995), pp. 2903–2907. DOI: 10.1063/1.470477 (cit. on pp. 18, 30, 52).

[76] W. Bao, D. Jaksch, and P. A. Markowich. “Numerical solution of the Gross–Pitaevskii equation for Bose–Einstein condensation”. In: *Journal of Computational Physics* 187.1 (2003), pp. 318–342. ISSN: 0021-9991. DOI: 10.1016/S0021-9991(03)00102-5 (cit. on p. 19).

[77] D. Neuhauser and M. Baer. “The application of wave packets to reactive atom–diatom systems: a new approach”. In: *The Journal of chemical physics* 91.8 (1989), pp. 4651–4657 (cit. on p. 19).

[78] H. Tal-Ezer, R. Kosloff, and I. Schaefer. "New, highly accurate propagator for the linear and nonlinear Schrödinger equation". In: *Journal of Scientific Computing* 53.1 (2012), pp. 211–221 (cit. on p. 19).

[79] W. Magnus. "On the exponential solution of differential equations for a linear operator". In: *Communications on Pure and Applied Mathematics* 7.4 (1954), pp. 649–673. DOI: 10.1002/cpa.3160070404 (cit. on p. 19).

[80] Z. Sun, W. Yang, and D. H. Zhang. "Higher-order split operator schemes for solving the Schrödinger equation in the time-dependent wave packet method: applications to triatomic reactive scattering calculations". In: *Physical Chemistry Chemical Physics* 14.6 (2012), pp. 1827–1845 (cit. on p. 19).

[81] U. Peskin, R. Kosloff, and N. Moiseyev. "The solution of the time dependent Schrödinger equation by the (t, t') method: The use of global polynomial propagators for time dependent Hamiltonians". In: *The Journal of chemical physics* 100.12 (1994), pp. 8849–8855 (cit. on p. 19).

[82] H. Tal-Ezer. *Polynomial approximation of functions of matrices and applications*. 1989. DOI: 10.1007/bf01061265 (cit. on pp. 25, 42).

[83] J. Muga, J. Palao, B. Navarro, and I. Egusquiza. "Complex absorbing potentials". In: *Physics Reports* 395.6 (2004), pp. 357–426. DOI: 10.1016/j.physrep.2004.03.002 (cit. on p. 26).

[84] J. Palao and J. Muga. "A simple construction procedure of absorbing potentials". In: *Chemical Physics Letters* 292.1–2 (1998), pp. 1–6. DOI: 10.1016/s0009-2614(98)00635-6 (cit. on p. 26).

[85] D. Goldberg. "What every computer scientist should know about floating-point arithmetic". In: *ACM Computing Surveys* 23.1 (1991), pp. 5–48. DOI: 10.1145/103162.103163 (cit. on p. 27).

[86] D. Bailey, R. Barrio, and J. Borwein. "High-precision computation: Mathematical physics and dynamics". In: *Applied Mathematics and Computation* 218.20 (2012), pp. 10106–10121. ISSN: 0096-3003. DOI: https://doi.org/10.1016/j.amc.2012.03.087 (cit. on p. 27).

[87] W. Kahan. *On the Cost of Floating-Point Computation Without Extra-Precise Arithmetic*. https://people.eecs.berkeley.edu/~wkahan/Qdrtcs.pdf. 2004 (cit. on p. 27).

[88] K. Isupov. "Performance data of multiple-precision scalar and vector BLAS operations on CPU and GPU". In: *Data in Brief* 30 (2020), p. 105506. DOI: 10.1016/j.dib.2020.105506 (cit. on p. 27).

[89] L. Fousse, G. Hanrot, V. Lefèvre, P. Pélissier, and P. Zimmermann. "MPFR". In: *ACM Transactions on Mathematical Software* 33.2 (2007), p. 13. DOI: 10.1145/1236463.1236468 (cit. on p. 27).

[90] J. Laskar and M. Gastineau. "Existence of collisional trajectories of Mercury, Mars and Venus with the Earth". In: *Nature* 459.7248 (2009), pp. 817–819. DOI: 10.1038/nature08096 (cit. on p. 27).

[91] G. J. Sussman and J. Wisdom. "Chaotic Evolution of the Solar System". In: *Science* 257.5066 (1992), pp. 56–62. DOI: 10.1126/science.257.5066.56 (cit. on p. 27).

[92] P. H. Hauschildt and E. Baron. "Numerical solution of the expanding stellar atmosphere problem". In: *Journal of Computational and Applied Mathematics* 109.1–2 (1999), pp. 41–63. DOI: 10.1016/s0377-0427(99)00153-3 (cit. on p. 27).

[93] Y. He and C. H. Q. Ding. "Using accurate arithmetics to improve numerical reproducibility and stability in parallel applications". In: *The Journal of Supercomputing* 18.3 (2001), pp. 259–277. DOI: 10.1023/a:1008153532043 (cit. on p. 27).

