— Bachelor Thesis —

Improving Simulation of Two-Dimensional Infrared and Visible Spectra

Floris Westerman, S2970627

Supervisor: dr. Thomas la Cour Jansen
Second Examiner: prof.dr. Richard Hildner

Abstract

This thesis explores the use of parallelisation to further improve the performance of the simulation of the NISE scheme. A hybrid approach using both MPI and OpenMP was used to obtain excellent scaling of the computation across many machines. This new code allowed for the computation of detailed two-dimensional spectra of C8S3, with which the effect of disorder and polarisation orientation on the system was investigated. The results show a good correspondence between theory and experiment, while also identifying an unexpected anti-diagonal peak shape that is not yet understood.

July 31, 2020

Contents

Chapter 1

Introduction

In the wide field of nanotechnology, one topic of particular importance is molecular self-assembly. Self-assembling complexes are common in nature: for example, they can be found in the systems responsible for photosynthesis in plants [1] and bacteria [2] where they convert light to chemical energy with high efficiency. Consequently, such systems have a wide range of possible applications, and building synthetic variants of these systems can open up an entire new field of nanotechnology.

Amphiphilic cyanine dyes are a specific class of complexes that exhibit self-assembling behaviour, forming nanotubes similar in structure to the photosynthesis complexes in nature. These tubes typically consist of two walls, formed by amphiphilic molecules: molecules with both hydrophobic and hydrophilic sides. The molecules form a long chain where the hydrophobic sides are oriented inwards, see figure 5.1. In this thesis, we will focus on the C8S3 dye [3–5] in particular.

The light-harvesting behaviour of these dyes can be attributed to the behaviour of excitons in the material. Incoming light causes excitations in the tubes that can travel between molecules due to strong coupling between them. From previous research [6], it turns out that the excitons travel mostly in the inner wall of the tube.

In order to investigate the behaviour of excitons on the inner walls, two-dimensional spectroscopy is used. Experiments with this technique have already been performed [6–9], so in this thesis we will focus on simulating C8S3 complexes to build a theoretical model.

The aim of this thesis is to improve an existing code library that employs the *Numerical Integration of the Schrödinger Equation* approach [10], as the current library [11] does not support parallel computation. Consequently, the complexity of C8S3 causes the simulations with the current code to be restricted in accuracy. This problem will be solved by parallelising the code, allowing spreading the calculation over multiple nodes in a supercomputer cluster, and enabling us to obtain and interpret accurate and realistic spectra of the complexes.

The application of the parallelised code will not be limited to the C8S3 complex, but can be used for further analysis of, e.g., the LHCII complexes in plants [2], the chlorosomes in green sulphur bacteria [1], and the LH2 complex in purple bacteria [12–14]. The new code can also be integrated into the existing library to allow for general usage.

Chapter 2

Multidimensional Spectroscopy

As described before, two-dimensional spectroscopy is employed for the analysis of exciton behaviour. Multidimensional spectroscopy offers additional information compared to more 'traditional' one-dimensional techniques [15]. For a single dimension, various physical phenomena could manifest themselves similarly, or identically, in the spectrum. For many cases, multidimensional spectroscopy allows to distinguish between these phenomena more easily.

This chapter is devoted to the theory of multidimensional spectroscopy; we will explore the formal aspects in a mathematical sense, and discuss how the obtained spectra can be interpreted. The relevant simulation techniques will be discussed in chapter 3.

2.1 Qualitative description

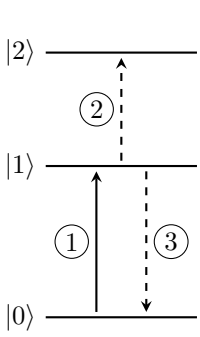
Before diving into the mathematical formalism, and in order to be able to interpret the results, it is important to have a good qualitative mental model for the functioning of multidimensional spectroscopy. In particular, we will focus on transient spectroscopy and its two-dimensional extension, as those methods are used throughout this thesis. The underlying ideas and principles can be extended to other forms of spectroscopy as well.

2.1.1 Transient spectroscopy

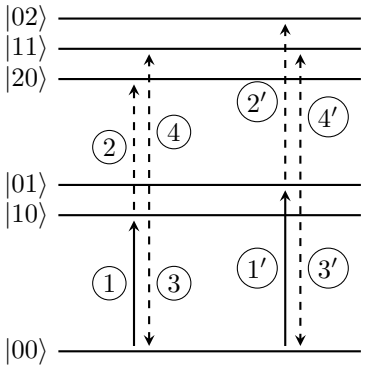
The particular two-dimensional spectroscopy we consider in this thesis is an extension to 'normal' transient spectroscopy, also known as transient pump-probe or transient-absorption spectroscopy. In this case, a sample is hit by two laser pulses: a 'pump' and a 'probe' pulse. First, a short laser pulse excites the sample (pump). Then, after a waiting time t, another short laser pulse hits the sample (probe). The absorption of this probe pulse by the sample is then measured.

Initially, the absorption of the probe pulse is measured *without* activating the pump pulse; this is considered to be the baseline. Then, the pump pulse is activated, and the absorption is measured again; the absorption difference compared to the baseline is the actual measurement. Typically, the probe pulse has a broad spectral bandwidth to equally test a large range of frequencies, while the pump pulse has a more well-defined frequency to probe a very specific transition in isolation. The signal of the probe pulse is then spectrally decomposed, in order to get a spectrum: absorption vs. frequency.

A number of phenomena may cause an absorption difference. In figure 2.1a the relevant transitions in a molecule are mentioned. In the baseline situation, without the pump pulse, the probe can only cause an excitation corresponding to transition ① in part of the sample. This causes some absorption of the incoming signal, and this is the baseline.



(a) Possible transitions for a single molecule



(b) Possible transitions for a coupled system

Figure 2.1: The relevant transitions for pump-probe spectroscopy in both a single molecule and a coupled system. The solid lines indicate the transition typically excited by the pump pulse, while the dashed lines indicate transitions that are available for the probe pulse.

With the pump pulse present, the situation is different. The pump pulse will excite part of the sample to the excited state $|1\rangle$ already before the probe pulse. Consequently, this opens up possibilities for the probe pulse, and now three things can happen:

1. The probe pulse further excites a molecule, corresponding to transition ②. This has a positive effect on the relative absorption (i.e. it becomes larger), and is referred to as *excited state absorption (EA)*.
2. The probe pulse causes stimulated emission in the excited state, corresponding to transition ③. This has a negative effect on the relative absorption, as another photon is emitted. This is referred to as *stimulated emission (SE)*.
3. The probe pulse does not interact with the sample. This is a more indirect effect, but compared to the baseline situation, fewer molecules in the sample are left in the ground state; therefore, fewer molecules can be excited along transition ①. This has a net negative effect on the relative absorption as well, and is referred to *ground state bleach (GSB/GB)*.

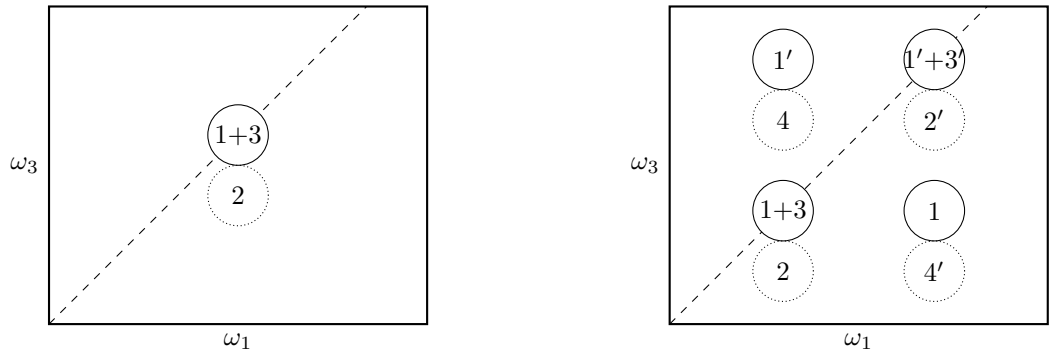
In the final spectrum, these phenomena cannot always be distinguished. For example, GB and SE both have a negative effect at the same frequency, and it is not easy to determine the contribution of each option individually. However, EA typically occurs at a different frequency, where it will have a positive effect. Furthermore, the coupling between molecules cannot easily be detected; we will treat this in section 2.1.3.

Lastly, by changing the waiting time t, it is possible to investigate the dynamics in the system. For example, it might be that the system slowly relaxes from a partially excited state (due to the pump pulse) back to the ground state, which diminishes all effects.

2.1.2 Extra dimension

When switching to two-dimensional spectroscopy, we add an extra dimension of measurement by allowing the pulse frequency to change. In theory, one could just repeat normal transient spectroscopy for various frequencies, but there is a trade-off in signal quality: spectrally narrow pulses cannot be short, while short pulses are necessarily spectrally broad. This makes it difficult to deconvolute the result.

Instead, in our case, we will use two ultrashort, spectrally broad pump pulses. In the mathematics related to spectroscopy, these can be seen as delta pulses, as they have a similar effect as ideal



(a) 2D spectrum for a single molecule

(b) 2D spectrum for a system with coupling

Figure 2.2: Qualitative overview of 2D spectroscopy spectra, both for a system with and without coupling. The various peaks correspond to different transitions in figure 2.1. Solid circles correspond to negative peaks; while dotted circles correspond to positive peaks. The dashed line represents the diagonal for which $\omega_1 = \omega_3$.

delta pulses would have. It is then possible to vary the waiting time between the two pulses to achieve a net effect of a single, spectrally very narrow pulse. This can roughly be seen as some sort of Fourier transform.

The transitions as outlined in figure 2.1a and explained before remain the same, but now we have another variable. In practice, then, we have three laser pulses: two short pump ones with a delay t_1 in between, then a delay t_2, and finally the probe pulse after which we measure the signal over some time t_3. We compute the Fourier transform of t_1 and t_3, which allows us to make a 2D spectrum as shown in figure 2.2a.

This spectrum has two dimensions: ω_1, which corresponds to the variation in t_1, and ω_3, which corresponds to the signal over t_3. We see two peaks, indicated by the circles. The solid circle represents a negative peak, while the dotted circle represents a positive one. The negative peak corresponds to the contributions of GB and SE, therefore it is indicated by '1+3', in correspondence with the transitions in figure 2.1 (the 3 refers to the direct contribution of the stimulated emission, while the 1 refers to ground state bleach, the *lack* of transition ① happening).

Likewise, the positive peak corresponds to the contribution of ESA, and is shifted down slightly. This shift is called the *anharmonicity* and corresponds to the energy difference between the transitions $|0\rangle \rightarrow |1\rangle$, and $|1\rangle \rightarrow |2\rangle$. This peak occurs at the same value for ω_1, as an excitation to $|1\rangle$ by the pump pulse is required for this new transition to be available.

2.1.3 Coupling

All theory up until now discussed systems with independent molecules. However, in reality, molecules are not independent and the systems typically have some coupling. Concretely, this means that an excited state in one molecule can 'affect' other molecules as well, and that the exact frequencies of the transitions shift.

We will consider a system with two coupled molecules, as indicated in figure 2.1b. We indicate the states by $|ij\rangle$, where i is the excitation number of the first molecule, and j is that of the second molecule. When we focus only on the first molecule, while keeping the second one in the ground state, we have the same transitions as for the uncoupled case, annotated with the same numbers. When we just focus on the second molecule, we again have similar transitions, this time annotated with a prime.

New are the transitions where we excite both molecules, transitions 4 and $4'$. Due to the coupling, the transition $|00\rangle \rightarrow |01\rangle$ does not have the same energy difference as $|10\rangle \rightarrow |11\rangle$, so we have to

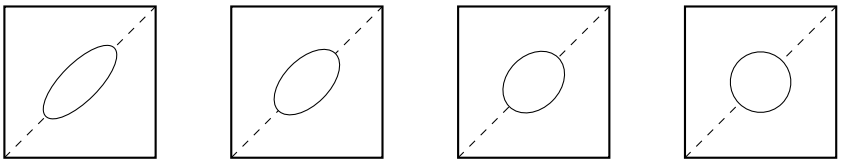


Figure 2.3: Different shapes of peaks in a 2D spectrum, giving us information about the homogeneity and dynamics of the system. From left to right: more homogenous systems at the same waiting time t_2, or the same system at higher waiting times t_2.

treat these separately. This difference in energies is again referred to as the anharmonicity.

