

High-sensitivity measurements of weak magnetic field effects

Marcin Konowalczyk

A Thesis submitted for the degree of
Doctor of Philosophy

Somerville College
University of Oxford

April 30th, 2020

Abstract

The study of spin chemistry and, in specific, magnetosensitivity has been growing in importance in recent years. Not only has there been great advances in the field of magnetic field effects (MFEs) on biological reactions,[1–3] but also on their applications in semiconductor[4;5] and polymer science.[6]

Many chemical systems exhibit sensitivity to comparatively weak magnetic fields[7;8] by a variety of mechanisms.[9–11] The most prevalent of such magnetic field effect (MFE) mechanisms in solution is the radical pair (RP) mechanism,[10] which will be investigated in this work. **Chapter 1** will introduce ideas behind the RP-based magnetosensitivity: the quantum mechanics of spin, the theory of long-range electron transfer, chemical kinetics and finally the RP-mechanism itself. The chapter will conclude by describing the magnetosensitivity of flavin-based systems investigated in this work.

A significant part of this work revolved around the understanding, using and development of the methodology involved. **Chapter 2** will discuss some of the mathematical and computational methods used throughout this work. The methods covered are the time-frequency analysis, fitting, numerical solutions to differential equations and analysis of uncertainty. The examples used throughout this chapter have been selected to illustrate important points of their respective subjects, but also to be relevant to understanding other parts of this work.

Chapter 3 will make use of some of the methodology described in the previous chapter and talk about the development and theoretical treatment of a modulated fluorescence experiment – ModMARY. The chapter will conclude by showcasing the methodology developed to measure a fluorescent exciplex-based system traditionally investigated by ModMARY (pyrene/1,3-dicyanobenzene). An example of this data can be seen in figure 1a. The work done in this chapter is the basis of a publication led by the author[12] (manuscript in preparation).

The flavin-based system often used in magnetosensitivity studies of biological systems is the solution of flavin mononucleotide (FMN) and hen egg-white lysozyme (HEWL). This system is well known, but not fully understood. The interactions of freely diffusing FMN within a crystal lattice of HEWL has been investigated in **Chapter 4**, by confocal laser-scanning microscopy (CLSM). The principle of CLSM, as well as the data analysis methodology developed, will be discussed. Then, the spatiotemporal evolution of the MFE of FMN

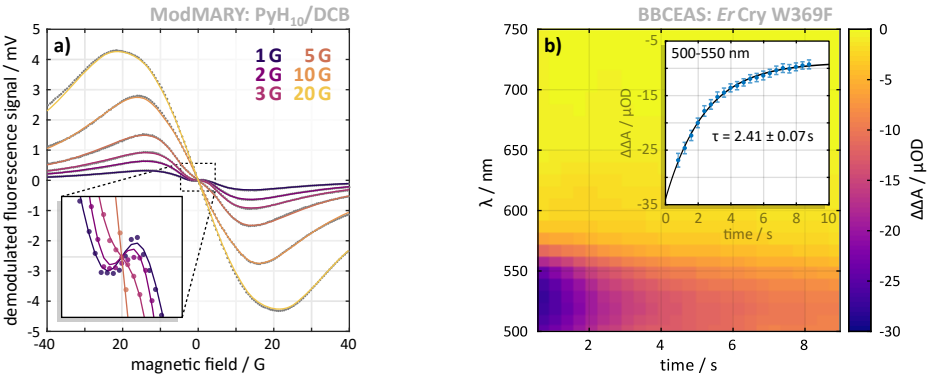


Figure 1: a) Demodulated fluorescence signal of PyH10/DCB at different modulation depths, measured with ModMARY. The insert shows the region of ±2 G. **b)** Time-resolved double-differential change in absorbance of a cryptochrome from *Erithacus rubecula* (European robin).

in HEWL crystals will be explored. The work in this chapter is part of a joined publications with Dejéan *et al.*[13] (manuscript submitted for review).

MFEs can be probed optically not only by fluorescence, as investigated in the aforementioned chapters, but also by absorption techniques. The understanding and development of broadband cavity-enhanced absorption spectroscopy (BBCEAS), and its application to measurements of solution-based cryptochrome samples will be discussed in **Chapter 5**.

The systems of most interest in the field of magnetoreception are the cryptochrome molecules[14] – flavin-binding proteins proposed as the source of the avian magnetic sense.[15] The methodology from previous chapter will be used in **Chapter 6** to measure the samples of cryptochrome proteins from five different animal species and their selected mutants. This data will be compared and its implications will be discussed. An example of a time-resolved MFE can be seen in figure 1b. The work is a part of a joined publications with Xu *et al.*[16] (manuscript submitted for review).

Preface

The primary purpose of this work is to document the research done to further the knowledge in the subject of magnetosensitivity. No research can be completed by one person, and so the intended recipients of this work are the people attempting to reproduce and continue it. For that purpose, the text has been liberally annotated with footnotes and observations one might find helpful. In many places, especially in chapter 2, many technical terms from different fields of science are used. This is done to familiarise the reader with them, and provide a kind of glossary of such terms. This is sometimes referred to as the *Rumpelstiltskin principle* – once one knows what to call something, they are able to use it and reason about it.[17]

There are two, strongly intertwined models of thinking: mathematical and algorithmic. Certain ideas are better expressed in the former and certain in the latter. This work will, therefore, make use of both. Any code listings will be shown in *working* Matlab code. This language will be used akin to pseudocode, as it is relatively simple, self-contained, and it has extensive online documentation available for free documentation. The exact version of Matlab used is 2017b, but care has been taken to avoid using any version-specific features.

Table of Contents

PREAMBLE

Abstract **ii**

Preface **iv**

Table of Contents **v**

MAIN MATTER

1 Introduction to spin chemistry **1**
1.1 Quantum mechanics of spin 1
1.2 Electron transfer reactions 16
1.3 Reaction kinetics 19
1.4 Magnetosensitivity of chemical reactions 22
1.5 Flavin-based systems 28

2 Computational methods for experimental physical chemistry **32**
2.1 Time-frequency analysis 32
2.2 Numerical optimisation 46
2.3 Numerical integration 65
2.4 Uncertainty and its propagation 77

3 Development of overtone modulated MARY spectroscopy **80**
3.1 ModMARY spectroscopy 80
3.2 Modulation spectroscopy 81
3.3 Apparatus 88
3.4 Data analysis 90
3.5 Pyrene/1,3-dicyanobenzene 92
3.6 Results 93
3.7 Conclusions 94

4 Measurements of MFEs by confocal microscopy **96**
4.1 Principle of CLSM 96
4.2 Image data analysis 98
4.3 Spatiotemporal evolution of MFEs 102
4.4 MFE dynamics imaging 108
4.5 Conclusions and outlook 110

5 Magnetic field effect measurements with cavity-enhanced absorption spectroscopy **113**
5.1 Principles of cavity spectroscopy 113
5.2 Resonator alignment 120
5.3 CEAS calibration 122

5.4 BBCEAS for detection of MFEs . . . 124
5.5 Experimental procedures . . . 132
5.6 Blank measurements . . . 137
5.7 Satellite MFE . . . 138
5.8 Outlook . . . 140

6 Study of animal cryptochrome magnetosensitivity 141
6.1 Preliminary studies . . . 143
6.2 *Er*Cry and *Dm*Cry . . . 145
6.3 Avian mutants . . . 150
6.4 Conclusions and outlook . . . 154

BACK MATTER

Acknowledgements 158

Bibliography 167

List of Figures 168

List of Listings 171

APPENDIX

A Additional code listings 175
A.1 Additional examples for FFT . . . 175
A.2 Random search optimisation . . . 176
A.3 Random coloured noise . . . 177
A.4 Eigenvalue optimisation . . . 178
A.5 Subfunctions for the resonator alignment simulation . . . 179
A.6 Polynomial fitting . . . 180
A.7 Model functions . . . 181
A.8 Additional code for ModMARY . . . 183
A.9 Allan deviation . . . 186
A.10 Lorentzian fitting . . . 187
A.11 Liouville von Neumann simulation . . . 189

B Additional figures 192

Main matter

Chapter 1

Introduction to spin chemistry

The subject of spin chemistry is a complicated and subtle interplay between 1) the chemical reaction, electron transfer (ET) and/or charge separation processes which lead to the formation of magnetosensitive species, 2) the quantum mechanics which modulates the interaction of the species involved with each other and with the external magnetic field, and 3) chemical kinetics which allows these effects to manifest in bulk samples. This chapter will introduce these three concepts and then conclude with the overview of the relevant mechanisms of magnetosensitivity.