[94] D. H. Bailey and A. M. Frolov. "Universal variational expansion for high-precision bound-state calculations in three-body systems. Applications to weakly bound, adiabatic and two-shell cluster systems". In: *Journal of Physics B: Atomic, Molecular and Optical Physics* 35.20 (2002), pp. 4287–4298. DOI: 10.1088/0953-4075/35/20/314 (cit. on p. 27).

[95] A. M. Frolov and D. H. Bailey. "Highly accurate evaluation of the few-body auxiliary functions and four-body integrals". In: *Journal of Physics B: Atomic, Molecular and Optical Physics* 37.4 (2004), pp. 955–955. DOI: 10.1088/0953-4075/37/4/c02 (cit. on p. 27).

[96] Z.-C. Yan and G. W. F. Drake. "Bethe Logarithm and QED Shift for Lithium". In: *Physical Review Letters* 91.11 (2003). DOI: 10.1103/physrevlett.91.113004 (cit. on p. 27).

[97] T. Zhang, Z.-C. Yan, and G. W. F. Drake. "QED Corrections of $O(mc^2\alpha^7\ln(\alpha))$ to the Fine Structure Splittings of Helium and He-like Ions". In: *Physical Review Letters* 77.9 (1996), pp. 1715–1718. DOI: 10.1103/physrevlett.77.1715 (cit. on p. 27).

[98] D. Broadhurst. "Massive 3-loop Feynman diagrams reducible to SC* primitives of algebras of the sixth root of unity". In: *The European Physical Journal C* 8.2 (1999), pp. 311–333. DOI: 10.1007/s100529900935 (cit. on p. 27).

[99] D. H. Bailey. "High-precision floating-point arithmetic in scientific computation". In: *Computing in Science and Engineering* 7.3 (2005), pp. 54–61 (cit. on p. 27).

[100] D. H. Bailey, K. Jeyabalan, and X. S. Li. "A Comparison of Three High-Precision Quadrature Schemes". In: *Experimental Mathematics* 14.3 (2005), pp. 317–329. DOI: 10.1080/10586458.2005.10128931 (cit. on p. 27).

[101] M. Lu, B. He, and Q. Luo. "Supporting Extended Precision on Graphics Processors". In: DaMoN '10 (2010), pp. 19–26. DOI: 10.1145/1869389.1869392 (cit. on p. 27).

[102] R. E. Caflisch. "Singularity formation for complex solutions of the 3D incompressible Euler equations". In: *Physica D: Nonlinear Phenomena* 67.1–3 (1993), pp. 1–18. DOI: 10.1016/0167-2789(93)90195-7 (cit. on p. 27).

[103] D. Bailey. "Integer relation detection". In: *Computing in Science and Engineering* 2.1 (2000), pp. 24–28. DOI: 10.1109/5992.814653 (cit. on p. 27).

[104] M. Joldes, V. Popescu, and W. Tucker. "Searching for Sinks for the Hénon Map using a Multipleprecision GPU Arithmetic Library". In: *ACM SIGARCH Computer Architecture News* 42.4 (2014), pp. 63–68. DOI: 10.1145/2693714.2693726 (cit. on p. 27).

[105] A. Abad, R. Barrio, and A. Dena. "Computing periodic orbits with arbitrary precision". In: *Physical Review E* 84.1 (2011). DOI: 10.1103/physreve.84.016701 (cit. on p. 27).

[106] T. P. Stefański. "Electromagnetic Problems Requiring High-Precision Computations". In: *IEEE Antennas and Propagation Magazine* 55.2 (2013), pp. 344–353 (cit. on p. 27).

[107] K. Pachucki and M. Puchalski. "Extended Hylleraas three-electron integral". In: *Physical Review A* 71.3 (2005). DOI: 10.1103/physreva.71.032514 (cit. on p. 27).

[108] M. Siłkowski and K. Pachucki. "Long-range asymptotics of exchange energy in the hydrogen molecule". In: *The Journal of Chemical Physics* 152.17 (2020), p. 174308. DOI: 10.1063/5.0008086 (cit. on p. 27).

[109] C. Pralavorio. *Welcome to the precision era*. https://home.cern/news/series/lhc-physics-ten/welcome-precision-era. 2020 (cit. on p. 27).

[110] J. C. Tully. "Molecular dynamics with electronic transitions". In: *The Journal of Chemical Physics* 93.2 (1990), pp. 1061–1071. DOI: 10.1063/1.459170 (cit. on pp. 32–34, 41).

[111] D. F. Coker. "Computer Simulation Methods for Nonadiabatic Dynamics in Condensed Systems". In: *Computer Simulation in Chemical Physics* (1993), pp. 315–377. DOI: 10.1007/978-94-011-1679-4_9 (cit. on pp. 32–34, 41).