In the spectrum, we now find more peaks than before, as can be seen in figure 2.2b. We assume that the excitation of one of the molecules has a different energy than the other, yielding two diagonal peak pairs; for GB+SE and ESA again. Additionally, we see new off-diagonal *cross-peaks*, a direct consequence of the coupling.

The negative peak of transition ① is the exact opposite of the positive peak of transition ④. Consider the peak pair $1'$ and 4. The pump pulse has excited the first molecule, while the second molecule is in the ground state. Now, the excitation of the second molecule has a different energy, due to the anharmonicity. This means that at the original ω_3 we see a negative peak: less absorption, since no transition happens at that energy. However, at a slightly shifted energy we now see a positive peak since absorption is happening due to ④ where that was not the case previously. If anharmonicity were to be zero, the peaks would overlap and cancel exactly.

2.1.4 Spectral peak features

Not only the peak location, but also the peak shape encodes information about the system; it tells us something about the homogeneity, or disorder in the system and its states. Peaks are not just circular, but they can be elongated along the diagonal, as illustrated in figure 2.3. As long as the pump pulse is sufficiently spectrally narrow (consisting of only a narrow range of frequencies), the elongation provides information about the disorder in the system.

Consider a sample with a relatively high disorder: all molecules have slightly different excitation energies, and therefore slightly different frequencies in the spectrum. This will create a bandwidth of excitation frequencies. A pump pulse with a bandwidth much narrower than the disorder will excite only a small part of the sample, only those molecules that are in that part of the bandwidth. For the waiting time $t_2 = 0$, the probe pulse will find the effects of GB and SE at that same specific frequency. Repeating this, while scanning over all frequencies in the system, will result in a diagonally elongated (negative) peak. Systems with low disorder will not experience this elongation, and will appear more circular.

The peak shape can also inform us about the dynamics. If the sample has a dynamic environment, the excitation energies of all molecules will continuously change a bit. That is, the disorder is not a static offset for each molecule, but the exact excitation energies per molecule change within a bandwidth dictated by the disorder. After increasing waiting time t_2, the excitations will have blended into the environment, and previously high-energy molecules might now be found on the lower end and vice versa, yielding a more circular peak.

2.1.5 Relative polarisation orientations

Another aspect of the spectra can tell us something about the orientations of the transition dipoles in the system. A transition dipole is the dipole moment associated with a transition such as the ones described before. We will not treat the underlying physics here in great detail, but it suffices to know that a transition dipole puts a constraint on the polarisation of incoming light in order for the state to become excited. Light that is polarised perpendicular to the transition dipole is

unlikely to excite the system, while light that is polarised parallel to the transition dipole is likely to do so.

This knowledge allows us to extract more detail out of the spectra, by using pulses with polarised light in the experiment, and varying the directions in between the pulses [16]. Usually, two configurations are used. The 'parallel' configuration where all pulses are polarised parallel to each other, and the 'perpendicular' configuration where the first two pulses are parallel relative to each other, but the third pulse is perpendicular to them. The emitted signal associated with this configuration is also perpendicular to the first two pulses.

With these two different configurations we can get more information about the structure of the system. For example, a system could show a stronger response for the parallel configuration than for the perpendicular one, telling us that the transition dipole for the transition to the second excited state (④) is parallel to the one from ground to first excited state (①).

2.2 Formalism

With an intuitive idea of the physics behind the spectroscopy, it is time to direct our attention towards the mathematical aspects. We will focus on the parts that are necessary for our understanding, and the parts that are necessary when simulating the system later on, following the style of several lecture notes [17, 18]. A more in-depth coverage of the formalism is also available [19]. We will use the convention that t_x indicates time spans, whereas τ_x indicates an absolute point in time: $t_n = \tau_{n+1} - \tau_n$.

2.2.1 Schrödinger Equation

We will discuss the spectroscopy from a quantum mechanical perspective. Consequently, we will be dealing with the Schrödinger equation and its solutions. In this thesis, we will not focus on how to find solutions to it, but rather on how to use these solutions in the context of spectroscopy.

We will start with the basics, the most general case: the time-dependent Schrödinger equation, which is given by

$$i\hbar \frac{\mathrm{d}\,|\Psi(\tau)\rangle}{\mathrm{d}\tau} = \hat{H}(\tau)\,|\Psi(\tau)\rangle\,. \tag{2.1}$$

Here, we have the wave function $|\Psi(\tau)\rangle$ that defines the state of the system, and the time-dependent Hamiltonian $\hat{H}(\tau)$ that defines the environment of the system. For now, we will restrict ourselves to a time-*in*dependent Hamiltonian $\hat{H}^0$, which we can do for reasons we will see later.

We define a time-evolution operator $\hat{U}$ that takes a wave function governed by a time-independent Hamiltonian from a starting time τ_0 to a new time τ:

$$\hat{U}(\tau, \tau_0) \equiv \exp\!\left(-\frac{i}{\hbar}\hat{H}^0(\tau - \tau_0)\right), \tag{2.2}$$

such that $|\Psi(\tau)\rangle = \hat{U}(\tau, \tau_0)\,|\Psi(\tau_0)\rangle$. It follows that $\hat{U}(\tau_1, \tau_2) = \hat{U}^\dagger(\tau_2, \tau_1)$ and $\hat{U}(\tau_1, \tau_0)\hat{U}(\tau_0, \tau_2) = \hat{U}(\tau_1, \tau_2)$.

So far, this might not seem very useful, as we just tried to get rid of the time dependence. However, time-independent Hamiltonians might still allow time-dependent wave functions: think of the harmonic oscillator or a free particle. Unfortunately, now we are still left with time dependence in the states, while the Hamiltonian (and the environment) is static. We can simplify this by making a shift of bases.

2.2.2 Interaction picture

The *interaction picture* is a new basis, completely equivalent to the standard *Schrödinger picture*, that eases the work with time dependence. In this picture, the time dependence is 'hidden' in the

operators, instead of in the wave functions. The transformation is defined as follows:

$$\begin{aligned} |\Psi_I(\tau)\rangle &= \exp\left(\frac{i}{\hbar}\hat{H}^0(\tau-\tau_0)\right)|\Psi(\tau)\rangle = \hat{U}^\dagger(\tau,\tau_0)\,|\Psi(\tau)\rangle \\ \hat{\Omega}_I(\tau) &= \exp\left(\frac{i}{\hbar}\hat{H}^0(\tau-\tau_0)\right)\hat{\Omega}(\tau)\exp\left(-\frac{i}{\hbar}\hat{H}^0(\tau-\tau_0)\right) = \hat{U}^\dagger(\tau,\tau_0)\hat{\Omega}(\tau)\hat{U}(\tau,\tau_0), \end{aligned} \tag{2.3}$$

for any operator $\hat{\Omega}(\tau)$. Note that $|\Psi_I(\tau)\rangle = \hat{U}^\dagger(\tau,\tau_0)\,|\Psi(\tau)\rangle = |\Psi(\tau_0)\rangle$ so that as long as the Hamiltonian is time-independent, there is no time dependence in the wave functions in this picture.

2.2.3 Time-dependent perturbation theory

Above, we made the simplification to only consider time-independent Hamiltonians. However, in spectroscopy, we are dealing with a dynamic system, full of interactions with laser beams, and therefore we have a time-dependent Hamiltonian. This is a much more complicated situation, with which we will deal by using time-dependent perturbation theory.

Time-dependent perturbation theory, in its core, means that we reduce a complicated Hamiltonian to a simpler, time-independent form, and consider the time-dependent part as a small perturbation to that system. In the limit of the perturbation going to zero, the system will reduce to the simple case. This means that we will split the base Hamiltonian in two parts, the time-independent part $\hat{H}^0$ introduced before, governing the system in isolation, and the time-dependent part $\hat{H}^P(\tau)$, governing the interaction with the laser pulses, such that $\hat{H}(\tau) = \hat{H}^0 + \hat{H}^P(\tau)$.

The Schrödinger equation can be restated with the new Hamiltonians, and in the interaction picture only the time-dependent part is of concern, so eq. 2.1 can be rewritten to:

$$i\hbar\frac{\mathrm{d}\,|\Psi_I(\tau)\rangle}{\mathrm{d}\tau} = \hat{H}_I^P(\tau)\,|\Psi_I(\tau)\rangle\,. \tag{2.4}$$

For the purposes of spectroscopy, the time-dependent perturbation is the interaction between the system and the external electric field, given by:

$$\hat{H}_I^P(\tau) = \hat{\vec{\mu}}_I(\tau)\cdot\vec{E}(\tau) + \cdots\,, \tag{2.5}$$

with $\hat{\vec{\mu}}$ the transition dipole operator (further denoted without the vector symbol) and $\vec{E}$ the external electric field. The dots at the end indicate higher-order contributions, but for absorption spectroscopy these are irrelevant and we will only deal with the transition dipoles.

To solve the Schrödinger equation, we will first integrate eq. 2.4 (using τ' for integration variables):

$$|\Psi_I(\tau)\rangle = |\Psi_I(\tau_0)\rangle - \frac{i}{\hbar}\int_{\tau_0}^{\tau}\mathrm{d}\tau_1'\,\hat{H}_I^P(\tau_1')\,|\Psi_I(\tau_0)\rangle\,,$$

and then plug it into the Schrödinger equation repeatedly, to gain higher-order corrections. These take the form

$$\begin{aligned} \left|\Psi_I^{(n)}(\tau)\right\rangle &= \left(-\frac{i}{\hbar}\right)^n\int_{\tau_0}^{\tau}\mathrm{d}\tau_n'\int_{\tau_0}^{\tau_n'}\mathrm{d}\tau_{n-1}'\cdots\int_{\tau_0}^{\tau_2'}\mathrm{d}\tau_1'\cdot\hat{H}_I^P(\tau_n')\cdots\hat{H}_I^P(\tau_1')\,|\Psi_I(\tau_0)\rangle \\ \left|\Psi^{(n)}(\tau)\right\rangle &= \left(-\frac{i}{\hbar}\right)^n\int_{\tau_0}^{\tau}\mathrm{d}\tau_n'\int_{\tau_0}^{\tau_n'}\mathrm{d}\tau_{n-1}'\cdots\int_{\tau_0}^{\tau_2'}\mathrm{d}\tau_1' \\ &\qquad\cdot\hat{U}(\tau,\tau_n')\hat{H}^P(\tau_n')\cdots\hat{U}(\tau_2',\tau_1')\hat{H}^P(\tau_1')\hat{U}(\tau_1',\tau_0)\,|\Psi(\tau_0)\rangle \end{aligned} \tag{2.6}$$

$$\begin{aligned} |\Psi_I(\tau)\rangle &= |\Psi_I(\tau_0)\rangle + \sum_{k=1}^{\infty}\left|\Psi_I^{(k)}(\tau)\right\rangle \\ |\Psi(\tau)\rangle &= \hat{U}(\tau,\tau_0)\,|\Psi(\tau_0)\rangle + \sum_{k=1}^{\infty}\left|\Psi^{(k)}(\tau)\right\rangle \end{aligned} \tag{2.7}$$

for an n^{th}-order correction and the full solution in both the interaction and Schrödinger pictures. Note that the time dependence is 'hidden' in $\hat{H}_I^P$ in the interaction picture, which is made explicit in the Schrödinger picture. This means that we can interpret the n^{th}-order correction as the correction for the n^{th} interaction with the external electric field: we have the base state $|\Psi_I(\tau_0)\rangle$, and apply time-evolution operators in between each interaction with operator $\hat{H}^P$. This interpretation will be similar when computing the system response.

2.2.4 System response

Now that we know how the system evolves in time, and how the interactions with an electric field affect its state, it is time to see what we can measure. For spectroscopy, we are interested in the polarisation of the system, as an oscillating polarisation yields a measurable electric field, we will see that the polarisation can be expanded in a series of $\vec{E}$. The macroscopic quantity of polarisation is the average of all microscopic polarisations, or in this case, all transition dipoles:

$$P(\tau) = \langle \Psi(\tau)|\hat{\mu}|\Psi(\tau)\rangle_E = \langle \Psi_I(\tau)|\hat{\mu}_I|\Psi_I(\tau)\rangle_E \,,$$

with $\langle \ldots \rangle_E$ denoting the *ensemble average* of the expectation value.