1.1 Quantum mechanics of spin

Spin-possessing particles, their interactions with each other and the external magnetic fields are inherent to the magnetosensitivity of chemical systems. The properties of spin are closely related those of quantum mechanical angular momentum.

Angular momentum

Quantum-mechanical angular momentum can be described by a *vector operator* $\hat{\boldsymbol{j}}$ and its projection onto three arbitrary spacial dimensions $\hat{j}_i$ such that:

$$\hat{\boldsymbol{j}} = \boldsymbol{x}\hat{j}_x + \boldsymbol{y}\hat{j}_y + \boldsymbol{z}\hat{j}_z \tag{1.1}$$

where $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$ are unit vectors along the axis of the Euclidean coordinate system, and $\hat{j}_x$, $\hat{j}_y$ and $\hat{j}_z$ are the corresponding angular momentum projection operators. The angular momentum can, by analogy to a classical precessing body, be written as a cross product of the position and linear momentum vectors:

$$\hat{\boldsymbol{j}} = \hat{\boldsymbol{r}} \times \hat{\boldsymbol{p}} = -i\hbar\left(\hat{\boldsymbol{r}} \times \hat{\boldsymbol{\nabla}}\right) \tag{1.2}$$

where $\hat{\boldsymbol{r}}$ is the position vector operator, $\hat{\boldsymbol{p}}$ is the *linear* momentum operator, $\times$ indicates the vector cross product, and $\hat{\boldsymbol{\nabla}}$ is a derivative vector operator.[18;19] Replacement of $\hat{\boldsymbol{p}}$ by $-i\hbar\hat{\boldsymbol{\nabla}}$ is referred to as quantum transcription[a] and it allows one to bring equations from classical to quantum mechanics.[20] The components of $\hat{\boldsymbol{j}}$ can be written explicitly (as opposed to the vector notation in 1.2) as:

$$\begin{aligned}
\hat{j}_x &= \hat{y}\hat{p}_z - \hat{z}\hat{p}_y = -i\hbar\left(y\frac{\partial}{\partial z} - z\frac{\partial}{\partial y}\right) \\
\hat{j}_y &= \hat{z}\hat{p}_x - \hat{x}\hat{p}_z = -i\hbar\left(z\frac{\partial}{\partial x} - x\frac{\partial}{\partial z}\right) \\
\hat{j}_z &= \hat{x}\hat{p}_y - \hat{y}\hat{p}_x = -i\hbar\left(x\frac{\partial}{\partial y} - y\frac{\partial}{\partial x}\right)
\end{aligned} \tag{1.3}$$

The eigenvalues of these operators cannot cannot, however, be solved simultaneously and precisely – the eigenstates of $\hat{j}_z$ are not the eigenstates of $\hat{j}_x$ or $\hat{j}_y$. The existence of such solutions is governed by whether the operators in question commute.

[a] This specific quantum transcription is done in the 'position representation', since it uses the eigenbasis of the position operator – $\hat{\boldsymbol{r}}$ as the basis for the angular momentum operator.

The commutation relations play a pivotal role in quantum mechanics. One of the reasons for this is the fact that any observables corresponding to quantum mechanical operators which commute with one another can be specified simultaneously and with arbitrary precision. It is convenient to define a commutator of $\hat{A}$ and $\hat{B}$ as:

$$[\hat{A}, \hat{B}] = \hat{A}\hat{B} - \hat{B}\hat{A} \tag{1.4}$$

The commutator is also an operator itself. $\hat{A}$ and $\hat{B}$ commute if the commutator evaluates to the null operator $\hat{0} = 0$.

The commutators for the projections of the angular momentum can be found based on equations 1.3 as:

$$[\hat{j}_x, \hat{j}_y] = i\hbar\hat{j}_z \qquad [\hat{j}_y, \hat{j}_z] = i\hbar\hat{j}_x \qquad [\hat{j}_z, \hat{j}_x] = i\hbar\hat{j}_y \tag{1.5}$$

The angular momentum projection operators do not, therefore, commute with one another and hence cannot be known simultaneously and with arbitrary precision.[b] It turns out, however, that the total (squared) angular momentum *does* commute with the projection operators, and hence it, as well as one of the projections *can* be specified at the same time. The total angular momentum operator can be written as the inner (dot) product of the an:

$$\hat{j}^2 = \hat{\boldsymbol{j}} \cdot \hat{\boldsymbol{j}} = \hat{j}_x\hat{j}_x + \hat{j}_y\hat{j}_y + \hat{j}_z\hat{j}_z \tag{1.6}$$

When operating on an eigenstate $|\psi\rangle$, the eigenvalues of the total, and the projection angular momentum operators are:

$$\hat{j}^2|\psi\rangle = \hbar^2 j(j+1)|\psi\rangle \tag{1.7}$$

$$\hat{j}_z|\psi\rangle = \hbar m_j|\psi\rangle \tag{1.8}$$

where j and m_j have been used as magnitude, and projection angular momentum quantum numbers. m_j can take values $-j, -j+1, \ldots, j-1, j$ for each value of j, such that angular momentum j has $2j+1$ possible projections onto the quantisation axis. Note that $\hbar$ can be thought of as a 'unit' of angular momentum. For that reason, one often works in an 'atomic units' system in which, amongst other things,[c] $\hbar = 1$. This will *not* be done in this section, but atomic units will be adopted in section 1.2 and 'Boundary value problems' for simplicity of the simulations involved.

Note that, throughout, the choice of the x/y/z Euclidean basis is indeed arbitrary. The basis vectors do not even have to be orthogonal, nor straight,[d] as long as they span the entirety of real 3D coordinate space (often called the 3-space $\mathbb{R}^3$). The choice of Euclidean basis *is* however the most convenient[e] and will be adopted here. Further, if the system in question has an obvious axis of symmetry (e.g. the direction of the external magnetic field), the z-axis of the Euclidean coordinate system will often be chosen to be aligned with it. This, again, is not necessary, merely convenient. Exactly the same results can be obtained by another choice of the orientation of the basis system.