[112] A. Quarteroni, F. Saleri, and P. Gervasio. *Scientific Computing with MATLAB and Octave*. Springer Berlin Heidelberg, 2014. DOI: 10.1007/978-3-642-45367-0 (cit. on p. 48).

[113] W. Jastrzębski, R. Kowalczyk, A. Pashov, and J. Szczepkowski. "The potential energy barrier of the 2(1) Pi state in KLi". In: *Spectrochimica Acta Part A-Molecular and Biomolecular Spectroscopy* 73.1 (2009), 117–120. ISSN: 1386-1425. DOI: 10.1016/j.saa.2009.02.002 (cit. on pp. 49, 55, 58).

[114] A. Grochola, W. Jastrzębski, P. Kowalczyk, P. Crozet, and A. Ross. "The molecular constants and potential energy curve of the D-1 Pi state in KLi". In: *Chemical Physics Letters* 372.1-2 (2003), 173–178. ISSN: 0009-2614. DOI: 10.1016/S0009-2614(03)00396-8 (cit. on pp. 49, 50, 55).

[115] S. A. Moses, J. P. Covey, M. T. Miecnikowski, D. S. Jin, and J. Ye. "New frontiers for quantum gases of polar molecules". In: *Nature Physics* 13.1 (2017), 13–20. ISSN: 1745-2473 (cit. on p. 49).

[116] M. Wiatr, P. Jasik, T. Kilich, J. E. Sienkiewicz, and H. Stoll. "Quasirelativistic potential energy curves and transition dipole moments of NaRb". en. In: *Chemical Physics* 500 (2018), pp. 80–87. DOI: 10.1016/j.chemphys.2017.10.005 (cit. on pp. 50, 59).

[117] M. Chaieb, H. Habli, L. Mejrissi, A. A. Al-Ghamdi, B. Oujia, and F. X. Gadea. "Diabatic investigation for the NaRb molecule". In: *International Journal of Quantum Chemistry* 117.22 (2017), e25437. ISSN: 0020-7608. DOI: 10.1002/qua.25437 (cit. on p. 50).

[118] J. M. Walter and S. Barrat. In: *Proc. R. Soc.* 119 (1928), p. 257 (cit. on p. 50).

[119] W. Weizel and M. Kulp. In: *Ann. Phys.Leipzig* 4 (1930), p. 971 (cit. on p. 50).

[120] K. F. Zmbov, C. H. Wu, and H. R. Ihle. "A mass spectrometric study of heteronuclear diatomic alkali metal molecules. Dissociation energies and ionization potentials of NaLi, KLi, and NaK." en. In: *The Journal of Chemical Physics* 67.10 (1977), pp. 4603–4607. DOI: 10.1063/1.434621 (cit. on p. 50).

[121] F. Engelke, H. Hage, and U. Sprick. "The B 1Π-X 1Σ+ system of $^{39}K^{6}Li$ and $^{39}K^{7}Li$: High-resolution laser excitation and fluorescence spectroscopy using selectively detected laser-induced fluorescence (SDLIF) in molecular beam and injection heat pipe (IHP)". en. In: *Chemical Physics* 88.3 (1984), pp. 443–453. DOI: 10.1016/0301-0104(84)87010-x (cit. on p. 50).

[122] F. Sievers et al. "Simultaneous sub-Doppler laser cooling of fermionic Li-6 and K-40 on the D-1 line: Theory and experiment". In: *Physical Review A* 91.2 (2015), p. 023426. ISSN: 2469-9926. DOI: 10.1103/PhysRevA.91.023426 (cit. on p. 50).

[123] E. A. Pazyuk, A. V. Zaitsevskii, A. V. Stolyarov, M. Tamanis, and R. Ferber. "Laser synthesis of ultracold alkali metal dimers: optimization and control". In: *Russian Chemical Reviews* 84.10 (2015), 1001–1020. ISSN: 0036-021X. DOI: 10.1070/RCR4534 (cit. on p. 50).

[124] W. Jastrzębski, P. Kowalczyk, and A. Pashov. "The perturbation of the B-1 Pi and C-1 Sigma+ states in KLi". In: *Journal of Molecular Spectroscopy* 209.1 (2001), 50–56. ISSN: 0022-2852. DOI: 10.1006/jmsp.2001.8389 (cit. on p. 50).

[125] A. Grochola, P. Kowalczyk, W. Jastrzębski, P. Crozet, and A. Ross. "The C-1 Sigma(+) state of KLi studied by polarisation labelling spectroscopy technique". In: *Acta Physica Polonica A* 102.6 (2002), 729–738. ISSN: 0587-4246 (cit. on p. 50).

[126] H. Salami, A. J. Ross, P. Crozet, W. Jastrzębski, P. Kowalczyk, and R. J. Le Roy. "A full analytic potential energy curve for the a (3)Sigma(+) state of KLi from a limited vibrational data set". In: *Journal of Chemical Physics* 126.19 (2007), p. 194313. ISSN: 0021-9606. DOI: 10.1063/1.2734973 (cit. on p. 50).