We can take the full expanded solution of eq. 2.7, put it in the equation above, rearrange the terms, and separate the polarisations in different orders as well, such that the n^{th}-order polarisation corresponds to n interactions with the external electric field:

$$P^{(n)}(\tau) = \sum_{k=0}^{n} \left\langle \Psi_I^{(n-k)}(\tau) \middle| \hat{\mu}_I \middle| \Psi_I^{(k)}(\tau) \right\rangle_E . \tag{2.8}$$

We will now derive a more usable version of this equation; a version that we can actually use to compare theory and experiment, and a version where we can better visualise what is happening using Feynman diagrams, to be introduced later. First, we expand eq. 2.8 using the definition in eq 2.6, and shift the integration variables in the bra by $+k$.

$$\begin{aligned} P^{(n)}(\tau) = \sum_{k=0}^{n} \left(\frac{i}{\hbar}\right)^{n-k} \int_{\tau_0}^{\tau} \mathrm{d}\tau_n' \cdots \int_{\tau_0}^{\tau_{k+2}'} \mathrm{d}\tau_{k+1}' \cdot \left(-\frac{i}{\hbar}\right)^{k} \int_{\tau_0}^{\tau} \mathrm{d}\tau_k' \cdots \int_{\tau_0}^{\tau_2'} \mathrm{d}\tau_1' \\ \cdot \left\langle \Psi_I(\tau_0) \middle| \hat{H}_I^P(\tau_{k+1}') \cdots \hat{H}_I^P(\tau_n') \cdot \hat{\mu}_I(\tau) \cdot \hat{H}_I^P(\tau_k') \cdots \hat{H}_I^P(\tau_1') \middle| \Psi_I(\tau_0) \right\rangle_E \end{aligned}$$

We will shift from moments in time (τ) to time spans (t), since those are what we typically control in experiments. This means that $\tau_x = \tau - t_n - \cdots - t_x$ denoted by $\tau \ldots t_x$, and since we are subtracting we will let the upper boundary of the integrals go to infinity under the assumption that the system is stationary before τ_0. Lastly, we set $\tau_0 = 0$.

$$\begin{aligned} &= \left(\frac{i}{\hbar}\right)^{n} \int_0^{\infty} \mathrm{d}t_n \cdots \int_0^{\infty} \mathrm{d}t_1 \cdot \sum_{k=0}^{n} (-1)^k \\ &\quad \left\langle \Psi_I(0) \middle| \hat{H}_I^P(\tau \ldots t_{k+1}) \cdots \hat{H}_I^P(\tau \ldots t_n) \cdot \hat{\mu}_I(\tau) \cdot \hat{H}_I^P(\tau \ldots t_k) \cdots \hat{H}_I^P(\tau \ldots t_1) \middle| \Psi_I(0) \right\rangle_E \end{aligned}$$

We will use the first-order expansion of the perturbed Hamiltonian from eq. 2.5.

$$\begin{aligned} &= \left(\frac{i}{\hbar}\right)^{n} \int_0^{\infty} \mathrm{d}t_n \cdots \int_0^{\infty} \mathrm{d}t_1 \cdot \vec{E}(\tau \ldots t_n) \cdots \vec{E}(\tau \ldots t_1) \cdot \sum_{k=0}^{n} (-1)^k \\ &\quad \left\langle \Psi_I(0) \middle| \hat{\mu}_I(\tau \ldots t_{k+1}) \cdots \hat{\mu}_I(\tau \ldots t_n) \cdot \hat{\mu}_I(\tau) \cdot \hat{\mu}_I(\tau \ldots t_k) \cdots \hat{\mu}_I(\tau \ldots t_1) \middle| \Psi_I(0) \right\rangle_E \\ &= \int_0^{\infty} \mathrm{d}t_n \cdots \int_0^{\infty} \mathrm{d}t_1 \cdot \vec{E}(\tau \ldots t_n) \cdots \vec{E}(\tau \ldots t_1) \cdot S^{(n)}(t_n, \ldots, t_1) \end{aligned} \tag{2.9}$$

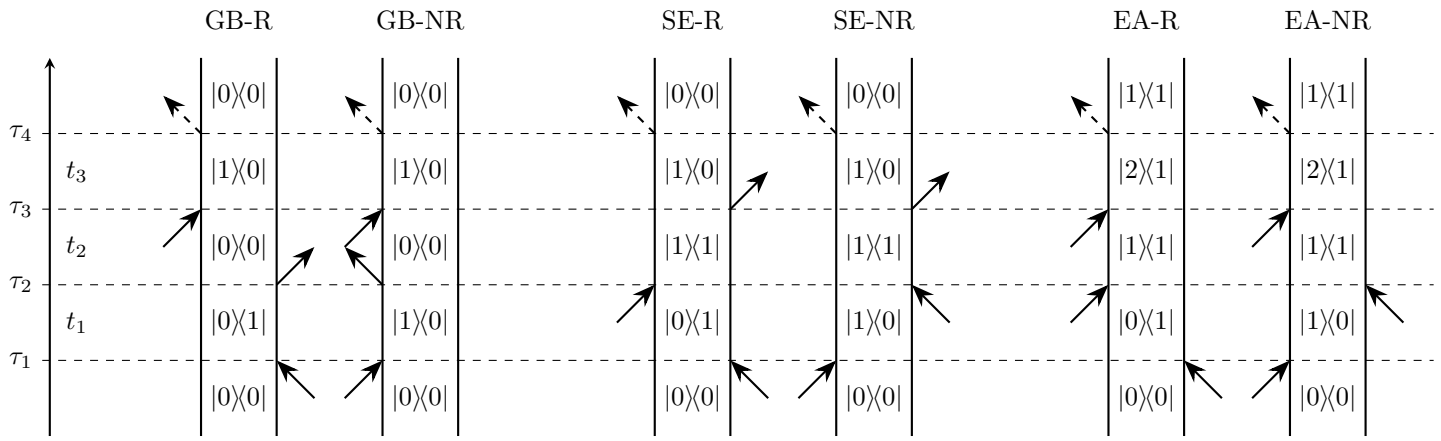


Figure 2.4: The six double-sided Feynman diagrams that are relevant for third-order two-dimensional spectroscopy.

where we define the *response function*

$$S^{(n)}(t_n, \ldots, t_1) = \left(\frac{i}{\hbar}\right)^n \sum_{k=0}^{n} (-1)^k \langle \Psi_I(0)| \hat{\mu}_I(t_k + \cdots + t_1) \cdots \hat{\mu}_I(t_n + \cdots + t_1) \cdot \hat{\mu}_I(t_{k-1} + \cdots + t_1) \cdots \hat{\mu}_I(0) |\Psi_I(0)\rangle_E . \tag{2.10}$$

This has an intuitive explanation: the polarisation contains terms for each applied electric field, independent from time ordering of the interactions or the exact dynamics. Then the response function contains the sum, that incorporates the contributions from *all possible time-orderings with n interactions*. These can be visualised in double-sided Feynman diagrams.

2.2.5 Feynman Diagrams

Let us consider the third-order polarisation, as that's the relevant case in two-dimensional spectroscopy. This will yield:

$$P^{(3)}(\tau) = \int_0^\infty \mathrm{d}t_3 \int_0^\infty \mathrm{d}t_2 \int_0^\infty \mathrm{d}t_1 \cdot \vec{E}(\tau - t_3)\vec{E}(\tau - t_3 - t_2)\vec{E}(\tau - t_3 - t_2 - t_1) \cdot S^{(3)}(t_3, t_2, t_1).$$

The many terms in the response function correspond to all different combinations of interactions between the electric field and the system at the different times. Each of them corresponds to a unique double-sided Feynman diagram, such as the ones in figure 2.4. These diagrams are a visual representation of what is actually happening in the system, and there is a one-to-one correspondence between them. They work as follows:

- The vertical lines in each diagram represent the time evolution, running forward upwards. The left line represents the *ket*, the right line represents the *bra*.
- Interactions between the electric field and the system are indicated by arrows at a specific time.
- Arrows pointing towards the system represent an excitation, while arrows pointing away from the system represent a de-excitation (emission of a photon).
- The direction of an arrow (left/right) corresponds to the phase of the electric field, $\mp\vec{k}$ respectively.
- The top arrow represents the measurement of the result, indicated by a dashed line.

- The diagram starts in the ground state $|0\rangle\langle 0|$, and must end in a balanced (*population*) state $|n\rangle\langle n|$.
- The sign of the contribution corresponds to $(-1)^m$ where m is the number of interactions on the right (bra) side.

With this information, it is possible to reproduce the response function from the diagram. Working from left to right, start with a $\langle 0|$, and work along the right vertical line upwards. For each interaction with this line, add a dipole operator $\mu(\tau)$. For each time evolution of the system between interactions, add a time-evolution operator $\hat{U}(\tau_2, \tau_1)$. At the top, we loop back around and go down the left vertical line, ending with a $|0\rangle$.

One more detail to consider is the phase of each electric field applied, $\pm\vec{k}$. In figure 2.4, we have drawn -R and -NR versions of each diagram, corresponding to *rephasing* and *non-rephasing* phase-matching directions. The technical implications of this difference are not relevant for this thesis, but they correspond to the phase combinations $\vec{k}_R = -\vec{k}_1 + \vec{k}_2 + \vec{k}_3$ and $\vec{k}_{NR} = +\vec{k}_1 - \vec{k}_2 + \vec{k}_3$: basically the -R and -NR versions have the first two interactions switched around. They are usually called $\vec{k}_R = \vec{k}_I$ and $\vec{k}_{NR} = \vec{k}_{II}$.

There is one more phase-matching direction, $\vec{k}_{III} = +\vec{k}_1 + \vec{k}_2 - \vec{k}_3$, that is not considered in this thesis. Experimental setups typically also do not measure these, so therefore it is also not depicted in the diagram.

2.2.6 Connecting to experiment

What is left now, is to combine our qualitative physical intuition and the formalism described before. In particular, we have to make the connection between the contributions of all Feynman diagrams in figure 2.4 and the final spectrum in figure 2.2.

The Feynman diagrams are already annotated with the contributions they represent. The first two diagrams correspond to ground state bleach, or in fact, the *lack* of contribution from these diagrams. These two diagrams end up in the ground state during the waiting time t_2, and it is the lack of molecules in the ground state during the probe pulse at τ_3 that causes the ground state bleach effect.

The middle two diagrams are for stimulated emission: between the pump and probe pulses, the system is in a singly-excited state, and afterwards, it is returned back to the ground state by the probe with the release of a photon. The last one is excited state absorption: once again, the system is in a singly-excited state during the waiting time, but now the probe pulse excites the system to a doubly-excited state.

Recall that both GB and SE caused a negative peak in the relative absorption spectrum, while EA caused a positive peak. This is also evident from the diagrams, governed by the rules above. The contribution from a Feynman diagram to the response is either negative or positive, according to $(-1)^m$ where m is the number of interactions on the bra side. For GB and SE, there is an even number of interactions on this side, so both have a positive contribution to the response of the system, and therefore a negative contribution to the absorption. The opposite holds for EA, which has an odd number of interactions on the bra side.

Note that these diagrams also hold for the situation of coupling as described in section 2.1.3. In that case, $|1\rangle$ corresponds to any state with a single excitation, and $|2\rangle$ corresponds to any state with two excitations, whether that be $|20\rangle$, $|02\rangle$, or $|11\rangle$.

Chapter 3

Simulation

With the theoretical background in place, we can turn our attention towards the simulation of two-dimensional spectroscopy. Simulation of the system should return results that can then be compared to experiment. Concretely, we want to obtain the response functions $S(t_3, t_2, t_1)$ as defined in eq. 2.10. In this thesis, we will use the *Numerical Integration of the Schrödinger Equation* (NISE) [10, 15] scheme to do so, as it is a good fit for simulating the C8S3 aggregate we investigate in this thesis [20].

3.1 NISE scheme

In the NISE scheme, the sample is roughly divided into two parts: the 'system', the part of the sample that interacts with the laser pulses, and the 'bath', that remains stationary [15]. This bath has a similar role as the ones in classical statistical physics, and it is assumed that the bath can affect the system, but not vice versa.

3.1.1 Time-dependence of the Hamiltonian

In the previous chapter, all theory was dependent on a Hamiltonian describing the system, but no concrete equations were given for this. Furthermore, this Hamiltonian was assumed to be time-independent between the laser pulses, in order to utilise time-dependent perturbation theory. Unfortunately, in practice, this does not hold. The system is in contact with the bath, causing the Hamiltonian and the transition dipoles to be time-dependent.