In a further analogy to classical mechanics, an angular momentum of a particle gives rise to a magnetic moment defined as:

$$\hat{\boldsymbol{\mu}} = \gamma\hat{\boldsymbol{j}} \tag{1.9}$$

where γ is the gyromagnetic ratio – the ratio of the particles' magnetic moment to its angular momentum. For classical-like precession e.g. of an electron in an orbital, or a spinning molecule, γ can be related to the

[b] Based on the Heisenberg uncertainty principle, it can be shown that for two Hermitian operators $\hat{A}$ and $\hat{B}$, the lower bound of the product of the uncertainties in their observables is directly related to the expectation value of their commutator. Specifically $\sigma_A\sigma_B \geq \frac{1}{2}\left|\langle[\hat{A}, \hat{B}]\rangle\right|$, where $\sigma_A = \sqrt{\langle\hat{A}^2\rangle - \langle\hat{A}\rangle^2}$ is the standard deviation uncertainty of the observable.

[c] The atomic units system sets the reduced Planck's constant ($\hbar$), mass and charge of the electron (m_e and e) as well as the Coulomb constant ($4\pi\epsilon_0$) to unity. All other units are derived from these basic ones.

[d] Any set of curves can define the axes of the coordinate system in some N-space, as long as there exists a points-to-point mapping between the appropriate N-dimensional Cartesian coordinate system. Such a coordinate system is referred to as *curvilinear* coordinates. A polar coordinate system is a common example of such a coordinate system.

[e] Note, for example, the circular symmetry $x \to y \to z \to x$ of Equations 1.3 and 1.5.

particles' mass m and its charge q:

$$\gamma = \frac{q}{2m} \tag{1.10}$$

Note that this implies that only the angular momentum of charged particles gives rise to a magnetic moment. Since the magnetic moment operator is related to the angular momentum operator by a multiplicative constant, all the properties of the particles angular momentum are carried over to its magnetic moment. $\hat{\boldsymbol{\mu}}$ is therefore quantised in both magnitude and projection, akin to $\hat{\boldsymbol{j}}$.

Since the magnitude of $\hat{\boldsymbol{j}}$ is larger than the largest possible magnitude of its projection ($\sqrt{j(j+1)} > j$ for $j > 0$), the angular momentum vector will never point parallel to the quantization axis. Further, when the projection of $\hat{\boldsymbol{j}}$ onto the z-axis is specified, the x and y-projections are maximally uncertain. These two facts can be represented by showing the angular momentum vector occupying an undefined position on the surface of a cone around the quantization axis. The internal angle θ between the z-axis and the surface of this cone is:

$$\cos(\theta) = \frac{m_j}{\sqrt{j\left(j+1\right)}} \tag{1.11}$$

This can be seen for spin (discussed in the next section) and the corresponding magnetic moment vectors in figure 1.1. Note, though, that this picture can be misleading, as the spin vector as a whole is not an observable since its projections do not commute with one another.[f] This is in contrast to, e.g. the position vector operator $\hat{\boldsymbol{r}}$, whose components *do* commute. This means all of its components can be measured simultaneously and the vector as a whole is observable. The position does not, of course, commute with (linear) momentum $\hat{\boldsymbol{p}}$, and hence that in turn cannot be known if $\hat{\boldsymbol{r}}$ is known.

The quantum mechanical angular momentum is inherently different from its classical analogue. The parallel to classical mechanics made by equation 1.2 restricts the solutions – eigenstates of $\hat{\boldsymbol{j}}$ and $\hat{j}_z$, to spherical harmonics. This allows j to take only positive integer values (including 0). This picture of angular momentum is able to describe quantum-mechanical phenomena which rely on rotational motion in space – orbital motion of electrons or rotation of individual molecules in the gas phase.[g] It cannot, however, fully explain the existence of spin.

Spin

Spin is a fundamental property of particles like electrons, protons and neutrons.[h] It behaves like an inherent, internal angular momentum, consistent with both integer, and half integer solutions to equation 1.7. In order to allow the concept of quantum-mechanical angular momentum to encapsulate spin, equations 1.5 are used as the *definition* of angular momentum, rather than its property. Vector operator $\hat{\boldsymbol{j}}$ is an angular momentum *because* its orthogonal projections into 3-space commute according to equations 1.5. Any vector operator whose components obey equations 1.5 is an angular momentum. This definition relaxes the allowed value of j and permits not only integer, but also half-integer angular momentum quantum numbers.[i]

[f] All of the angular momentum vectors' components cannot be measured simultaneously and precisely. If they were, instead, measured *sequentially*, in the order $x \to y \to z$, the measurement of x would collapse the wavefunction into the state corresponding to the measurement – one also associated with the uncertainty in y and z. Even if the state was not to evolve in time between the measurements, the following measurement of y would, also collapse the wavefunction in a similar manner into a state with uncertainty in x and z. The previous measurement of x, albeit in itself precise at the time, would now have an associated uncertainly. Following the same argument for the sequential z measurement, shows that all the components can be measured sequentially, but not precisely. At the end of the third measurement the system is left in the state where the z component is known precisely, while the uncertainty in x and y places them somewhere on the cone around the z-axis – the same situation as if we were to, simply, measure the z component.

[g] The integer-value quantum number solutions for total angular momentum, and the corresponding projection quantum numbers enumerate spherical harmonic solutions (eigenstates) of equations 1.7 and 1.8. [20]

[h] The property of spin was discovered in 1925 by joined efforts of Uhlenbeck and Goudsmit by analysis of the spectra of excited hydrogen atoms. This was also supported by, then unexplained, results by Stern and Gerlach from 1922 on a beam of silver atoms.

[i] The particles with integer and half-integer spin are called *bosons* and *fermions* respectively.

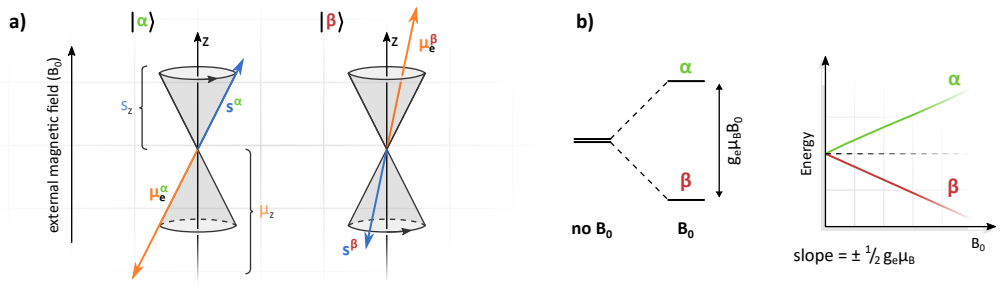


Figure 1.1: α and β spin states of a free electron in an external magnetic field. Note that the magnetic moment vectors are antiparallel to the spin vectors and so the α spin is *raised* and β spin is *lowered* in energy upon the application of the external magnetic field. The arrows on the rim of the cones containing the spin vectors indicate the direction of Larmor precession. The angle between the cone walls and the z-axis is ≈54.74°.

Analogously to equations 1.7, and 1.8, when operating on an eigenket $|\psi\rangle$, the total and the projection spin operators have the eigenvalues:

$$\hat{s}^2|\psi\rangle = \hbar^2 s(s+1)|\psi\rangle \tag{1.12a}$$

$$\hat{s}_z|\psi\rangle = \hbar m_s|\psi\rangle \tag{1.12b}$$

where s and m_s have been used as magnitude, and projection *spin* quantum numbers. Note that the operator $\hat{\mathbf{s}}$ has been used for the spin, while $\hat{\boldsymbol{j}}$ was used to indicate angular momentum in general.