[127] A. Adohi-Krou, W. Jastrzębski, P. Kowalczyk, A. V. Stolyarov, and A. J. Ross. "Investigation of the D (1)Pi state of NaK by polarisation labelling spectroscopy". In: *Journal of Molecular Spectroscopy* 250.1 (2008), 27–32. ISSN: 0022-2852. DOI: 10.1016/j.jms.2008.04.003 (cit. on p. 50).

[128] W. Jastrzębski, P. Kowalczyk, and J. Szczepkowski. "Investigation of highly excited electronic (1)Pi states in KLi molecule". In: *Chemical Physics Letters* 666 (2016), 19–21. ISSN: 0009-2614. DOI: 10.1016/j.cplett.2016.10.078 (cit. on p. 50).

[129] E. Wille et al. "Exploring an ultracold fermi-fermi mixture: Interspecies Feshbach resonances and scattering properties of Li-6 and K-40". In: *Physical Review Letters* 100.5 (2008), p. 053201. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.100.053201 (cit. on p. 50).

[130] A. C. Voigt et al. "Ultracold Heteronuclear Fermi-Fermi Molecules". In: *Physical Review Letters* 102.2 (2009), p. 020405. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.102.020405 (cit. on p. 50).

[131] A. Ridinger et al. “Large atom number dual-species magneto-optical trap for fermionic Li-6 and K-40 atoms”. In: *European Physical Journal D* 65.1-2 (2011), 223–242. ISSN: 1434-6060. DOI: 10.1140/epjd/e2011-20069-4 (cit. on p. 50).

[132] W. Muller and W. Meyer. “Ground-state properties of alkali dimers and their cations (including the elements Li, Na, and K) from abinitio calculations with effective core polarization potentials”. In: *Journal of Chemical Physics* 80.7 (1984), 3311–3320. ISSN: 0021-9606 (cit. on p. 50).

[133] S. Rousseau, A. Allouche, M. Aubert-Frecon, S. Magnier, P. Kowalczyk, and W. Jastrzębski. “Theoretical study of the electronic structure of KLi and comparison with experiments”. In: *Chemical Physics* 247.2 (1999), 193–199. ISSN: 0301-0104. DOI: 10.1016/S0301-0104(99)00206-2 (cit. on p. 50).

[134] B. Huron, J. Malrieu, and P. Rancurel. “Iterative perturbation calculations of ground and excited-state energies from multiconfigurational zeroth-order wavefunctions”. In: *Journal of Chemical Physics* 58.12 (1973), 5745–5759. ISSN: 0021-9606. DOI: 10.1063/1.1679199 (cit. on p. 50).

[135] R. Dardouri, H. Habli, B. Oujia, and F. X. Gadea. “Theoretical study of the electronic structure of KLi molecule: Adiabatic and diabatic potential energy curves and dipole moments”. In: *Chemical Physics* 399 (2012), 65–79. ISSN: 0301-0104. DOI: 10.1016/j.chemphys.2011.07.010 (cit. on p. 50).

[136] K.-L. Xiao, C.-L. Yang, M.-S. Wang, X.-G. Ma, and W.-W. Liu. “The effect of inner-shell electrons on the ground and low-lying excited states of KLi: Ab initio study with all-electron basis sets”. In: *Journal of Quantitative Spectroscopy & Radiative Transfer* 129 (2013), 8–14. ISSN: 0022-4073. DOI: 10.1016/j.jqsrt.2013.05.025 (cit. on p. 50).

[137] L. Miadowicz, P. Jasik, and J. E. Sienkiewicz. “Possible schemes of photoassociation processes in the KLi molecule with newly calculated potential energy curves”. In: *Central European Journal of Physics* 11.9 (2013), 1115–1122. ISSN: 1895-1082. DOI: 10.2478/s11534-013-0199-z (cit. on p. 50).

[138] O. M. Al-dossary and N. Khelifi. “Theoretical study of LiK and LiK+ in adiabatic representation”. In: *Russian Journal of Physical Chemistry A* 88.1 (2014), 73–84. ISSN: 0036-0244. DOI: 10.1134/S0036024414010336 (cit. on p. 50).

[139] M. Musial, A. Motyl, P. Skupin, and S. A. Kucharski. “Potential Energy Curves for the Low-Lying Electronic States of KLi with Fock Space Coupled Cluster Method”. In: *Concepts of Mathematical Physics in Chemistry: a tribute to Frank E. Harris, pt B*. Ed. by Sabin, JR and CabreraTrujillo, R. Vol. 72. Advances in Quantum Chemistry. Workshop on Concepts of Mathematical Physics in Chemistry in Honor of Frank E. Harris, Playa del Carmen, Mexico, DEC 10-13, 2014. 2016, 201–216. ISBN: 978-0-12-803984-7. DOI: 10.1016/bs.aiq.2015.05.001 (cit. on p. 50).