Fortunately, the majority of the equations keep up, as they were designed with the time-dependent perturbation $\hat{H}^P(\tau)$ in mind. Only the time-evolution operator $\hat{U}$ (eq. 2.2) depends on a time-independent Hamiltonian. This can be accounted for by generalising this operator:

$$\hat{U}(\tau, \tau_0) \equiv \exp\left(-\frac{i}{\hbar}\int_{\tau_0}^{\tau} \hat{H}(\tau')\,\mathrm{d}\tau'\right). \tag{3.1}$$

3.1.2 Hamiltonian of a three-level system

Now that all theory is ready for time-dependent Hamiltonians, we can get more concrete with an actual Hamiltonian of a system. As we are considering two-dimensional spectroscopy, we will consider a three-level system. Additionally, we will incorporate the effects of interaction with the bath, consider coupling between molecules, and include the first order of the interaction between

the system and the electric field, as given by eq. 2.5. For N molecules, this yields [21]:

$$\hat{H}(\tau) = \sum_{k=1}^{N} \left(\epsilon_k(\tau) \hat{b}_k^\dagger \hat{b}_k - \frac{\Delta_k(\tau)}{2} \hat{b}_k^\dagger \hat{b}_k^\dagger \hat{b}_k \hat{b}_k \right) + \sum_{k,l}^{N} J_{kl}(\tau) \hat{b}_k^\dagger \hat{b}_l + \sum_{k=1}^{N} \hat{\mu}_k(\tau) \cdot \vec{E}(\tau) \left(\hat{b}_k^\dagger + \hat{b}_k \right), \quad (3.2)$$

where $\epsilon_k(\tau)$, $\Delta_k(\tau)$, $J_{kl}(\tau)$, and $\hat{\mu}_k(\tau)$ are the time-dependent frequency, anharmonicity, coupling, and transition dipoles, respectively, and where $\hat{b}_k^\dagger$ and $\hat{b}_k$ are the Bose creation and annihilation operators.

The unknowns $\epsilon_k(\tau)$, $\Delta_k(\tau)$, $J_{kl}(\tau)$, and $\hat{\mu}_k(\tau)$ should be found in advance of running the NISE scheme using more traditional molecular dynamics simulations combined with precomputed mappings [22], stochastic models [23], or a combination of both [24]. The disorder, as mentioned in section 2.1.4, is incorporated by giving each molecule a random energy gap and re-evaluating that gap with a given interval [23].

Note that for visible spectroscopy, only two-level systems are considered. It is still possible to do two-dimensional spectroscopy, as it is still possible for two excitations to 'live' in the system: we only exclude a doubly-excited state from our consideration. In practice, this is done by either excluding such a state from the basis, or by setting the anharmonicity (Δ_k) to 'infinity'.

3.1.3 Simplification of the Hamiltonian

The Hamiltonian as given above might fully describe the system, but is not easy to work with in a simulation. We will now employ some tricks to simplify this, as the features of two-dimensional spectroscopy limit the problem size and possible interactions in the Hamiltonian.

The first thing to note is that only the term governing the transition dipoles is able to mix states with a different number of excitations. Therefore, we can make a simplification here: in between the laser pulses, the Hamiltonian is block diagonal and we can separate the blocks for the ground state, singly-excited states, and doubly-excited states $\hat{H}^{00}$, $\hat{H}^{11}$, and $\hat{H}^{22}$. Later on, we will continue using $\hat{H}(\tau)$, where one can add one of the suffixes for each block.

Additionally, we only ever considered transitions spanning a single energy level difference, see fig. 2.1. Consequently, we are only concerned with the specific transition dipoles between those states, $\hat{\mu}^{01}$ and $\hat{\mu}^{12}$.

3.1.4 Discretisation

The Hamiltonian in its current, simplified, form is still subject to the continuous variable τ. For simulation purposes, we want to discretise this, to work in discrete time steps so the computer can efficiently make calculations with a given precision (time step size).

We start by considering a basis in which we can express our resulting wave functions $\Psi(\tau)$. We will use the so-called 'site basis', where the excitations are localised on *sites* (individual molecules in the system): $\psi_k = \hat{b}_k^\dagger |0\rangle$ at an initial state. This can be generalised as:

$$\Psi_k(\tau) = \sum_l \psi_l c_{lk}(\tau) \quad (3.3)$$

with the coefficients matrix c, for which $c_{lk}(\tau_0) = \delta_{lk}$. This general form is subject to the time-evolution operator $\hat{U}$ that will mix the basis states into a new wave function. This form can be used to solve the Schrödinger equation, eq. 2.1:

$$\frac{\partial}{\partial \tau} c_{km}(\tau) = -\frac{i}{\hbar} \sum_l \hat{H}_{kl}(\tau) \cdot c_{lm}(\tau)$$

$$\frac{\partial}{\partial t} c(\tau) = -\frac{i}{\hbar} \hat{H}(\tau) c(\tau).$$

This we can now discretise: we divide the integration in small time increments Δt and assume the Hamiltonian to be constant during each such time step. Starting at $\tau_0 = 0$, for the n^{th} time step this yields:

$$c\big((n+1)\Delta t\big) = \exp\!\left(-\frac{i}{\hbar}\hat{H}(n\Delta t)\Delta t\right) c(n\Delta t) = \hat{U}\Big((n+1)\Delta t,\ \ n\Delta t\Big) c(n\Delta t),$$

using the original time-independent time-evolution operator (eq. 2.2) with the Hamiltonian at $\tau = n\Delta t$. It can be generalised for any time difference of n time steps:

$$\hat{U}(n\Delta t, 0) = \prod_{m=1}^{m=n} \hat{U}\Big(m\Delta t, (m-1)\Delta t\Big) \tag{3.4}$$

3.2 Signal

The formalism from NISE allows us to now calculate a response function $S(t_n, \ldots, t_1)$. However, in practice, we cannot measure this directly: typically we measure the spectrum of the response. The last step is then to convert the response function into such a spectrum. We do this by first adding a so-called relaxation factor $\Gamma(t_n, \ldots, t_1)$ that captures the lifetime of a state, and then we perform a Fourier transform of the result from the time domain to the frequency domain.

3.2.1 Linear absorption

The simplest case for conversion from the response function to a spectrum is linear absorption, which is the one-dimensional variant of pump-probe spectroscopy. We assume we start with $\Psi(0) = |0\rangle$. From eq. 2.10, we obtain the response function (using $n = 1$):

$$S^{(1)}(t_1) = \frac{i}{\hbar}\left\langle 0\middle|\hat{\mu}^{01}(t_1)\hat{U}^{11}(t_1, 0)\hat{\mu}^{10}(0)\middle|0\right\rangle_E,$$

where $\hat{U}^{11}$ is the time-evolution operator using the $\hat{H}^{11}$ part of the Hamiltonian.

For the time evolution, we add the relaxation factor

$$\Gamma(t_1) = \exp\!\left(-\frac{t_1}{2T_1}\right),$$

with T_1 the lifetime of the singly-excited states. Lastly, we add the Fourier transform of the system. The spectrum is then the imaginary part of the result, leaving us with:

$$I(\omega) = \mathrm{Im}\left\{\int_0^\infty \Big(S^{(1)}(t_1)\Gamma(t_1)\Big)\exp(-i\omega t_1)\,\mathrm{d}t_1\right\} \tag{3.5}$$

3.2.2 Two-dimensional results

This next case is the one applicable to two-dimensional spectroscopy. The basics are the same as for the linear case, but now we have two dimensions and three time delays. The two dimensions correspond to a double Fourier transform along both t_1 and t_3, while t_2 stays a normal variable: the waiting time.

Again, we start with writing out the response functions. We limit the number of equations by only considering the contributions from the six Feynman diagrams in fig. 2.4. To simplify the equations, we will reintroduce $\tau_i = t_{i-1} + \cdots + t_1$ with $\tau_1 = 0$, and we omit the terms $\hat{U}^{00}(\tau_i, \tau_j)$ that would have been present as those reduce to the identity matrix (only the ground state with zero energy is relevant).

$$S^I_{GB}(t_3,t_2,t_1) = -\left(\frac{i}{\hbar}\right)^3 \left\langle 0\middle|\hat{\mu}^{01}(\tau_1)\hat{U}^{11}(\tau_1,\tau_2)\hat{\mu}^{10}(\tau_2)\hat{\mu}^{01}(\tau_4)\hat{U}^{11}(\tau_4,\tau_3)\hat{\mu}^{10}(\tau_3)\middle|0\right\rangle_E$$
$$S^I_{SE}(t_3,t_2,t_1) = -\left(\frac{i}{\hbar}\right)^3 \left\langle 0\middle|\hat{\mu}^{01}(\tau_1)\hat{U}^{11}(\tau_1,\tau_3)\hat{\mu}^{10}(\tau_3)\hat{\mu}^{01}(\tau_4)\hat{U}^{11}(\tau_4,\tau_2)\hat{\mu}^{10}(\tau_2)\middle|0\right\rangle_E$$
$$S^I_{EA}(t_3,t_2,t_1) = +\left(\frac{i}{\hbar}\right)^3 \left\langle 0\middle|\hat{\mu}^{01}(\tau_1)\hat{U}^{11}(\tau_1,\tau_4)\hat{\mu}^{12}(\tau_4)\hat{U}^{22}(\tau_4,\tau_3)\hat{\mu}^{21}(\tau_3)\hat{U}^{11}(\tau_3,\tau_2)\hat{\mu}^{10}(\tau_2)\middle|0\right\rangle_E$$

$$S^{II}_{GB}(t_3,t_2,t_1) = -\left(\frac{i}{\hbar}\right)^3 \left\langle 0\middle|\hat{\mu}^{01}(\tau_4)\hat{U}^{11}(\tau_4,\tau_3)\hat{\mu}^{10}(\tau_3)\hat{\mu}^{01}(\tau_2)\hat{U}^{11}(\tau_2,\tau_1)\hat{\mu}^{10}(\tau_1)\middle|0\right\rangle_E$$
$$S^{II}_{SE}(t_3,t_2,t_1) = -\left(\frac{i}{\hbar}\right)^3 \left\langle 0\middle|\hat{\mu}^{01}(\tau_2)\hat{U}^{11}(\tau_2,\tau_3)\hat{\mu}^{10}(\tau_2)\hat{\mu}^{01}(\tau_4)\hat{U}^{11}(\tau_4,\tau_1)\hat{\mu}^{10}(\tau_1)\middle|0\right\rangle_E$$
$$S^{II}_{EA}(t_3,t_2,t_1) = +\left(\frac{i}{\hbar}\right)^3 \left\langle 0\middle|\hat{\mu}^{01}(\tau_2)\hat{U}^{11}(\tau_2,\tau_4)\hat{\mu}^{12}(\tau_4)\hat{U}^{22}(\tau_4,\tau_3)\hat{\mu}^{21}(\tau_3)\hat{U}^{11}(\tau_3,\tau_1)\hat{\mu}^{10}(\tau_1)\middle|0\right\rangle_E$$

Again, we add a relaxation factor to account for the lifetime T_1,

$$\Gamma(t_3,t_2,t_1) = \exp\left(-\frac{t_3+2t_2+t_1}{2T_1}\right).$$

As we can only measure a single phase matching direction ($\vec{k}_I$ or $\vec{k}_{II}$) at the same time, we will combine the relevant response functions and perform the Fourier transforms on them separately. Lastly, we combine these and take the imaginary part to obtain the result:

$$\begin{aligned} I(\omega_3,t_2,\omega_1) = \operatorname{Im}\Bigg\{ &\int_0^\infty\int_0^\infty \left(S^I_{GB}+S^I_{SE}+S^I_{EA}\right)\Gamma(t_3,t_2,t_1)\exp(+i\omega_1 t_1 - i\omega_3 t_3)\,\mathrm{d}t_3\,\mathrm{d}t_1 \\ &+\int_0^\infty\int_0^\infty \left(S^{II}_{GB}+S^{II}_{SE}+S^{II}_{EA}\right)\Gamma(t_3,t_2,t_1)\exp(-i\omega_1 t_1 - i\omega_3 t_3)\,\mathrm{d}t_3\,\mathrm{d}t_1 \Bigg\}. \end{aligned} \tag{3.6}$$

All this together is sufficient to perform the simulation, but the computation will not have optimal performance. Chapter 4 covers the parallelisation of the algorithm to improve that.