Even though spin is a quantum mechanical phenomenon, the spin of a particle also gives rise to a magnetic moment.[j] The spin-induced magnetic moment $\hat{\boldsymbol{\mu}}$ of a particle is related to its spin by an equation analogous to 1.9.[k] The observed magnetic properties of spin-possessing particles do not, however, match those expected from a classical definition of γ. The spin gyromagnetic ratio differs by a so-called g-factor. An electron gyromagnetic ratio is:

$$\gamma_\mathrm{e} = -g_\mathrm{e}\frac{e}{2m_\mathrm{e}} = -\frac{g_\mathrm{e}\mu_\mathrm{B}}{\hbar} \tag{1.13}$$

where μ_B is called the Bohr magneton – a quantity often used as a 'unit' of magnetic moment $\mu_\mathrm{B} = \frac{e\hbar}{2m_\mathrm{e}}$. The g-factor of a free electron is $g_\mathrm{e} \approx 2.00232$.

From now on we will, in fact, consider mainly the electrons, as they are at the core of the magnetosensitivity of chemical reactions. Electrons are spin $s = {}^1/_2$ particles. Their spin has, therefore, two possible projections onto the quantisation axis: $m_s = +{}^1/_2$ and $m_s = -{}^1/_2$. These are commonly referred to as $|\alpha\rangle$ and $|\beta\rangle$ states respectively (see figure 1.1a), and they are energetically degenerate for an electron in absence of any other spins or interacting fields.[l]

Zeeman interaction

If an external magnetic field is applied to the free electron, it interacts with the electrons' magnetic moment. The energy of the states whose magnetic moments point with the field direction are lowered in energy, while those whose magnetic moments point against the field are correspondingly raised in energy (see figure 1.1b). Since the magnetic moment is tied to the spin by equation 1.9, the external magnetic field effectively interacts with the spin states of the electron. This is referred to as the Zeeman interaction. The corresponding (electron) Zeeman Hamiltonian $\hat{\mathrm{H}}_\mathrm{Z}$ can be derived from a classical expression for an energy of a magnetic dipole in a magnetic field, as a dot-product of the applied magnetic vector field $\mathbf{B}$, with the

[j] It is important to emphasise that spin-possessing particles do not actually spin, at least not in any *spatial* coordinate.

[k] More specifically, magnetic moment of a fundamental fermion (e.g. electron), compound fermion (e.g. ^{13}C or ^{31}P with spin $^1/_2$ and $^3/_2$ respectively), or a compound boson (e.g. ^{14}N or ^{2}H both with spin 1). This encapsulates most, if not all, particles relevant to chemistry. Photon is a boson with spin 1, but no magnetic moment.

[l] Degenerate states have the same eigenvalue under the Hamiltonian operator. They are therefore spectroscopically indistinguishable even though they still *are* distinct states.

magnetic moment vector operator:

$$\hat{H}_Z = -\mathbf{B}\cdot\hat{\boldsymbol{\mu}} = \frac{g\mu_B}{\hbar}\mathbf{B}\cdot\hat{\mathbf{s}} \tag{1.14}$$

where equation 1.13 has been used to express the magnetic moment of an electron in terms of its spin, and the multiplicative constants have been taken out of the vector dot product.

Equation 1.14 implies an isotropic interaction between **B** and $\hat{\mathbf{s}}$. This is indeed the case for a free electron, since the basis for the spin vector can be chosen arbitrarily. Any direction of **B** in the laboratory reference frame, can be chosen as the quantisation axis of $\hat{\mathbf{s}}$. Most commonly though, the (unpaired) electron is located in a singly occupied molecular orbital. There is, therefore, a coupling between the orbital angular momentum and the spin angular momentum of the electron (spin-orbit coupling). This coupling ties the total spin to the orientation of the molecule the electron is located in. It therefore matters how that molecular framework is located in the laboratory reference frame, and the interaction between **B** and $\hat{\mathbf{s}}$ can be anisotropic. This anisotropy is expressed as an interaction tensor. By convention, the g-value of the electron in question is the quantity that is thought of as the one being anisotropic. The **g**-tensor, therefore, encapsulates the spin-orbit coupling of the electron and the resulting anisotropy of the interaction between the external magnetic field and the spin. The Zeeman Hamiltonian can therefore be written as:

$$\hat{H}_Z = \frac{\mu_B}{\hbar}\mathbf{B}\cdot\mathbf{g}\cdot\hat{\mathbf{s}} \tag{1.15}$$

Since **B** and $\hat{\mathbf{s}}$ are vectors in $\mathbb{R}^3$, **g** is a 3x3 matrix. For unpaired electrons residing in fast tumbling molecules in solution, the **g**-tensor averages out, since the molecular framework rotates with equal probability though all of its orientations with respect to the external magnetic field.[m] The tumbling must be 'fast' in comparison to the timescale of the measurement. This is the case for most magneto-sensitive samples measured in this work. Equation 1.14 can in fact be used for most purposes with g interpreted as the *average*, isotropic g-value, whether because the **g**-tensor is isotropic, or because it has an average value due to molecular tumbling. The average g-factor for electron spin in most organic molecules varies only very slightly from the free electron g-factor due to a small degree of spin-orbit coupling. The effective g-value can, however, be vastly different from g_e for molecules containing heavy nuclei, since the strength of the spin-orbit coupling is proportional to the 4^{th} power of nuclear charge.

If the direction of the external magnetic field is taken as the quantisation axis, equation 1.14 can be further simplified to:

$$\hat{H}_Z = \frac{g\mu_B}{\hbar}\mathbf{B}\cdot\hat{\mathbf{s}} = \frac{g\mu_B}{\hbar}B_0\hat{s}_z \tag{1.16}$$

where B_0 is the magnitude of **B**. For a spin-½ electron with isotropic g-value, the eigenvalues of the Zeeman Hamiltonian can therefore be found by using equation 1.12b.

$$\hat{H}_Z|\psi\rangle = \frac{g\mu_B}{\hbar}B_0\hat{s}_z|\psi\rangle = g\mu_B B_0 m_s|\psi\rangle \tag{1.17}$$

The magnetic field lifts the degeneracy of the two electronic spin projection states α and β. The Zeeman energies of the two states are:

$$E_Z = \pm\frac{1}{2}g\mu_B B_0 \tag{1.18}$$

and the corresponding energy level splitting ΔE_Z is therefore:

$$\Delta E_Z = \left|E_\alpha - E_\beta\right| = g\mu_B B_0 \tag{1.19}$$

[m] Averaging of the molecular orientations also occurs when measuring the bulk average of powders – an ensemble of randomly aligned single crystals, and glasses – a frozen solution where the molecular motion has stopped, but has been halted in an amorphous random orientation rather than in a crystal. There, however, all orientations of the molecules with respect to the external magnetic field are not equally likely. The resulting orientational average spectrum is referred to as the Pake pattern.

The vector model representation of the $|\alpha\rangle$ and $|\beta\rangle$ states, as well as their Zeeman splitting energies are shown in figure 1.1. The splitting in energy between the states is directly proportional to the magnitude of the applied field B. Note that, since the gyromagnetic ratio of an electron γ_e is negative, the magnetic moment vector points in the exact opposite direction to the spin vector. This means that the $|\alpha\rangle$ state is destabilised by the magnetic field, while the $|\beta\rangle$ state is, correspondingly, stabilised.