[140] P. Jasik, J. Kozicki, T. Kilich, J. Sienkiewicz, and N. Henriksen. *Potential energy curve, rovibrational energies and nuclear wave functions of 2 singlet Pi state in KLi dimer*. en. https://mostwiedzy.pl/en/open-research-data/potential-energy-curve-rovibrational-energies-and-nuclear-wave-functions-of-2-singlet-pi-state-in-kli-dimer,616015246646282-0. 2020. DOI: 10.34808/3XZS-S638 (cit. on pp. 50, 54, 56).

[141] R. J. Le Roy. "LEVEL: A computer program for solving the radial Schrodinger equation for bound and quasibound levels". English. In: *J. Quant. Spectrosc. Radiat. Transfer* 186.SI (2017), p. 167. ISSN: 0022-4073. DOI: {10.1016/j.jqsrt2016.05.028} (cit. on pp. 51–54).

[142] M. Frigo and S. G. Johnson. "The Design and Implementation of FFTW3". In: *Proceedings of the IEEE* 93.2 (2005). Special issue on "Program Generation, Optimization, and Platform Adaptation", pp. 216–231 (cit. on p. 52).

[143] R. Kosloff. "Time–Dependent Quantum–Mechanical Methods for Molecular Dynamics". In: *J. Phys. Chem.* 92 (1988), pp. 2087–2100 (cit. on pp. 52, 53).

[144] N. E. Henriksen and F. Y. Hansen. *Theories of Molecular Reaction Dynamics: The Microscopic Foundation of Chemical Kinetics*. New York: Oxford University Press Inc., 2008 (cit. on p. 52).

[145] V. Bednarska, I. Jackowska, W. Jastrzębski, and P. Kowalczyk. "The Molecular Constants and Potential Energy Curve of the Ground State $X^1\Sigma^+$ in KLi". In: *Journal of Molecular Spectroscopy* 189.MS987543 (1998), pp. 244–248 (cit. on p. 54).

[146] A. Grochola. *Investigation of the electronic structure of alkali metal dimers using the polarization spectroscopy technique*. PhD Thesis. 2004 (cit. on p. 55).

[147] H. Schmidt et al. "Predissociation dynamics of lithium iodide". en. In: *The Journal of Chemical Physics* 142.4 (2015), p. 044303. DOI: 10.1063/1.4906512 (cit. on p. 59).

[148] J. Kong et al. "Revisiting the Luminescence Decay Kinetics of Energy Transfer Upconversion". en. In: *The Journal of Physical Chemistry Letters* 11.9 (2020), pp. 3672–3680. DOI: 10.1021/acs.jpclett.0c00619 (cit. on p. 59).

[149] M. Gruebele. "Intramolecular vibrational dephasing obeys a power law at intermediate times". en. In: *Proceedings of the National Academy of Sciences* 95.11 (1998), pp. 5965–5970. DOI: 10.1073/pnas.95.11.5965 (cit. on p. 59).

[150] Y. Mizuno and K. Hukushima. "Power-law decay in the nonadiabatic photodissociation dynamics of alkali halides due to quantum wavepacket interference". en. In: *The Journal of Chemical Physics* 149.17 (2018), p. 174313. DOI: 10.1063/1.5048957 (cit. on pp. 59, 60, 63, 64).

[151] J. K. Utterback, J. L. Ruzicka, H. Hamby, J. D. Eaves, and G. Dukovic. "Temperature-Dependent Transient Absorption Spectroscopy Elucidates Trapped-Hole Dynamics in CdS and CdSe Nanorods". en. In: *The Journal of Physical Chemistry Letters* 10.11 (2019), pp. 2782–2787. DOI: 10.1021/acs.jpclett.9b00764 (cit. on p. 59).

[152] Y. Dong, L. Dong, M. Gong, and H. Pu. "Dynamical phases in quenched spin–orbit-coupled degenerate Fermi gas". en. In: *Nature Communications* 6.1 (2015). DOI: 10.1038/ncomms7103 (cit. on p. 59).

[153] M. Tamanis et al. "High resolution spectroscopy and channel-coupling treatment of the $A^1\Sigma^+ - b^3\Pi$ complex of NaRb". en. In: *The Journal of Chemical Physics* 117.17 (2002), pp. 7980–7988. DOI: 10.1063/1.1505442 (cit. on p. 59).

[154] O. Docenko, M. Tamanis, R. Ferber, A. Pashov, H. Knöckel, and E. Tiemann. "Potential of the ground state of NaRb". In: *Phys. Rev. A* 69 (4 2004), p. 042503. DOI: 10.1103/PhysRevA.69.042503 (cit. on p. 59).