Chapter 4

Parallelisation

We can now direct our attention to the performance of the simulation of two-dimensional spectroscopy following the NISE scheme. Unfortunately, a naive implementation would be very slow, as the problem size is very large. The Hamiltonian is of size $N \times N$ with N the size of the system. Furthermore, we need to take the polarisation and relative orientation of the laser pulses into account, causing us to average over 21 different polarisation directions [25, 26]. On top of that, because of the interactions with the bath and the time-dependence of the frequencies, anharmonicities, couplings, and transition dipoles, every calculation yields a slightly different result. To compensate for these fluctuations, we need to average over a number of runs.

For linear absorption, this all is manageable, as we only consider singly-excited states, and we only have one time evolution to perform with only one variable. For two-dimensional spectroscopy, this is more complicated: there are $N^2/2$ doubly-excited states from the EA contributions to consider (on top of the N singly-excited states), both t_1 and t_3 have to be varied, and there are more interactions between the system and the electric field.

The most efficient calculations achieved have a complexity of $\mathcal{O}(N^3)$ [27], which means that as the system size N doubles, the computation time increases eightfold. This makes it enormously expensive to simulate systems larger than tens of molecules.

The main work done for this thesis lies in optimising the simulation code used [11], and in particular, parallelising it. This is done to allow the simulation of larger systems, as this is necessary to obtain more accurate results and further extend the theoretical framework of the systems under investigation.

4.1 Problem decomposition

All parallelisation depends on how well the problem can be decomposed into smaller, independent chunks of work. For example, opening a second cash register generally allows double the number of customers served, as serving a customer is a small, independent chunk of work. On the other hand, a single customer is not served in half the time by opening a second cash register. This illustrates the basic problem at hand: we need to decompose the total computational work in small, independent chunks.

In the case of the NISE scheme, we can identify a number of places where work can be decomposed into independent chunks:

- As indicated before, to compensate for the fluctuations in the system, we run the calculation multiple times and average over these runs. Each of these can be made parallel, as they are independent.

- Each of the 21 different polarisation directions [27] constitutes an independent calculation.
- Every different value for t_1 and t_3 yields an independent result, all of which have to be combined.

4.2 Parallelisation techniques

Traditionally, there are two main 'levels' of parallelisation. The easiest to achieve is to simply use multiple cores or CPUs in the same physical machine, all with access to the same shared resources. This can be compared to having multiple cash registers in the same shop, all handling a customer at the same time. This offers some degree of extra performance, but it does not scale perfectly: one cannot put a thousand cash registers in a single shop, as then other resources would form the bottleneck, from parking space to size of hallways.

Another approach to parallelisation is by using multiple machines, which is more difficult to achieve. Typically, multiple machines are connected via a relatively slow network connection, so there are no shared resources, limiting the tasks suitable for this approach. This can be compared to opening up multiple shops in the city, each handling their own smaller number of customers concurrently. It would scale very well, but it does require building multiple buildings, supplying all shops with the same products, etc. If a certain product is out of stock in one shop, it cannot be easily retrieved at another shop, as that takes time.

These analogues translate to the computer realm quite well. When parallelising in a single machine, scaling is limited to the number of cores one can fit in a single machine. In more complicated machines, such as ones with multiple CPUs, scaling to all available cores might already be infeasible due to the underlying design [28, 29].

Lastly, we could consider offloading work onto a GPU, where some tasks can be parallelised within one system very efficiently [30]. Unfortunately, the characteristics of the NISE scheme and the mathematics involved do not allow for this.

4.3 Parallelisation approach

In practice, each parallelisation problem has different considerations, and there is no 'silver bullet' or 'best' parallelisation approach. We will analyse the details of each of the decompositions mentioned above and see which approach is best for each decomposition. Various aspects are relevant in this analysis:

- Computation time of a single chunk of work
- The resources necessary for each chunk of work
- Whether the resources for a chunk can be shared with other chunks, or whether they are unique for a specific chunk
- Typical number of chunks

Different runs

To reiterate: to account for the fluctuations in the system, we want to average the spectrum over multiple runs, so we need to perform the same computation multiple times. Each run is completely independent of the others: only the Hamiltonian and system parameters are shared between runs, but they are fixed and do not change. Furthermore, each run takes a long time as it constitutes the full computation of the spectrum.

It seems best to spread the different runs over multiple machines, as only a small amount of communication is required (the system parameters and Hamiltonian need to be shared) for a large

computational task.

Polarisation directions

Within each run, we need to average the result over 21 different polarisation directions. Again, each of these polarisation directions constitutes an independent calculation, with just the system parameters and the Hamiltonian shared between them. Moreover, each polarisation still takes a long time to calculate. We will treat this on equal footing as the different runs before: we spread the chunks over multiple machines, for the same reasons.

Different values of t_1 and t_3

Lastly, within each run, different values for the waiting times are combined to actually calculate the result for multiple frequencies in both dimensions. These tasks are not completely independent, as in the code they share many variables describing the system that are computationally expensive to obtain. On top of that, the result for each different value is used immediately for a further calculation. Therefore, it would be useful to have all these variables present in the same memory, to save on communication overhead. Lastly, the computation for each value of t_1 is rather short.

All these considerations together seem to indicate that parallelisation within a single machine is the best approach for this decomposition, as simultaneous access to the same memory is required and the chunks of work are small.

4.4 Hybrid parallelisation

Summarising the above, it seems best to do a so-called *hybrid* parallelisation, using multiple machines that all use all their cores. Other research shows that this can indeed be a beneficial approach for suitable problems [30–33], and the NISE scheme seems to be such a problem.

OpenMP

To get more concrete: for the intra-machine parallelisation, we will use the toolbox *OpenMP* [34]. This API facilitates shared-memory parallelisation and is easy to use: a simple annotation at a loop will parallelise the execution of it, on the condition that the loop iterations are independent. This means that the result of one iteration should not influence the result of another.

The shared memory aspect means that all parallel chunks of work can make use of the same memory, i.e. the same variables, open files, and other constructs. This makes it really easy to use, and makes it suitable for small-scale parallelisation of short tasks. An example usage of OpenMP can be found in code sample 4.1, where it is implemented with a single line of code to optimise calculating the squares from 0 to 100,000.

As described before, the shared memory aspect of OpenMP also poses a limitation: code running on different machines, or even code running on different CPUs within a single machine will experience major delays as the memory access will become very slow. This is where the advantage of OpenMP breaks down, and where we will have to look for a more sophisticated method.

Code sample 4.1: Basic use of *OpenMP*, calculating squares

```
1 int array[100000];
2
3 #pragma omp parallel for
4 for (int i = 0; i < 100000; i++) {
5     array[i] = i * i;
6 }
```

Code sample 4.2: Basic use of *MPI*, calculating squares

```
int current_rank, count_processes;
MPI_Init(); // Initialise MPI
MPI_Comm_rank(MPI_COMM_WORLD, &current_rank); // Obtain 'ID' of this process
MPI_Comm_size(MPI_COMM_WORLD, &count_processes); // Obtain number of processes

int partial_array[100000 / count_processes]; // Array to store the partial result of this process
for (int i = 0; i < 100000; i++) {
    partial_array[i] = i * i;
}

int array[100000]; // Array to store the full result
MPI_Gather(partial_array, 100000 / count_processes, MPI_INT, array, 100000, MPI_INT, 0,
↪  MPI_COMM_WORLD);
// Gathers all partial results back at the main process (ID = 0).
```

MPI

This more sophisticated method is found in the *MPI* (Message Passing Interface) standard [35]. This standard facilitates the communication between processes, typically to exchange information on what to compute and what tasks to perform. MPI gives the programmer full control over the implementation, making it very flexible, but also quite labour-intensive.

A typical MPI programme requires some 'main' process that will hand out tasks to the other processes, and that will receive all the results and process them. A short example can be found in code sample 4.2, where again we are calculating the squares from 0 to 100,000 and in the end send all results to the main process. It is clear that this is a more convoluted program, but since no memory is shared, it can scale almost indefinitely.

As a result, MPI is what we will use for the inter-machine parallelisation. The standard is typically well-suited for the type of long-running computation that each NISE run is, and only requires communication between processes at the start and end of calculations.

4.5 Implementation

Now that we have decided how we will approach the parallelisation of the NISE scheme, we can finally implement it. The implementation of MPI in conjunction with OpenMP is somewhat more tricky, as we need to carefully consider the work division. This will be explained below.

4.5.1 OpenMP

The implementation of the OpenMP part is relatively straightforward. We identify a main loop in the code that computes the propagation of the states over the various possible values of t_1 [36], see code sample 4.3. After analysis of the non-parallelised program with a profiler (Valgrind [37] and Visual Studio [38]), the method called in this loop turned out to take about 90% of the execution time of the program, and the loop only has relatively few iterations. This makes it a perfect candidate for parallelisation.

The code in sample 4.3 starts with a short description of the defined variables that are relevant for this loop. After that, the parallelisation command is given (`#pragma omp parallel for`), with two additional components. The `shared` keyword indicates that the variables listed are shared by all threads, and thus, iterations of the loop. In general, this might pose a risk of values being overwritten when the loop iterations are not fully independent, but in this case all variables except for `ft1r` and `ft1i` are read-only. For these remaining two variables, the writes are completely

Code sample 4.3: A loop that can be parallelised using OpenMP

```
// Previously defined variables
t_non* non;
float* Urs, Uis, Rs, Cs;
float** ft1r, ft1i;
int elements;

// Loop with OpenMP ([36], line 498)
int t1;
#pragma omp parallel for shared(non, Urs, Uis, Rs, Cs, ft1r, ft1i) schedule(static, 1)
for (t1 = 0; t1 < non->tmax1; t1++) {
    propagate_double_sparce_ES(non, Urs, Uis, Rs, Cs, ft1r[t1], ft1i[t1], elements, non->ts);
}
```

independent to separate chunks of the array.

The `schedule(static, 1)` keyword indicates how the work should be divided. Usually, OpenMP will look at the range of iterations in the loop (in this case: `t1` from 0 to `tmax1`), and divide this range into equal, contiguous parts for each thread. However, in this case the calculation for higher values of `t1` is significantly more time-consuming, as the system has to be propagated over a longer timespan. Therefore, we will use a static scheduling with size 1. This will alternatingly assign loop iterations to all threads, so the total runtime of each thread is similar. This behaviour is also called *round-robin* and the comparison with the default behaviour is illustrated in figure 4.1.

One could imagine other, more elaborate scheduling systems. For example, dynamic scheduling could be employed where loop iterations are handed out in chunks based on demand: as soon as a thread finishes its chunk of iterations, it requests the next chunk. However, for the relatively small problem size we are dealing with in this case, that would provide unnecessary communication overhead that would defeat the purpose of parallelisation.

4.5.2 MPI

The implementation of MPI in the simulation is slightly more tricky, mainly due to the increased 'boilerplate' code required for using the standard. The implementation starts with laying a foundation for the architecture of the 'network' of all parallel processes. Then, a main process will determine the work set, i.e. the sample/polarisation direction combinations to compute, for each

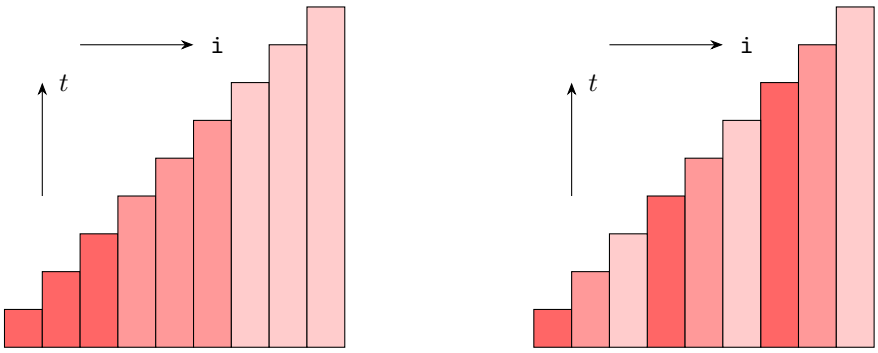


Figure 4.1: Illustration of parallelisation efficiency for static domain decomposition using contiguous blocks (left) and round-robin distribution (right). Different colours indicate different threads. The total computational time on the left is much longer than on the right, since the lightest shade of red entails the three largest computational tasks.