Time dependence of an observable

So far the picture of spin and its interaction with the magnetic field has been purely time invariant. The mechanisms of magnetic sensitivity considered in this chapter rely, crucially, on the time evolution of the spin system in the presence of the magnetic field. It is, therefore, crucial to consider such time varying states. The full time-dependent Schrödinger equation which describes the time evolution of the wavefunction of the system is:

$$i\hbar\frac{\partial}{\partial t}\psi(x,t) = \hat{H}\psi(x,t) \tag{1.20}$$

where the time t and space x dependence of the state of the system ψ have been explicitly indicated. This is the Schrödinger formulation of quantum mechanics where the operators which represent the observables remain constant in time, and the state of the system ψ changes. This is in contrast to the Heisenberg formulation where the state of the system (and, by extension the density matrix) is time invariant, and it is the operators which represent the observables which evolve in time; and the Dirac formulation where both the states and the operators are allowed to vary in time. These are, however, just different representations, and the results derived therefrom are not affected by the choice between the two. We will adopt the Schrödinger formulation.

Let us consider the states of equation 1.20 which are separable into a product of two functions of space- and time-only $\psi(x,t) = \sigma(x)\,\theta(t)$. This can be substituted into equation 1.20. On the left-hand side the partial derivative is with respect to time only, and therefore operates only on $\theta(t)$. Similarly, since in the Schrödinger picture the operators are invariant in time, $\hat{H}$ operates only on the spatial component $\sigma(x)$. Equation 1.20 therefore becomes:

$$i\hbar\sigma(x)\frac{\partial\theta(t)}{\partial t} = \theta(t)\,\hat{H}\sigma(x) \tag{1.21}$$

Dividing both sides by $\theta(t)\sigma(x)$ gives:

$$\frac{i\hbar}{\theta(t)}\frac{\partial\theta(t)}{\partial t} = \frac{1}{\sigma(x)}\hat{H}\sigma(x) \tag{1.22}$$

Both sides of the equation must be equal to a constant E, since the left-hand side is a function of time- and the right is space-only. This allows the Equation 1.22 to be separated into:[n]

$$\frac{\partial\theta(t)}{\partial t} = -\frac{i}{\hbar}E\theta(t) \tag{1.23a}$$

$$\hat{H}\sigma(x) = E\,\sigma(x) \tag{1.23b}$$

The constant has been given the symbol E, since it represents the energy of the state. The solution to Equation 1.23a is $\theta(t) = \theta(0)\exp\left(-\frac{Et}{\hbar}\right)$ – an oscillation between the real and imaginary components of the wavefunction with frequency $E/\hbar$. Hence, the higher the energy of a state the faster it rotates between the two phases. Equation 1.23b is a better known, time-independent Schrödinger equation $\hat{H}\psi = E\psi$. this is an eigenvalue equation, and the spatial components of the overall wavefunction are the eigenstates

[n] The argument for the separation of variables proceeds as follows: Lets consider a general case where a function of time-only Θ is equal to a function of space-only Σ. At each point in time and space (t,x) the two functions will have some value v, such that $\Theta(t) = \Sigma(x) = v(t,x)$. As, say, time changes, $\Theta(t)$ and $v(t,x)$ can change, but $\Sigma(x)$ cannot. The equality above can, therefore, hold only if v is not a function of time and therefore $\Theta(t)$ is constant in time. Analogous argument can be made for the the spatial dimension and, therefore, $\Theta(t) = v$ and $\Sigma(x) = v$.

of the Hamiltonian operator. In general there is no reason to assume that a state of Equation 1.20 is separable. However, only separable states are stationary in time, and are therefore called *stationary states*. The probability density of stationary states (product $\psi^*\psi$) is constant. The dynamics of all quantum systems can be expressed in terms of stationary states and mixed states which evolve in time accordingly. This is encapsulated by the density matrix representation (see section 'Density matrix formalism').

When the state of the system changes, so do its observables. The time dependence of the expectation value of a Hermitian operator $\hat{Q}$ can be found as follows:

$$\frac{\partial}{\partial t}\langle\hat{Q}\rangle = \frac{\partial}{\partial t}\langle\psi|\hat{Q}|\psi\rangle = \frac{\partial}{\partial t}\int \psi^*\hat{Q}\psi \,\mathrm{d}x \tag{1.24}$$

Taking the derivative under the integral, using the product rule for differentiation and assuming time independence of $\hat{Q}$, gives:

$$\frac{\partial}{\partial t}\langle\hat{Q}\rangle = \int \frac{\partial\psi^*}{\partial t}\hat{Q}\psi \,\mathrm{d}x + \int \psi^*\hat{Q}\frac{\partial\psi}{\partial t} \,\mathrm{d}x \tag{1.25}$$

The time dependence of the state of the system ψ, is described by the time-dependent Schrödinger equation. Substituting Equation 1.20, and its conjugate transpose, into the above equation gives:

$$-i\hbar\frac{\partial}{\partial t}\langle\hat{Q}\rangle = \int \left(\hat{H}\psi\right)^\dagger \hat{Q}\psi \,\mathrm{d}x - \int \psi^*\hat{Q}\hat{H}\psi \,\mathrm{d}x \tag{1.26}$$

Note that the conjugate *transpose* (indicated by a †) has to be applied to the Hamiltonian operator as well as to the wavefunction of the system. Since the conjugate transpose distributes according to $(\hat{A}\hat{B})^\dagger = \hat{B}^\dagger\hat{A}^\dagger$, and since $\hat{H}$ is hermitian ($\hat{H}^\dagger = \hat{H}$), the above equation rearranges to:

$$\begin{aligned} -i\hbar\frac{\partial}{\partial t}\langle\hat{Q}\rangle &= \int \psi^*\hat{H}\hat{Q}\psi \,\mathrm{d}x - \int \psi^*\hat{Q}\hat{H}\psi \,\mathrm{d}x \\ &= \int \psi^*\left(\hat{H}\hat{Q} - \hat{Q}\hat{H}\right)\psi \,\mathrm{d}x \end{aligned} \tag{1.27}$$

The term in brackets is a commutator of the Hamiltonian with $\hat{Q}$, and the integral is its expectation value. The overall equation can be, therefore written as:

$$\frac{\partial\langle\hat{Q}\rangle}{\partial t} = \frac{i}{\hbar}\left\langle\left[\hat{H},\hat{Q}\right]\right\rangle \tag{1.28}$$

The change in time of the expectation value of an operator (under state ψ), is directly proportional to the expectation value of the *commutator* of that operator with the system Hamiltonian. This equation is referred to as the differential form of the Heisenberg equation, and it can be solved to obtain time evolution of (the expectation value of) any operator.[o] An obvious upshot of Equation 1.28 is that any operator which commutes with the Hamiltonian (i.e. the commutator is equal to 0) is time invariant. Note that the Hamiltonian operator, whose eigenvalue is the energy of the system, commutes with itself.[p] This means that the total energy of a system is invariant in time with respect to its Hamiltonian, i.e. it is a conserved quantity.

As an illustration; the total Hamiltonian $\hat{\mathrm{H}}_\mathrm{T}$ for a single spin in a magnetic field is, as shown in Equation 1.14 the (isotropic) Zeeman Hamiltonian $\hat{\mathrm{H}}_\mathrm{Z}$:

$$\hat{\mathrm{H}}_\mathrm{T} = \hat{\mathrm{H}}_\mathrm{Z} = \frac{g\mu_\mathrm{B}}{\hbar}\mathbf{B}_0\cdot\hat{\mathbf{s}} = \frac{g\mu_\mathrm{B}}{\hbar}B_0\hat{s}_z \tag{1.29}$$

[o] Note that this is *not* the Liouville-Von-Neumann equation (Equation 1.46) which, as opposed to describing the time evolution of the expectation value of an operator, describes the time evolution of the density of states.