[155] A. Pashov, W. Jastrzebski, P. Kortyka, and P. Kowalczyk. "Experimental long range potential of the $B^1\Pi$ state in NaRb". en. In: *The Journal of Chemical Physics* 124.20 (2006), p. 204308. DOI: 10.1063/1.2198199 (cit. on p. 59).

[156] O. Docenko et al. "Deperturbation treatment of the $A\,^1\Sigma^+ - -b\,^3\Pi$ complex of NaRb and prospects for ultracold molecule formation in $X\,^1\Sigma^+(v=0;J=0)$". In: *Phys. Rev. A* 75 (4 2007), p. 042503. DOI: 10.1103/PhysRevA.75.042503 (cit. on p. 59).

[157] Z. Guo, F. Jia, B. Zhu, L. Li, J. M. Hutson, and D. Wang. "Improved characterization of Feshbach resonances and interaction potentials between ^{23}Na and ^{87}Rb atoms". In: *Phys. Rev. A* 105 (2 2022), p. 023313. DOI: 10.1103/PhysRevA.105.023313 (cit. on p. 59).

[158] M. Korek et al. "Theoretical study of the electronic structure of the LiRb and NaRb molecules". en. In: *Chemical Physics* 256.1 (2000), pp. 1–6. DOI: 10.1016/s0301-0104(00)00061-6 (cit. on p. 59).

[159] M. Chaieb, H. Habli, L. Mejrissi, B. Oujia, and F. X. Gadéa. "Ab initio spectroscopic study for the NaRb molecule in ground and excited states". en. In: *International Journal of Quantum Chemistry* 114.11 (2014), pp. 731–747. DOI: 10.1002/qua.24664 (cit. on p. 59).

[160] M. Wiatr, P. Jasik, and J. E. Sienkiewicz. "The adiabatic potentials of low-lying electronic states of the NaRb molecule". In: *Physica Scripta* 90.5 (2015), p. 054012. DOI: 10.1088/0031-8949/90/5/054012 (cit. on pp. 59–61, 65).

[161] M. Guo et al. "Creation of an Ultracold Gas of Ground-State Dipolar ^{23}Na^{87}Rb Molecules". en. In: *Physical Review Letters* 116.20 (2016). DOI: 10.1103/physrevlett.116.205303 (cit. on pp. 59, 61).

[162] M. Guo et al. "High-resolution molecular spectroscopy for producing ultracold absolute-ground-state ^{23}Na^{87}Rb molecules". en. In: *Physical Review A* 96.5 (2017). DOI: 10.1103/physreva.96.052505 (cit. on pp. 59, 61).

[163] K. B. Møller, N. E. Henriksen, and A. H. Zewail. "On the role of coherence in the transition from kinetics to dynamics: Theory and application to femtosecond unimolecular reactions". en. In: *The Journal of Chemical Physics* 113.23 (2000), pp. 10477–10485. DOI: 10.1063/1.1323729 (cit. on p. 63).

[164] M. Shapiro. "A Uniform Theory of Preparation, Dissociation, and Product Formation in the Decay of Overlapping Resonances". en. In: *The Journal of Physical Chemistry A* 102.47 (1998), pp. 9570–9576. DOI: 10.1021/jp982182d (cit. on p. 63).

[165] V. Brems and M. Desouter-Lecomte. "Memory kernels and effective Hamiltonians from time-dependent methods. II. Vibrational predissociation". en. In: *The Journal of Chemical Physics* 116.19 (2002), p. 8318. DOI: 10.1063/1.1467053 (cit. on p. 63).

[166] J. Orear. "Least squares when both variables have uncertainties". en. In: *American Journal of Physics* 50.10 (1982), pp. 912–916. DOI: 10.1119/1.12972 (cit. on p. 64).

List of my publications

Journal publications related to this PhD thesis

[1] J. Kozicki, P. Jasik, T. Kilich, J.E. Sienkiewicz, Optimization of the Femtosecond Laser Impulse for Photoassociation and the Power-Law Decay for the Spin-Orbit Mediated Dissociation in the NaRb Dimer, *Physical Chemistry Chemical Physics (submitted)*, **2022**.[100 points, IF:3.94]

[2] J. Kozicki, A. Gladky, K. Thoeni, Implementation of high-precision computation capabilities into the open-source dynamic simulation framework YADE, *Computer Physics Communications*, DOI: 10.1016/j.cpc.2021.108167, **2022**.[140 points, IF:4.71]

[3] P. Jasik, J.Franz, D. Kędziera, T. Kilich, J. Kozicki, J.E. Sienkiewicz, Spontaneous electron emission vs dissociation in internally hot silver dimer anions, *Journal of Chemical Physics*, DOI: 10.1063/5.0046060, **2021**.[100 points, IF:4.30]