Code sample 4.4: MPI Setup code ([39], lines 30-54)

```
// Initialise MPI
int parentRank, subRank, parentSize, subSize;
MPI_Comm subComm, rootComm;

MPI_Comm_rank(MPI_COMM_WORLD, &parentRank);
MPI_Comm_size(MPI_COMM_WORLD, &parentSize);

// We split up the processing in smaller chunks, each set of MPI processes will make shared memory
// for the global state. Then only the master processes within each chunk will communicate among
// each other, to the main master that will also do all logfile printing and reductions.
MPI_Comm_split_type(MPI_COMM_WORLD, MPI_COMM_TYPE_SHARED, 0, MPI_INFO_NULL, &subComm);
MPI_Comm_rank(subComm, &subRank);
MPI_Comm_size(subComm, &subSize);

// We now make another communicator that contains only all roots of the newly created subComms
MPI_Comm_split(MPI_COMM_WORLD, subRank == 0 ? 0 : MPI_UNDEFINED, 0, &rootComm);
```

process using static domain decomposition, and it will at the end gather all results and store them.

MPI works with so-called *communicators* that represent a set of processes that can communicate with each other, all with a unique ID. To allow for the most generic setup with future expansion of the system in mind, we split all processes (the 'world', `MPI_COMM_WORLD`) into groups (communicators) of processes that live on the same physical machine. Furthermore, we make another communicator containing all main processes from each group. The code used for this is summarised in code sample 4.4. We now have three communicators: world - with all processes, and a root process with ID 0, local - with all processes on a single machine, with a main process for that machine, and roots - with all main processes for a machine and the root of all processes.

The idea behind this tiered approach is as follows: the results from all runs running on the same machine will first be combined. This should be quick, as no network communication is necessary. Then, there will only be one message per machine to all other machines to gather the other results. Especially with large numbers of processes, this speeds up the calculations.

The initialisation continues with determining the settings and parameters for the program and sharing those between all processes using a number of `MPI_Bcast()` calls. After that, the root process will calculate the work set for each process. This is done using static domain decomposition: each process will receive a contiguous block of work items, similar to the left diagram in figure 4.1. This is the easiest and does not pose problems, since each work item takes roughly the same amount of time and there is no relation between runtime and polarisation direction. Each process will receive an array of sample/polarisation direction combinations that it will go and calculate.

The procedure is shown in code sample 4.5. First, the work set calculation is done by the root process by simply determining how many work items there are at all. Then, the root process divides the work items evenly across all processes. If the number of work items is not evenly divisible by the number of processes, some processes might be assigned one more work item. Therefore, it is recommended to choose the settings such that this is not the case. Lastly, the processes communicate the work set per process. This code illustrated the functioning of MPI: all processes should call the same functions with the same arguments, but the behaviour (send or receive) will depend on whether the current process is the root process in the communicator or not.

Finally, all partial results are combined. Since we want to calculate the average, we should add all partial results first. Luckily, MPI has this functionality built-in using a so-called 'reduce' operation. This will take a number (or array) from all processes, and add (or subtract, multiply, etc.) all numbers together and store the result in the main process. This is illustrated in code sample 4.6. In practice, this process happens twice for both the per-machine and final reductions.

Code sample 4.5: MPI Work set computation

```
// Determine all work items to be calculated ([40], lines 70-102)
// This is an array of 2 * number of work items, as each work item is represented
// with two integers: the sample number and polarisation direction.
int *workset = calloc(2 * 21 * sampleCount, sizeof(int));

int currentWorkItem = 0;
for (int currentSample = non->begin; currentSample < non->end; currentSample++) {
    // Skipped some condition checking
    // Set work items
    for (int molPol = 0; molPol < 21; molPol++) {
        (*workset)[currentWorkItem * 2] = currentSample;
        (*workset)[currentWorkItem * 2 + 1] = molPol;
    }
}

int* worksetSizes = calloc(processCount, sizeof(int));
// Determine, send, and receive work items per process ([36], lines 38-67)
if (parentRank == 0) {
    int baseWorksetSize = totalWorkItems / processCount;
    int remainder = totalWorkItems % processCount; // dealing with non-integer divisions

    int* worksetOffsets = calloc(processCount, sizeof(int));
    for (int i = 0; i < processCount; i++) {
        worksetSizes[i] = baseWorksetSize * 2 + (i < remainder) * 2;
        worksetOffsets[i] = (i == 0) ? 0 : worksetOffsets[i-1] + worksetSizes[i];
    }

    // Send worksets as root process
    MPI_Bcast(worksetSizes, processCount, MPI_INT, 0, MPI_COMM_WORLD);
    workset = malloc(worksetSizes[0] * sizeof(int));
    MPI_Scatterv(fullWorkset, worksetSizes, worksetOffsets, MPI_INT, workset, worksetSizes[0],
    ↪  MPI_INT, 0, MPI_COMM_WORLD);
} else {
    // Receive worksets
    MPI_Bcast(worksetizes, processCount, MPI_INT, 0, MPI_COMM_WORLD);
    workset = malloc(worksetSizes[parentRank] * sizeof(int));
    MPI_Scatterv(NULL, NULL, NULL, MPI_INT, workset, worksetSizes[parentRank], MPI_INT, 0,
    ↪  MPI_COMM_WORLD);
}
```

Code sample 4.6: MPI Reduction of results ([36], lines 586-610)

```
// The following variables have the correct values populated
int arraySize;
float** reducationArray[12] = { ... }; // 12 arrays of size arraySize representing the complex
↪  values R_I and R_II in three different directions

if (rank == 0) { // root process will do in-place reduction to save memory
    for (int i = 0; i < 12; i++) {
        MPI_Ireduce(MPI_IN_PLACE, reductionArray[i][0], arraySize, MPI_FLOAT, MPI_SUM, 0, comm,
        ↪  NULL);
    }
} else {
    for (int i = 0; i < 12; i++) {
        MPI_Ireduce(reductionArray[i][0], NULL, arraySize, MPI_FLOAT, MPI_SUM, 0, comm, NULL);
    }
}
```

Chapter 5

Results

With our understanding of theory and parallelisation we can now look into the results. As mentioned before, we will focus on C8S3 [4] in our simulations, as experimental data is readily available [6–9]. We will calculate the two-dimensional spectra of the system for different values of disorder and different polarisation orientations, in order to gain more insight into the dynamics of the system.

5.1 C8S3 Aggregate

The C8S3 aggregate is a so-called amphiphilic cyanine dye; a class of aggregates that exhibit self-assembling behaviour. As illustrated in figure 5.1b, C8S3 takes the form of nanotubes with a double wall. This behaviour is caused by the fact that the individual molecules that constitute this aggregate are *amphiphilic*: they contain both a hydrophobic and a hydrophilic side. This is apparent from the molecular structure, as illustrated in figure 5.1a, where both sides are marked accordingly. In the final tube, the hydrophilic sides are oriented outward.

C8S3 is of particular interest, as it has a potential to serve as an efficient mode of exciton transport. Typically, the excitons will appear on the outer wall, and travel on the inner wall [6]. To further understand the behaviour and structure of this inner wall, we will run simulations on an isolated inner wall. The results can tell us something about the uniformity of the aggregates, and the interaction dynamics.

In order to accurately run the simulation, we first need to obtain values for the time-dependent variables in the Hamiltonian (eq. 3.2). Fortunately, previous research [5, 20, 24] offers us concrete data to use for this, so we won't treat this in detail. Instead, we will focus on the simulations and its results.

5.2 Simulations

Earlier versions of the NISE library only supported the OpenMP parallelisation, which was therefore limited to the number of cores in a single machine. Consequently, scaling is limited and the calculations become impractical for simulations with larger system sizes, higher temporal or frequency resolution, or more samples. The new code, parallelised with MPI as well as OpenMP, offers a significant scaling advantage.

For example, using the old code, the spectrum for $N = 1000$ molecules, all 21 polarisation directions [27], but just a single sample could be calculated in just under a week on a machine with 24 cores (2x Intel Xeon E5 2680v3), whereas the new code can calculate 11 samples, with a $2\times$ higher frequency resolution (increasing the maximal values of t_1 and t_3) in the same time, using 504 cores

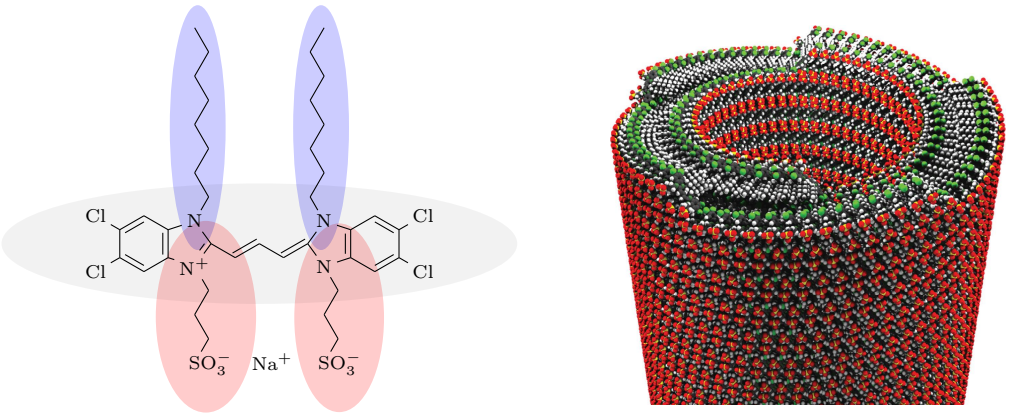


Figure 5.1: Chemical structure of C8S3. (a) The structural formula, with the hydrophobic groups coloured blue and the hydrophilic groups coloured red. (b) 3D computer visualisation of the entire complex with the two tubes, illustrating how the molecules combine. Again, the red parts are hydrophilic, but this time the green parts are hydrophobic (illustration from [41])

(42 machines of 12 cores, 1x Intel Xeon E5 2680v3). This is approximately a 22× speed-up, close to the theoretical maximum. This shows that the parallelisation works well and has the desired effect.

5.3 Analysis

This speed-up allowed for the calculation of many different spectra in a relatively short time-frame, enabling us to do more in-depth analysis of the aggregate. There are three aspects of the aggregate we want to investigate in particular: 1) the naturally occurring amount of disorder in excitation energies, 2) the effect of disorder on the behaviour of the system, and 3) the differences in response for different polarisation orientations.

5.3.1 Naturally occurring disorder

The disorder in a system is a measure of the time-dependent excitation energy differences between molecules, which roughly maps to the broadness of the excitation band, as discussed in section 2.1.4. It also influences the exciton transport capabilities, as 'bumps' in the energy surface slow down this transport. Therefore, it is important to understand how this disorder manifests itself in the system, and how it affects the two-dimensional spectra. With that knowledge, we can draw parallels with the real systems, and thus predict how they function.

The first objective is to let the simulations correspond to experimental data, by tuning the disorder in the model. To accurately and quickly find an appropriate value, the linear absorptive spectrum was used, shown in figure 5.2. Such a linear spectrum is easy to calculate (see section 3.2.1) and provides fast results.

In the figure, the experimental spectrum (red) from [6] is shown, serving as the 'ground truth' to which the disorder parameter is to be fitted. Simulated spectra for two system sizes ($N = 1000$ and $N = 500$) are shown as well, for various different values of the disorder. It is clear that differences in disorder mainly manifest themselves in the FWHM of the first (*primary*) peak in the spectrum, so we will use this to find the best fit.