[p] Every operator commutes with itself since the order of two identical operations does not matter.

where the *static* external magnetic field **B** has been given a subscript '0'. If the spin is in an eigenstate of the Hamiltonian, it will remain unchanged ($\langle \hat{s}_z \rangle$ is conserved). How can the spin state ever change? The answer is that the Hamiltonian has to change. A quantum of electromagnetic (EM) radiation could induce a transition between the $|\alpha\rangle$ and $|\beta\rangle$ states by adding an oscillating magnetic field $\mathbf{B}_1$ to the total Hamiltonian. For the EM field to induce a transition, the EM field must have a non-zero component along an axis perpendicular to the direction of $\mathbf{B}_0$. If the direction of $\mathbf{B}_0$ is taken as the z-axis, and the EM field is polarised along the x-axis, the total Hamiltonian becomes:

$$\begin{aligned} \hat{\mathrm{H}}_\mathrm{T} &= \hat{\mathrm{H}}_\mathrm{Z} + \hat{\mathrm{H}}_\mathrm{EM}(t) \\ &= \frac{g\mu_\mathrm{B}}{\hbar}(\mathbf{B}_0 \cdot \hat{\mathbf{s}} + \mathbf{B}_1(t) \cdot \hat{\mathbf{s}}) \\ &= \frac{g\mu_\mathrm{B}}{\hbar}(B_0 \hat{s}_z + B_1 \sin(\omega t + \theta)\hat{s}_x) \end{aligned} \tag{1.30}$$

where the EM field of magnitude B_1 is oscillating at frequency ω and arbitrary phase θ. The total energy of the system is still conserved under the *total* Hamiltonian, but not necessarily under either the Zeeman $\hat{\mathrm{H}}_\mathrm{Z}$, or the EM field $\hat{\mathrm{H}}_\mathrm{EM}$ components – if the transition between the α and β levels occurs, the EM field will either lose (absorption) or gain (emission) energy. Further, both the Zeeman and the EM field Hamiltonian commute with the total spin operator $\hat{\mathbf{s}}$. The total Hamiltonian must therefore also commute with the components of $\hat{\mathbf{s}}$. From Equation 1.28, it follows that the expectation value of the total spin does not change. This contrasts with the spin projection operator $\hat{s}_z$ which, in turn commutes only with the Zeeman Hamiltonian. It therefore does not have to be conserved with respect to the overall Hamiltonian, and it does indeed change upon a transition between α and β levels. Note that, albeit this is true in this picture of quantum mechanics, it is in fact missing the inclusion of other effects such as the spontaneous emission, and the spin relaxation (not to mention quantum field theory and electrodynamics).

This example is the basis of the transitions in electron paramagnetic resonance (EPR) spectroscopy – where the EM field $\mathbf{B}_1$ is provided by the microwaves, and the external magnetic field which splits the Zeeman levels is the main magnetic field of the EPR spectrometer $\mathbf{B}_0$. An analogous example could be created for optical absorption spectroscopy: an electron undergoing transitions between its eigenstates due to presence of a (resonant) electric field component of the EM radiation (light). This example is, however, inherently more complicated as the interaction of the electron with its molecular environment cannot be reduced to an average electric field. In EPR one *can* however often think of a single spin in a magnetic field, which is why it was chosen as the primary example.

Larmor precession

The discussion of the Zeeman interaction so far, has concentrated on the energy level splitting, and the z-component of the angular momentum. As established, there is an inherent uncertainty in the x and y components of the spin vector such that they can be thought of as being located somewhere on a conical surface around the z-axis. More conclusions *can* however be drawn about the behaviour of the expectation value of these projections. Equation 1.28 allows one to understand how the expectation values of $\hat{s}_x$ and $\hat{s}_y$ behave in time. Substituting the (isotropic) Zeeman Hamiltonian (Equation 1.16) and setting $\hat{Q} \rightarrow \hat{s}_x$ into Equation 1.28 yields:

$$\frac{\partial \langle \hat{s}_x \rangle}{\partial t} = \frac{ig\mu_\mathrm{B}}{\hbar^2} B_0 \langle [\hat{s}_z, \hat{s}_x] \rangle \tag{1.31}$$

The multiplicative constants have been taken outside of the commutator, as well as the expectation value integral. The commutator of $\hat{s}_z$ and $\hat{s}_x$ is known as it is part of the definition of angular momentum (see Equations 1.5). Equation 1.31 therefore becomes:

$$\frac{\partial \langle \hat{s}_x \rangle}{\partial t} = -\frac{g\mu_\mathrm{B}}{\hbar} B_0 \langle \hat{s}_y \rangle \tag{1.32}$$

The same procedure for $\hat{s}_y$ yields:

$$\frac{\partial\langle\hat{s}_y\rangle}{\partial t} = \frac{g\mu_B}{\hbar} B_0 \langle\hat{s}_x\rangle \tag{1.33}$$

Since $\hat{s}_z$ commutes with itself, is is conserved and therefore its expectation value does not change with time:

$$\frac{\partial\langle\hat{s}_z\rangle}{\partial t} = 0 \tag{1.34}$$

The solution to the system of differential equations defined by Equations 1.32, 1.33 and 1.34 is:

$$\langle\hat{s}_x\rangle = A\cos\left(\frac{g\mu_B}{\hbar} B_0 t + \theta\right) \qquad \langle\hat{s}_y\rangle = A\cos\left(\frac{g\mu_B}{\hbar} B_0 t + \theta + \frac{\pi}{2}\right) \qquad \langle\hat{s}_z\rangle = m_s \tag{1.35}$$

where A and θ are arbitrary constants to be determined by boundary conditions, and m_s is the z-projection of spin angular momentum. In order to determine both A and θ, and obtain a particular solution, two boundary conditions are required. In this case we are able to determine only A, as there is *only one* boundary condition – the length the spin vector is fixed as $\hbar\sqrt{s(s+1)}$. The value of A is therefore determined as $A = \sqrt{2}/2$, while θ remains unknown.

One can think of a spin vector as *represented* by a a vector of its expectations values. In analogy to Equation 1.1, one can talk about the expectation value of the spin vector $\langle\boldsymbol{s}\rangle$ – the vector of the expectation values of the projections the spin vector operator.

$$\langle\boldsymbol{s}\rangle = \boldsymbol{x}\langle\hat{s}_x\rangle + \boldsymbol{y}\langle\hat{s}_y\rangle + \boldsymbol{z}\langle\hat{s}_z\rangle \tag{1.36}$$

This is referred to as the *vector model* of spin, and it is a useful picture to think about elements of spin chemistry. The cones shown in Figure 1.1 have, in fact already made use of this model. The $\langle\boldsymbol{s}\rangle$ vector precesses clockwise around the z-axis with frequency ω_L of:

$$\omega_L = \frac{g\mu_B}{\hbar} B_0 \tag{1.37}$$

This is referred to as the Larmor precession, and is a direct result of the Heisenberg equation (Equation 1.28).

Note that the existence of only one boundary condition for the Equations 1.35 is consistent with the uncertainty associated with $\hat{s}_x$ and $\hat{s}_y$. Since θ is undetermined, the exact position (phase) of $\langle\boldsymbol{s}\rangle$ is not known, only its precession frequency and length.