[4] P. Jasik, J. Kozicki, T. Kilich, J.E. Sienkiewicz, N. E. Henriksen, Electronic structure and rovibrational predissociation of the $2^1\Pi$ state in KLi, *Physical Chemistry Chemical Physics*, DOI: 10.1039/c8cp02551g, **2018**.[100 points, IF:3.94]

[5] P. Jasik, T. Kilich, J. Kozicki, J.E. Sienkiewicz, Potential energy surfaces of the low–lying electronic states of the Li+LiCs system, *Chemical Physics Letters*, DOI: 10.1016/j.cplett.2018.02.005, **2018**.[70 points, IF:2.71]

Conference materials related to this PhD thesis

[6] J. Kozicki, P. Jasik, Optimization of the Femtosecond Laser Impulse for Photoassociation and the Power-Law Decay for the Spin-Orbit Mediated Dissociation in the NaRb Dimer, *Annual Meeting of the COST action CA18222*, Prague, 19-21 September, **2022**.

[7] J. Kozicki, Quantum dynamical simulations of the process of creation of ultracold NaRb molecules, *FoKA Workshop: Photons, Quanta, Atoms*, 11–13 May, **2022**.

[8] P. Jasik, J. Kozicki, T. Kilich, J.E. Sienkiewicz, Deperturbation analysis of the polar alkali diatomic molecules spectra, *AttoChem: First Annual Workshop for the Cost Action CA18222*, Faculty of Physics, Babes–Bolyai University Cluj–Napoca, 9–11 September 2020, on–line conference , **2020**.

[9] P. Jasik, J. Kozicki, T. Kilich, J.E. Sienkiewicz, N.E. Henriksen, Dynamics of rotational predissociation of KLi, *AttoChem: First Annual Workshop for the Cost Action CA18222*, Faculty of Physics, Babes–Bolyai University Cluj–Napoca, 9–11 September 2020, on–line conference , **2020**.

[10] D. Kędziera, P. Jasik, J. Kozicki, T. Kilich, J.E. Sienkiewicz, Ag_2 and Ag_2^- — potential curves: pseudopotential-based basis set considerations, *Waves in inhomogeneous media and integrable system*, Kaliningrad Immanuel Kant Baltic Federal University, 24–25 September 2020, on–line conference , **2020**.

[11] J. Kozicki, Rovibrational predissociation dynamics: spectrum and population decay of selected vibrational levels of the KLi molecule, *Modern Numerical Methods for Quantum Mechanics II*, University of Gdańsk and Polish Academy of Sciences, 27–29 June, **2018**.

Journal publications from my earlier career

[12] M. Krzaczek, M. Nitka, J. Kozicki, J. Tejchman, Simulations of hydro-fracking in rock mass at meso-scale using fully coupled DEM/CFD approach, *Acta Geotechnica*, DOI: 10.1007/s11440-019-00799-6, **2019**.[140 points, IF:5.57]

[13] J. Kozicki, J. Tejchman, Relationship between vortex structures and shear localization in 3D granular specimens based on combined DEM and Helmholtz–Hodge decomposition, *Granular Matter*, DOI: 10.1007/s10035-018-0815-0, **2018**.[100 points, IF:3.01]

[14] J. Kozicki, J. Tejchman, Investigations of quasi–static vortex structures in 2D sand specimen under passive earth pressure conditions based on DEM and Helmholtz–Hodge vector field decomposition, *Granular Matter*, DOI: 10.1007/s10035-017-0714-9, **2017**.[100 points, IF:3.01]

[15] J. Kozicki, J. Tejchman, DEM investigations of two–dimensional granular vortex– and anti–vortex structures during plane strain compression. *Granular Matter*, DOI: 10.1007/s10035-016-0627-z, **2016**.[100 points, IF:3.01]

[16] M. Nitka, J. Tejchman, J. Kozicki, D. Leśniewska, DEM analysis of micro–structural events within granular shear zones under passive earth pressure conditions. *Granular Matter*, DOI: 10.1007/s10035-015-0558-0, **2015**.[100 points, IF:3.01]

[17] J. Kozicki, J. Tejchman, H.B. Muhlhaus, Discrete simulations of a triaxial compression test for sand by DEM. *International Journal for Numerical and Analytical Methods in Geomechanics*, DOI: 10.1002/nag.2285, **2014**.[140 points, IF:4.22]

[18] Ł. Skarżyński, J. Kozicki, J. Tejchman, Application of DIC Technique to Concrete–Study on Objectivity of Measured Surface Displacements. *Experimental Mechanics*, DOI: 10.1007/s11340-013-9781-y, **2013**.[100 points, IF:2.79]

[19] M. Niedostatkiewicz, J. Kozicki, J. Tejchman, H.B. Muhlhaus, Discrete modelling results of a direct shear test for granular materials versus FE results. *Granular Matter*, DOI: 10.1007/s10035-013-0423-y, **2013**.[100 points, IF:3.01]