The primary peak in the experimental data has a FWHM of $105\,\mathrm{cm}^{-1}$. For $N = 1000$, the simulation with $\sigma = 250\,\mathrm{cm}^{-1}$ has the closest matching FWHM, while for $N = 500$ this is $\sigma = 125\,\mathrm{cm}^{-1}$. These three results are shown on the right-hand side of figure 5.2 and show a close match in the

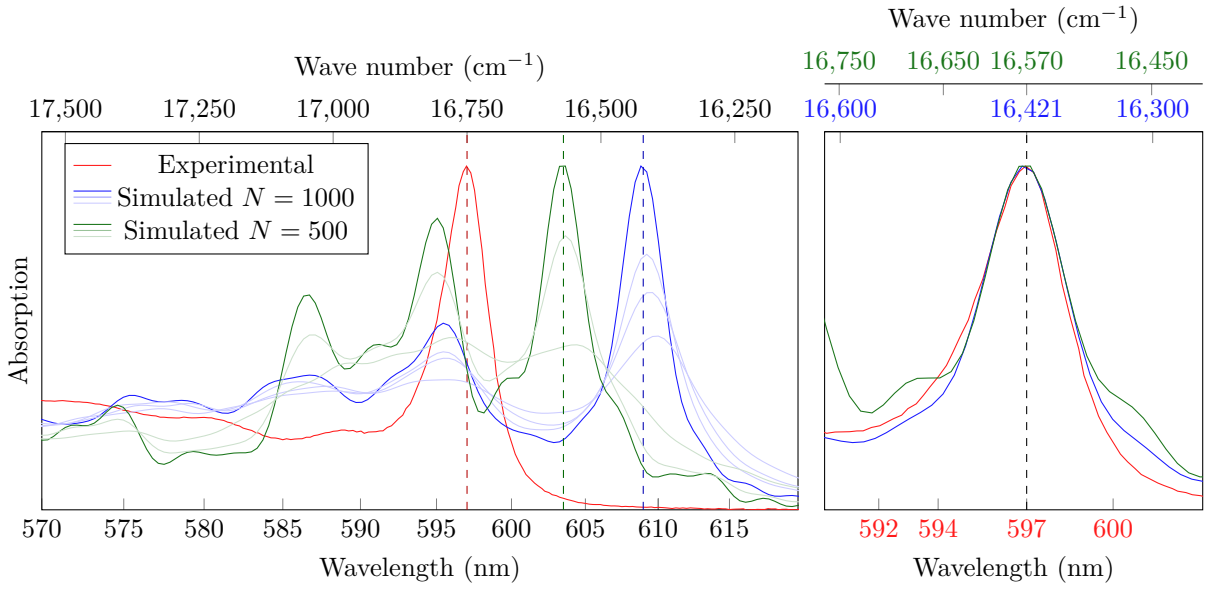


Figure 5.2: Linear absorption spectra for C8S3, both experimental (red) and simulated (blue, green) data. Absorption in arbitrary units. **Left**: wide part of the spectrum with multiple simulated spectra with different values of disorder. For $N = 1000$: $\sigma \in \{250, 350, 400, 500\}\,\mathrm{cm}^{-1}$, for $N = 500$: $\sigma \in \{125, 250, 500\}\,\mathrm{cm}^{-1}$. Primary peaks in the spectrum indicated with a vertical dashed line. **Right**: Close-up of the spectrum with a relative shift to make the primary peaks align, only the best-matching simulations are shown.

peak shape. On the left-hand side of the same figure it can be seen that simulations with higher values of disorder show a broader peak (drawn with a lighter colour compared to the best fitting values); for very high values, the peak is not even clearly distinguishable. This is to be expected, as the primary effect of the disorder is to broaden the absorption band (see section 2.1.4).

Previous molecular dynamics simulations of the disorder in C8S3 have found values of $\sigma = 213\,\mathrm{cm}^{-1}$ for an entire tube [24], so the values found here are in line with those simulations.

Two more features in the linear spectrum are interesting to observe: first of all, the simulated spectra show more peaks at higher energies beyond the primary peak compared to the experimental data. This is likely an effect of the small aggregate sizes considered here. Smaller systems allow the excitations to become quantised over the entire tube [42], since the couplings between the molecules are very strong. For larger systems, the excitations average out and the features diminish, which can already be seen by comparing $N = 500$ and $N = 1000$: the former has more significant peaks in the higher energies than the latter.

This quantisation effect is likely also responsible for the fact that both system sizes require different amounts of disorder to obtain the same FWHM. A smaller system will experience a stronger effect from a certain amount of disorder, since the effect can now spread over the entire system and is not contained locally.

The second interesting feature is that the first peaks are redshifted compared to experimental data. This shift is $180\,\mathrm{cm}^{-1}$ for $N = 500$, and $329\,\mathrm{cm}^{-1}$ for $N = 1000$. The easiest explanation of this is that it simply is a relative offset due to unknown parameters in the model. However, another factor which might contribute to this is a recently-discovered disorder in the coupling between molecules [24], which cannot currently be considered in the NISE scheme.

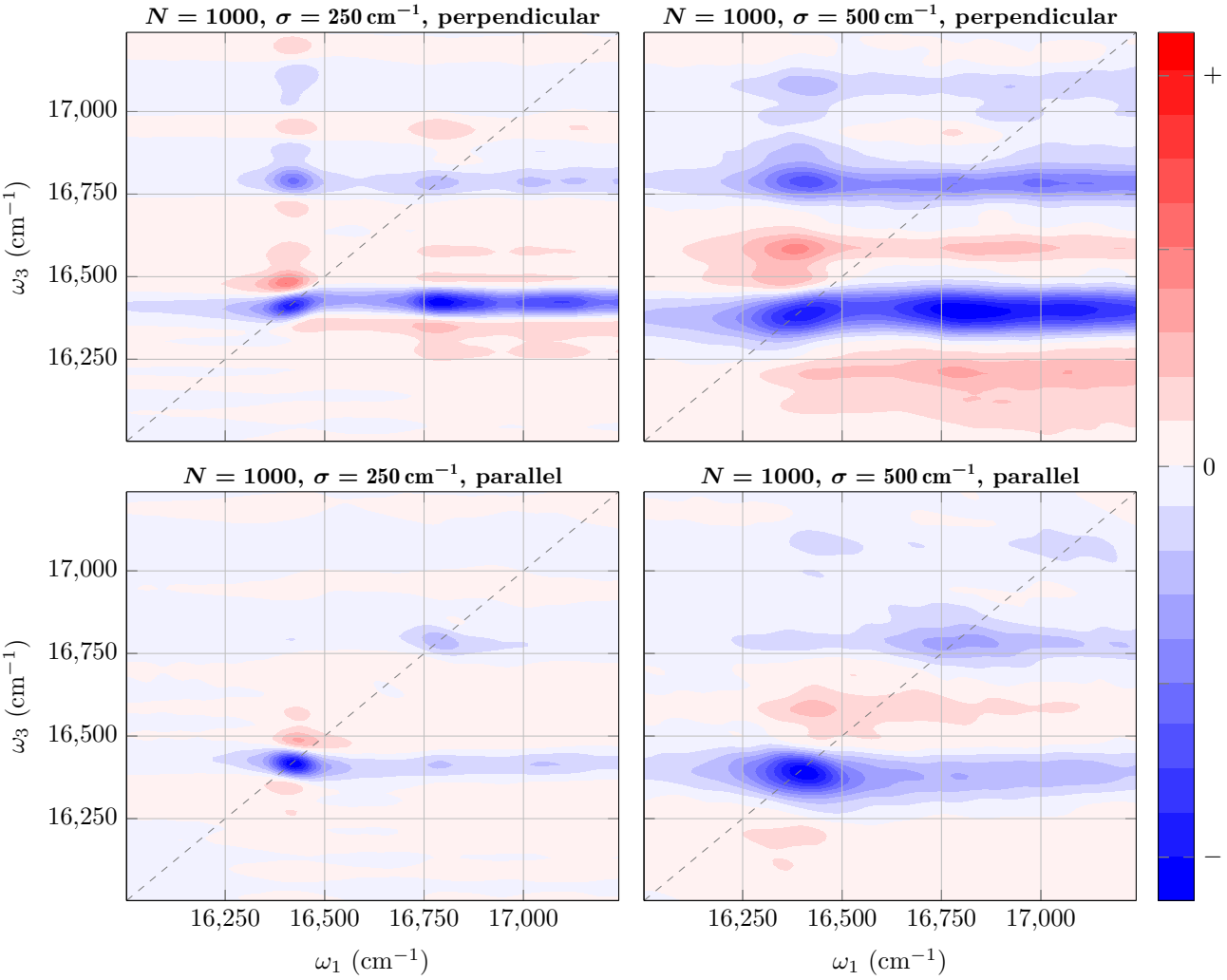


Figure 5.3: Relative absorptive two-dimensional spectra for C8S3, with a time-delay $t_2 = 0$, for $N = 1000$. Spectra for a disorder of $\sigma = 250\,\mathrm{cm}^{-1}$ (left) and $\sigma = 500\,\mathrm{cm}^{-1}$ (right), both perpendicular (top) and parallel (bottom) polarisations shown. The colour scale is from -1 to 1, relative to the maximal value in the plot. The diagonal $\omega_1 = \omega_3$ is shown for clarity.

5.3.2 Effect of disorder

The second aspect to look into is the effect of disorder on the system. As already became apparent in the linear absorption spectra (fig. 5.2), a higher disorder manifested itself in a less pronounced first peak, with a smoother 'tail'. This trend transfers to the two-dimensional spectra in fig. 5.3: the peaks are much broader, more spread-out and not as well-defined.

It might seem like the magnitude of the peaks is much greater in the spectra with higher disorder, but this is merely an effect of the normalisation. Each plot is normalised independently, as the response magnitudes of the spectra with higher disorder are about $10\times$ lower.

In general, the simulations with a disorder corresponding to experiment (as described above) produce a two-dimensional spectrum that also shares many features with the spectra gained from experiment [6, 7]: there is a main peak around $16\,500\,\mathrm{cm}^{-1}$, with cross-peak bands that extend along ω_1. Just like with the linear spectrum, this main peak is redshifted, most likely due to the smaller system size, the strong coupling in the system itself, and some unknown parameters in the model. Furthermore, the diagonal shape of the peak is shared between the simulations and experiment.

A difference in the spectra is again seen in the higher energies, where the simulations show much more detail and other peaks than experiment. This too can be attributed to the limited system size being simulated here, causing local inhomogeneities in the system to have a rather large effect in the system.

5.3.3 Polarisation directions

As described in section 2.1.5, the polarisation configuration has an influence on the final spectra. For both disorder values, both the perpendicular and parallel two-dimensional spectra were obtained, as shown in figure 5.3. In the top row, the perpendicular configuration is shown, while in the bottom row, the parallel orientation is shown.

Comparing the two polarisation directions clearly reveals that the horizontal cross-peak 'band' that is present in the perpendicular configuration is much less pronounced in the parallel one. This indicates that the transitions in this band are oriented perpendicular to each other, which corresponds to experiment [24].

Another peculiar difference between the parallel and perpendicular spectra is the shape of the peak: this is diagonal for the perpendicular, but anti-diagonal for the parallel orientation. The diagonal shape in the perpendicular spectrum can easily be explained by the disorder in the system, causing the peak to be diagonally elongated. However, the anti-diagonal shape is surprising and there is no good explanation for this yet. A possible explanation for this could lie in the precise line shape of the primary peak in the linear spectrum (fig. 5.2, right). For the simulated spectra, there are small 'bumps' to both sides of the primary peak. This is easier to see for the spectrum of $N = 500$, but is also present in the spectrum for $N = 1000$ when compared to the experimental data. These bumps might represent other excitations that are quite weak in a linear spectrum, but might become stronger when the system is already in an excited state.

It is quite unlikely that the shape is a 'fluke' caused by the disorder in the system, as we could expect those to be averaged out over the 11 separate runs that constitute these spectra. This fact hints that there is some underlying physical explanation instead. It might be possible to test this hypothesis in experiment by measuring the spectrum during assembly, at a time when the aggregates are still small. As the time spans in two-dimensional spectroscopy are typically extremely short, this aggregate assembly should not interfere with these measurements.

Chapter 6

Conclusions and Discussion

In short, this thesis set out to improve and speed up the existing code that implements the NISE scheme. As shown above, this goal was achieved and given a sufficiently large problem, the new code can scale extremely well. This allowed us to calculate more involved spectra of larger systems while averaging over multiple runs with a higher frequency resolution.

6.1 C8S3 spectra

These new results allow us to confirm several findings from previous experiments. First of all, we found concrete values for the disorder in the system, dependent on its size, and an efficient and reliable method to find them. Future research could focus on determining the disorder for more system sizes, and investigating a relation between the two. This could also be done with the help of molecular dynamics simulations [24], for an alternative approach to compare the results.

Secondly, we can confirm that the simulated two-dimensional spectra correspond to experimental findings, thus confirming theoretical assumptions of the system. Unfortunately, there are still some aspects that do not correspond to experiment: the peaks are redshifted and the cross-peaks are much stronger than one would expect from experiment. Incorporating the disorder in coupling into NISE could be an approach to account for the redshifted peaks, while a more thorough analysis for different waiting times t_2 could open up more information on the dynamics of the simulated system. Analysis of even larger systems could give more information about the (cross-)peaks at higher energies.

An aspect of the results presented here that is not yet understood, is the anti-diagonal peak shape as observed in the two-dimensional spectra for the parallel polarisation orientation. This could be related to the small system sizes considered in this thesis, but future research is necessary to further investigate the origins of this behaviour.