Hyperfine interaction

Angular momenta can interact with one another. This was already mentioned in section 'Zeeman interaction' when talking about the g-anisotropy and its origin – spin-orbit coupling. The spin of the electrons in atoms and molecules interacts with the overall spin of the nuclei. This is referred to as the *hyperfine interaction*. The hyperfine Hamiltonian $\hat{H}_{HF}$ is, similarly to the Zeeman Hamiltonian:

$$\hat{H}_{HF} = \hat{\boldsymbol{i}} \cdot \boldsymbol{a} \cdot \hat{\mathbf{s}} \tag{1.38}$$

where $\boldsymbol{a}$ is the hyperfine interaction tensor, $\hat{\boldsymbol{i}}$ is the nuclear spin angular momentum vector operator and $\hat{\mathbf{s}}$ is the corresponding operator for the spin of the electron. The hyperfine interaction has two components: the dipolar interaction and the Fermi contact interaction.

The dipolar interaction is the interaction of the magnetic field associated with the nuclear spin with the magnetic moment of the electron and vice versa (akin to the mutual interaction of two bar magnets). The dipolar interaction depends on the relative orientation of the electronic and nuclear dipoles, and it is therefore anisotropic. The orientational average of the dipolar interaction is zero for fast tumbling molecules (e.g. in solution). The Fermi contact interaction arises due to the electron density penetrating the nucleus.

This is the case only for electrons occupying orbitals with some s character as these are the only ones with non-zero probability of existing at the nucleus.[q] The Fermi contact interaction is isotropic.

The isotropic hyperfine Hamiltonian can be written as:

$$\hat{\mathrm{H}}_{\mathrm{HF}} = a\,\hat{\boldsymbol{i}}\cdot\hat{\boldsymbol{s}} = a\left(\hat{i}_x\hat{s}_x + \hat{i}_y\hat{s}_y + \hat{i}_z\hat{s}_z\right) = a\,\hat{i}_z\hat{s}_z + \frac{a}{2}\left(\hat{i}_+\hat{s}_- - \hat{i}_-\hat{s}_+\right) \tag{1.39}$$

where a is the average, isotropic hyperfine coupling constant, and $\hat{s}_\pm/\hat{i}_\pm$ operators are the raising and lowering operators for the electron and nuclear spins respectively.[r] Lets consider an (isotropic) interaction of an electron with a single magnetic nucleus, in an external magnetic field. The total spin Hamiltonian $\hat{\mathrm{H}}_{\mathrm{T}}$ for this system consists of the Zeeman interactions of the electron and the nucleus with the external magnetic field, as well as the hyperfine Hamiltonian:

$$\begin{aligned}\hat{\mathrm{H}}_{\mathrm{T}} &= \hat{\mathrm{H}}_{\mathrm{Z,e}} &&+ \hat{\mathrm{H}}_{\mathrm{Z,n}} &&+ \hat{\mathrm{H}}_{\mathrm{HF}} \\ &= \frac{g_e\mu_{\mathrm{B}}}{\hbar}B_0\hat{s}_z &&+ \frac{g_n\mu_{\mathrm{N}}}{\hbar}B_0\hat{i}_z &&+ a\,\hat{i}_z\hat{s}_z + \frac{a}{2}\left(\hat{i}_+\hat{s}_- - \hat{i}_-\hat{s}_+\right)\end{aligned} \tag{1.40}$$

The Hamiltonian operator can be represented by a matrix $\mathbf{H}$ where the m,n$^{\mathrm{th}}$ element is the expectation value of $\hat{\mathrm{H}}$ between the m^{th} and n^{th} state, $\mathbf{H}_{m,n} = \langle m|\hat{\mathrm{H}}|n\rangle$. The Hamiltonian from Equation 1.40 can be seen in this form in Figure 1.2 in the product ($\alpha\beta$) basis. Note that all the elements corresponding to interactions of the spins with the external magnetic field lie on the main diagonal of $\mathbf{H}$, while the spin-spin interactions appear on the off-diagonal elements. All the entries of $\mathbf{H}$, as well as its eigenvalues can be calculated numerically as follows:

```
1  ge = 2.0023; gn = 5.585695; % Electron and proton g-values
2  h = 6.62607004e-34; % Planck's constant
3  muB = 9.27400968e-24; muN = 5.05078324e-27; % Joule per Tesla
4  a = 200e6; % Hertz, Hyperfine coupling constant
5  % Diagonal elements of the Hamiltonian
6  d1 = @(B) + ge*muB/2*B + gn*muN/2*B + a/4*h;
7  d2 = @(B) + ge*muB/2*B - gn*muN/2*B - a/4*h;
8  d3 = @(B) - ge*muB/2*B + gn*muN/2*B - a/4*h;
9  d4 = @(B) - ge*muB/2*B - gn*muN/2*B + a/4*h;
10 % Solve for a range of fields
11 B0 = linspace(0,300,1e2); % B0 in Gauss
12 for Bi = 1:numel(B0)
13     B = B0(Bi)./1e4; % Field in Tesla
14     H=[d1(B) 0     0     0      ;...
15        0     d2(B) h*a/2 0      ;...
16        0     h*a/2 d3(B) 0      ;...
17        0     0     0     d4(B) ];
18     E(Bi,:) = eig(H)/h/1e6; % Energy in MHz
19 end
20 % Field at which ab<->bb transitions occurs under 600MHz radiation field
21 Bt = interp1(abs(E(:,1)-E(:,3)),B0,600);
```

Listing 1.1: Diagonalisation of the Hamiltonian for a simple spin system. Line 21 demonstrates how one might find a magnetic field at which the system resonates with a specific EM frequency (in this case 600 MHz).

Note that, for the sake of units consistency with the rest of the chapter, this calculation has not been done in atomic units. In practise this is almost never done due to the orders of magnitude of the quantities involved and the associated round-off errors.

[q] There are certain cases where electron *does not* need to be in an orbital with a s character to have hyperfine interaction with the nucleus in question. For example, both anthracene or azulene anions can couple to the nucleus with so-called polarisation mechanism – perturbation of the orbitals by the solvent.[21] The hyperfine coupling strengths of such systems depends, therefore, on anything which modulated that interaction: temperature, concentration, the nature of the solvent.

[r] The definition of a ladder (raising or lowering) spin operator is $\hat{j}_\pm = \hat{j}_x \pm i\,\hat{j}_y$. The following relation, which has been used to rearrange Equation 1.39, therefore follows: $2\left(\hat{i}_x\hat{s}_x + \hat{i}_y\hat{s}_y\right) = \hat{i}_+\hat{s}_- + \hat{i}_-\hat{s}_+$.

$$\mathbf{H} = \begin{array}{c} \\ \langle\alpha,\alpha| \\ \langle\alpha,\beta| \\ \langle\beta,\alpha| \\ \langle\beta,\beta| \end{array} \begin{array}{c} \begin{array}{cccc} |\alpha,\alpha\rangle & |\alpha,\beta\rangle & |\beta,\alpha\rangle & |\beta,\beta\rangle \end{array} \\ \begin{bmatrix} B_e + B_n \\ +A \end{bmatrix} & 0 & 0 & 0 \\ 0 & \begin{bmatrix} B_e - B_n \\ -A \end{bmatrix} & 2A & 0 \\ 0 & 2A & \begin{bmatrix} -B_e + B_n \\ -A \end{bmatrix} & 0 \\ 0 & 0 & 0 & \begin{bmatrix} -B_e - B_n \\ +A \end{bmatrix} \end{array}$$

$$A = \frac{\pi}{2}\hbar a \qquad B_e = \frac{g_e \mu_B}{2} B_0 \qquad B_n = \frac{g_n \mu_N}{2} B_0$$

Figure 1.2: Hamiltonian for interaction of a single unpaired electron with a magnetic (spin ½) nucleus, in a presence of an external magnetic field. The state labels shown (α/β) represent the electronic, and nuclear spin states: $|e,n\rangle$. h is the Planck's constant.