[20] J. Kozicki, J. Tejchman, Z. Mróz, Effect of grain roughness on strength, volume changes, elastic and dissipated energies during quasi-static homogeneous triaxial compression using DEM. *Granular Matter*, DOI: 10.1007/s10035-012-0352-1, **2012**. 100 points, IF:3.01

[21] J. Kozicki, J. Kozicka, Human Friendly Architectural Design for a small Martian Base. ***Advances in Space Research*** Vol. 48, 15 December 2011, pages 1997–2004, DOI: 10.1016/j.asr.2011.08.032, **2011**.[70 points, IF:2.61]

[22] Ł. Widuliński, J. Kozicki, J. Tejchman, D. Leśniewska, Discrete simulations of shear zone patterning in sand in earth pressure problems of a retaining wall. *International Journal of Solids and Structures*, DOI: 10.1016/j.ijsolstr.2011.01.005, **2011**. 140 points, IF:3.66

[23] J. Kozicki, J. Tejchman, Effect of steel fibres on concrete behavior in 2D and 3D simulations using lattice model. *Archives of Mechanics*, **2010**.[100 points, IF:1.18]

[24] J. Kozicki, F. V. Donzé, Yade–open DEM: an open–source software using a discrete element method to simulate granular material, *Engineering Computations*, DOI: 10.1108/02644400910985170, **2009**.[70 points, IF:1.67]

[25] Ł. Widuliński, J. Kozicki, J. Tejchman, Numerical simulations of triaxial test with sand using DEM *Archives of Hydro–Engineering and Environmental Mechanics*, **2009**.[20 pts]

[26] J. Kozicki, F. V. Donzé, A new open–source software developed for numerical simulations using discrete modeling methods. *Computer Methods in Applied Mechanics and Engineering*, DOI: 10.1016/j.cma.2008.05.023, **2008**.[200 points, IF:6.58]

[27] J. Kozicki, J. Tejchman, Modelling of fracture process in concrete using a novel lattice model. *Granular Matter*, DOI: 10.1007/s10035-008-0104-4, **2008**.[100 points, IF:3.01]

[28] J. Kozicki, J. Tejchman, Effect of aggregate structure on fracture process in concrete using 2D lattice model. *Archives of Mechanics*, Vol. 59, No 4–5, pages 365–384,, **2007**.[100 points, IF:1.18]

[29] J. Kozicki, J. Tejchman, Experimental investigations of strain localization in concrete using Digital Image Correlation (DIC) technique. *Archives of Hydro–Engineering and Environmental Mechanics*, Vol. 54, No 1, pages 3–24, **2007**.[20 points]

[30] J. Kozicki, J. Tejchman, 2D Lattice Model for Fracture in Brittle Materials. *Archives of Hydro–Engineering and Environmental Mechanics*, Vol. 53, No 2, pages 71–88, **2006**.[20 points]

[31] J. Kozicki, J. Tejchman, Application of a cellular automaton to simulations of granular flow in silos. *Granular Matter*, Vol. 7, No 1, pages 45–54, DOI: 10.1007/s10035-004-0190-x, **2005**.[100 points, IF:3.01]

[32] J. Kozicki, J. Tejchman, Simulations of Behavior of Granular Bodies using a Lattice Gas Automaton. *Archives of Hydro–Engineering and Environmental Mechanics*, Vol. 52, No 1, pages 21–37, **2005**.[20 points]

[33] J. Kozicki, J. Tejchman, Simulations of flow pattern in silos with a cellular automaton, part I. *Task Quarterly, Gdansk University of Technology*, No 1, **2005**.

[34] J. Kozicki, J. Tejchman, Simulations of flow pattern in silos with a cellular automaton, part II. *Task Quarterly, Gdansk University of Technology*, No 2, **2005**.

[35] J. Kozicki, Discrete lattice model used to describe the fracture process of concrete. *Discrete Element Group for Risk Mitigation Annual Report 1, Grenoble University of Joseph Fourier, France*, pages 95–101, **2005**.

[36] J. Kozicki, J. Tejchman, Application of a cellular automata model to granular flow. *Task Quarterly, Gdansk University of Technology*, Vol. 6, No 3, pages 429–436, **2002**.

[37] J. Kozicki, J. Tejchman, Simulations of granular flow in silos with a cellular automata model. *International Journal of Storing, Handling and Processing Powder*, pages 267–275, **2001**.

Books

[38] J. Tejchman, J. Kozicki, Experimental and Theoretical Investigations of Steel–Fibrous Concrete. *Springer Series in Geomechanics & Geoengineering*, **2010**.

[39] J. Tejchman, J. Kozicki, Steel–fibrous concrete, experiments and a numerical model. *Gdansk University of Technology Publishers*, **2009**.

[40] J. Kozicki, Application of Discrete Models to Describe the Fracture Process in Brittle Materials. *Gdansk University of Technology*, PhD thesis, **2007**.