6.2 Simulation code

The new version of the simulation code is now generally available [11], and has already been used for other research [14]. With the parallelisation implemented, it could also be used for other systems, such as various light-harvesting complexes like LHCII, LH2, and chlorosomes [1, 2, 12, 13], or other energy transport systems [43].

While the parallelisation is a major improvement, there always remain future opportunities. For example, a more detailed look at the possibilities for GPU utilisation would be useful, as this might speed up some computationally expensive matrix operations. Unfortunately, in the program's current form, this is not trivial and it might require adaptation of the algorithm.

Other areas of improvement lie in the robustness of the program. Currently, all parallelisation is done through MPI and OpenMP, both of which do not provide fault-tolerance. Especially with longer run times on more machines, it is increasingly more likely that at some point a machine will fail or some other error occurs. At the moment, this renders all computational work up to that point useless. This could be tackled by storing intermediate results somewhere, or using frameworks such as Apache Spark [44], Apache Flink [45], or Chapel [46] that offer checkpoints and dynamic workload balancing, taking much of the work out of the programmer's hands.

Bibliography

[1] Paula I. van Noort et al. "Chlorosomes of green sulfur bacteria: Pigment composition and energy transfer". In: *Photosynthesis Research* 41.1 (July 1994), pp. 193–203. ISSN: 1573-5079. DOI: 10.1007/BF02184160.

[2] Bartlomiej Drop et al. "Light-harvesting complex II (LHCII) and its supramolecular organization in Chlamydomonas reinhardtii". In: *Biochimica et Biophysica Acta (BBA) - Bioenergetics* 1837.1 (2014), pp. 63–72. ISSN: 0005-2728. DOI: 10.1016/j.bbabio.2013.07.012.

[3] Erik Bloemsma et al. "Signature of Anomalous Exciton Localization in the Optical Response of Self-Assembled Organic Nanotubes". In: *Physical review letters* 114 (Apr. 2015), p. 156804. DOI: 10.1103/PhysRevLett.114.156804.

[4] Cătălin Didraga et al. "Structure, Spectroscopy, and Microscopic Model of Tubular Carbocyanine Dye Aggregates". In: *The Journal of Physical Chemistry B* 108.39 (2004), pp. 14976–14985. DOI: 10.1021/jp048288s.

[5] Björn Kriete et al. "Steering Self-Assembly of Amphiphilic Molecular Nanostructures via Halogen Exchange". In: *The Journal of Physical Chemistry Letters* 8.13 (2017). PMID: 28594561, pp. 2895–2901. DOI: 10.1021/acs.jpclett.7b00967.

[6] Björn Kriete et al. "Interplay between structural hierarchy and exciton diffusion in artificial light harvesting". In: *Nature Communications* 10.1 (Oct. 2019), p. 4615. ISSN: 2041-1723. DOI: 10.1038/s41467-019-12345-9.

[7] Jaroslaw Sperling et al. "Excitons and Disorder in Molecular Nanotubes: A 2D Electronic Spectroscopy Study and First Comparison to a Microscopic Model". In: *The Journal of Physical Chemistry A* 114.32 (2010). PMID: 20701329, pp. 8179–8189. DOI: 10.1021/jp102173n.

[8] Franz Milota et al. "Vibronic and Vibrational Coherences in Two-Dimensional Electronic Spectra of Supramolecular J-Aggregates". In: *The Journal of Physical Chemistry A* 117.29 (2013). PMID: 23461650, pp. 6007–6014. DOI: 10.1021/jp3119605.

[9] Dörthe M. Eisele et al. "Robust excitons inhabit soft supramolecular nanotubes". In: *Proceedings of the National Academy of Sciences* 111.33 (2014), E3367–E3375. ISSN: 0027-8424. DOI: 10.1073/pnas.1408342111.

[10] Thomas la Cour Jansen and Jasper Knoester. "Nonadiabatic Effects in the Two-Dimensional Infrared Spectra of Peptides: Application to Alanine Dipeptide". In: *The Journal of Physical Chemistry B* 110.45 (2006). PMID: 17092043, pp. 22910–22916. DOI: 10.1021/jp064795t.

[11] *NISE implementation*. URL: https://github.com/GHlacour/NISE_2017.

[12] Marco Ferretti et al. “Dark States in the Light-Harvesting complex 2 Revealed by Two-dimensional Electronic Spectroscopy”. In: *Scientific Reports* 6.1 (Feb. 2016), p. 20834. ISSN: 2045-2322. DOI: 10.1038/srep20834.

[13] Tenzin Kunsel et al. “Simulating Fluorescence-Detected Two-Dimensional Electronic Spectroscopy of Multichromophoric Systems”. In: *The Journal of Physical Chemistry B* 123.2 (2019), pp. 394–406. DOI: 10.1021/acs.jpcb.8b10176.

[14] Andy S. Sardjan et al. “Observation of ultrafast coherence transfer and degenerate states with polarization controlled two-dimensional electronic spectroscopy”. In preparation.

[15] Thomas L.C. Jansen and Jasper Knoester. “Waiting Time Dynamics in Two-Dimensional Infrared Spectroscopy”. In: *Accounts of Chemical Research* 42.9 (2009). PMID: 19391619, pp. 1405–1411. DOI: 10.1021/ar900025a.

[16] O. Golonzka et al. “Coupling and orientation between anharmonic vibrations characterized with two-dimensional infrared vibrational echo spectroscopy”. In: *The Journal of Chemical Physics* 115.23 (2001), pp. 10814–10828. DOI: 10.1063/1.1417504.

[17] Thomas L.C. Jansen. *Multi-dimensional spectroscopy*. 2009. URL: https://www.theochem.ru.nl/han/LaCourJansen-multidim-spectrosc-Han11jan2010.pdf (visited on 07/10/2020).

[18] Peter Hamm. *Principles of Nonlinear Optical Spectroscopy: A Practical Approach*. 2005. URL: https://www.chem.uci.edu/~dmitryf/manuals/Fundamentals/Mukamel%20for%20dummies.pdf (visited on 07/10/2020).

[19] Peter Hamm and Martin Zanni. *Concepts and Methods of 2D Infrared Spectroscopy*. Cambridge University Press, 2011. DOI: 10.1017/CBO9780511675935.

[20] Anna S. Bondarenko, Jasper Knoester, and Thomas L.C. Jansen. “Comparison of methods to study excitation energy transfer in molecular multichromophoric systems”. In: *Chemical Physics* 529 (2020), p. 110478. ISSN: 0301-0104. DOI: 10.1016/j.chemphys.2019.110478.

[21] Jasper Knoester. “Optical properties of molecular aggregates”. In: *Organic Nanostructures: Science and Applications*. Proceedings of the International School of Physics ”Enrico Fermi”. Vol. 149. IOS Press, pp. 149–186. DOI: 10.3254/978-1-61499-005-5-149.

[22] Carlos R. Baiz et al. “Vibrational Spectroscopic Map, Vibrational Spectroscopy, and Intermolecular Interaction”. In: *Chemical Reviews* 0.0 (0). PMID: 32598850, null. DOI: 10.1021/acs.chemrev.9b00813.

[23] Dan Cringus et al. “Ultrafast anisotropy dynamics of water molecules dissolved in acetonitrile”. In: *The Journal of Chemical Physics* 127.8 (2007), p. 084507. DOI: 10.1063/1.2771178.

[24] Anna S. Bondarenko et al. “Multiscale Modeling of Molecular Structure and Optical Properties of Complex Supramolecular Aggregates”. In: (June 2020). DOI: 10.26434/chemrxiv.12279956.v1.

[25] Robin M. Hochstrasser. “Two-dimensional IR-spectroscopy: polarization anisotropy effects”. In: *Chemical Physics* 266.2 (2001), pp. 273–284. ISSN: 0301-0104. DOI: 10.1016/S0301-0104(01)00232-4.

[26] Thomas la Cour Jansen and Jasper Knoester. “A transferable electrostatic map for solvation effects on amide I vibrations and its application to linear and two-dimensional spectroscopy”. In: *The Journal of Chemical Physics* 124.4 (2006), p. 044502. DOI: 10.1063/1.2148409.

[27] Chungwen Liang and Thomas L. C. Jansen. "An Efficient N3-Scaling Propagation Scheme for Simulating Two-Dimensional Infrared and Visible Spectra". In: *Journal of Chemical Theory and Computation* 8.5 (2012). PMID: 26593664, pp. 1706–1713. DOI: 10.1021/ct300045c.

[28] William J. Bolosky and Michael L. Scott. "False Sharing and Its Effect on Shared Memory Performance". In: *USENIX Systems on USENIX Experiences with Distributed and Multiprocessor Systems - Volume 4*. Sedms'93. San Diego, California: USENIX Association, 1993, p. 3.

[29] Christopher Hollowell et al. "The Effect of NUMA Tunings on CPU Performance". In: *Journal of Physics: Conference Series* 664.9 (Dec. 2015), p. 092010. DOI: 10.1088/1742-6596/664/9/092010.

[30] Chao-Tung Yang, Chih-Lin Huang, and Cheng-Fang Lin. "Hybrid CUDA, OpenMP, and MPI parallel programming on multicore GPU clusters". In: *Computer Physics Communications* 182.1 (2011). Computer Physics Communications Special Edition for Conference on Computational Physics Kaohsiung, Taiwan, Dec 15-19, 2009, pp. 266–269. ISSN: 0010-4655. DOI: https://doi.org/10.1016/j.cpc.2010.06.035.

[31] R. Rabenseifner, G. Hager, and G. Jost. "Hybrid MPI/OpenMP Parallel Programming on Clusters of Multi-Core SMP Nodes". In: *2009 17th Euromicro International Conference on Parallel, Distributed and Network-based Processing*. 2009, pp. 427–436.

[32] R. Rabenseifner, G. Hager, and G. Jost. *Hybrid MPI and OpenMP Parallel Programming*. Half-day Tutorial at SC13, Denver. Nov. 2013. URL: https://www.openmp.org/press-release/sc13-tutorial-hybrid-mpi-openmp-parallel-programming/.

[33] Gabriele Jost, Hao-Qiang Jin, Ferhat F Hatay, et al. *Comparing the OpenMP, MPI, and hybrid programming paradigm on an SMP cluster*. Tech. rep. NASA, Nov. 2003. URL: https://www.nas.nasa.gov/assets/pdf/techreports/2003/nas-03-019.pdf.

[34] Leonardo Dagum and Ramesh Menon. "OpenMP: an industry standard API for shared-memory programming". In: *Computational Science & Engineering, IEEE* 5.1 (1998), pp. 46–55.

[35] Message Passing Forum. *MPI: A Message-Passing Interface Standard*. Tech. rep. USA, 1994.

[36] calc_2DES.c. URL: https://github.com/GHlacour/NISE_2017/blob/master/src/calc_2DES.c.

[37] Nicholas Nethercote and Julian Seward. "Valgrind: A Framework for Heavyweight Dynamic Binary Instrumentation". In: *SIGPLAN Not.* 42.6 (June 2007), pp. 89–100. ISSN: 0362-1340. DOI: 10.1145/1273442.1250746.

[38] Microsoft. *Measure application performance by analyzing CPU usage*. Apr. 2019. URL: https://docs.microsoft.com/en-us/visualstudio/profiling/beginners-guide-to-performance-profiling?view=vs-2019.

[39] NISE.c. URL: https://github.com/GHlacour/NISE_2017/blob/master/src/NISE.c.

[40] MPI_subs.c. URL: https://github.com/GHlacour/NISE_2017/blob/master/src/MPI_subs.c.

[41] Dörthe M. Eisele et al. "Uniform exciton fluorescence from individual molecular nanotubes immobilized on solid substrates". In: *Nature Nanotechnology* 4.10 (Oct. 2009), pp. 658–663. ISSN: 1748-3395. DOI: 10.1038/nnano.2009.227.

[42] Anna S. Bondarenko, Thomas L. C. Jansen, and Jasper Knoester. “Exciton localization in tubular molecular aggregates: Size effects and optical response”. In: *The Journal of Chemical Physics* 152.19 (2020), p. 194302. DOI: 10.1063/5.0008688.

[43] Andreas T. Haedler et al. “Long-range energy transport in single supramolecular nanofibres at room temperature”. In: *Nature* 523.7559 (July 2015), pp. 196–199. DOI: 10.1038/nature14570.

[44] Apache Software Foundation. *Apache Spark*. URL: https://spark.apache.org/.

[45] Apache Software Foundation. *Apache Flink*. URL: https://flink.apache.org/.

[46] Chapel. *Chapel*. URL: https://chapel-lang.org/.