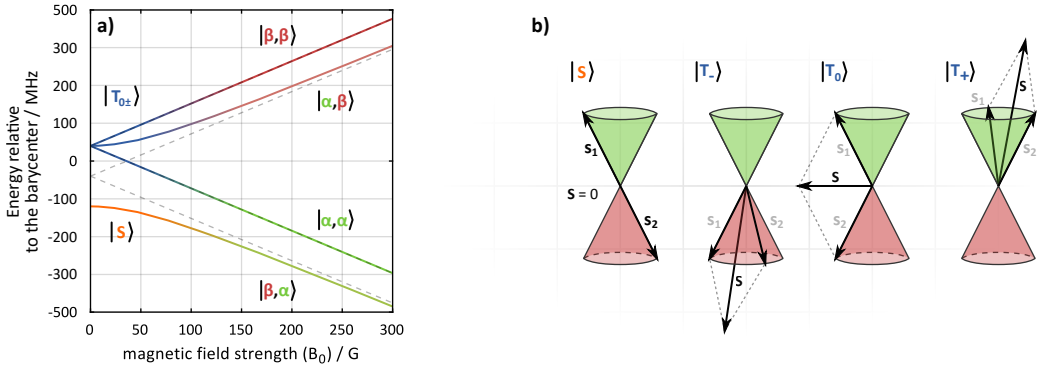


Figure 1.3: Zeeman interaction of the electron-nucleus spin system, and spin representation in product and coupled basis. **a)** Energy levels of the Hamiltonian from Figure 1.2, as a function of external magnetic field, as calculated in Listing 1.1. The α/β basis is a good basis set for high-field electron-nucleus interactions, since the Hamiltonian is almost completely diagonal (when expressed in these basis). The S/T basis is a good basis set for low- and zero-field electron-nucleus interactions, since the α and β levels couple strongly through the hyperfine interaction. The dashed line shows the effect of ignoring the off-diagonal elements of the Hamiltonian from from Figure 1.2. The hyperfine splitting at high field is still reproduced, but there is no correlation between the states at low fields, and hence no mixing.[t] **b)** Vector model representation of S and T states of a couple of spins. The upper (green) cone corresponds to the spin in the α state, and the lover (red) cone to the β one. $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ represent spins of two species. Note that the total spin vectors $\boldsymbol{S}$ of the T_+ and T_- states do not point along the quantisation axis, but also precess on their own respective conical surfaces (not shown).

The plot of the energy of each of the four states, as a function of applied magnetic field can be seen in Figure 1.3a. At high field, the four spin levels are split predominantly by the electron Zeeman interaction and the hyperfine interaction.[s] The nuclear Zeeman interaction is very weak and does not visibly perturb the energy levels at these field strengths. It has been included here for completeness. At low fields the diagonal elements of **H** become small, and the state of the system becomes governed by the off-diagonal elements – the hyperfine interaction. At zero field, the levels are split by the value of a into a set of one and three. The system can still be represented in the α/β basis, but they are no longer the eigenbasis of the low-field Hamiltonian – the Hamiltonian is not diagonal in this basis. The electron and nuclear spins are considered strongly coupled.

Spin coupling

[s] The hyperfine coupling constant chosen here (200 MHz) is large in order to show the effect of interaction of the spin levels. A typical hyperfine coupling to e.g. a proton (hydrogen atom) is ca. $78\,\mathrm{MHz} \times (R\,\text{Å})^{-3}$ where R is the distance between the nucleus and the electron.

[t] This approximation can therefore can be made for high field simulations of spin interactions – for example EPR spectra.

Coupled basis	Product basis	$\lvert S, M_S\rangle$
$\lvert S\rangle$	$\frac{1}{\sqrt{2}}\big(\lvert\alpha,\beta\rangle - \lvert\beta,\alpha\rangle\big)$	$\lvert 0,0\rangle$
$\lvert T_+\rangle$	$\lvert\alpha,\alpha\rangle$	$\lvert 1,+1\rangle$
$\lvert T_0\rangle$	$\frac{1}{\sqrt{2}}\big(\lvert\alpha,\beta\rangle + \lvert\beta,\alpha\rangle\big)$	$\lvert 1,0\rangle$
$\lvert T_-\rangle$	$\lvert\beta,\beta\rangle$	$\lvert 1,-1\rangle$

Figure 1.4: The representation of coupled spin basis vectors as linear combinations of product basis vectors. Note the conservation of information between different representations. In the uncoupled states there are 2 distinct quantum states of 2 spins – 4 combinations of $|\alpha\rangle$ and $|\beta\rangle$ states of each electron. Information ought to be neither lost, nor gained upon changing the mathematical representation of the system. Hence there are 4 states and 2 quantum numbers in both product and coupled basis.[v]

Strongly coupled spins and indeed, more generally, coupled angular momenta will evolve in time when viewed from the perspective of the α/β basis (referred to sometimes as the 'Zeeman basis').[u] A 'good' basis set (or even a good quantum number) is one which strongly resembles the true eigenbasis of the system, and therefore the system can be considered to be in a stationary state. A basis set for two strongly-coupled spins is one where the total spin of the pair, as opposed to the two individual spins is considered. This is referred to as the 'coupled basis'. The number of quantum numbers describing the system cannot change from basis to basis. As opposed to the projections spin quantum numbers of the two spins (${}_1m_s$ and ${}_2m_s$), the system is now described by the total spin angular momentum S, and *its* projection onto an arbitrary quantisation axis M_S. These quantum numbers, in analogy to Equations 1.12a and 1.12b, correspond to the spin operators operators:

$$\hat{S}^2|\psi\rangle = \hbar^2 S(S+1)|\psi\rangle \tag{1.41a}$$

$$\hat{S}_z|\psi\rangle = \hbar M_S|\psi\rangle \tag{1.41b}$$

The individual spin vectors add together to give the total spin according to the Clebsch-Gordan series:

$$\begin{aligned} S &= s_1+s_2,\ s_1+s_2-1,\ \ldots,\ |s_1-s_2|+1,\ |s_1-s_2| \\ M_S &= +S,\ +S-1,\ \ldots,\ -(S-1),\ -S \end{aligned} \tag{1.42}$$

S is the total spin (quantum number) of the pair, taking the values from the sum of the individual spin, in integer steps to their absolute difference – S is necessarily a positive quantity. For example, two spin-½ electrons can exist in one of four different quantum states – one *singlet* state (S) with no magnetic moment $S=0$ and three *triplet* states (T) with $S=1$ and $M_S \in [+1,0,-1]$. The ST basis are a linear combination of the uncoupled $\alpha\beta$ basis representation, as can be seen in Figure 1.4. The singlet and one of the triplet states (S_0 and T_0) have no overall magnetic moment projection onto the quantisation axis, and hence do not interact with the external magnetic field. The other two $T_\pm$ states have non-zero projection of the magnetic moment and can therefore interact with the external magnetic field. If the coupling between the two electrons is stronger than the magnitude of the external field applied, they can be considered to behave as if they were one spin-1 particle. This can be seen in Figure 1.9a where two electronic spins interact with one another much more strongly than with the applied magnetic field. The energy of the S_0 and T_0 states stays constant, while the T_+ and T_- states split in energy as B_0 increases. This is also the case in Figure 1.3a for fields below ca. 30 G.

[u] Bear in mind this is indeed only a matter of perspective. The system in question does not stop evolving in time. It is only the frame of reference which evolves in time with it. This is analogous to the way a rotational frame of reference makes the wooden horses on a merry-go-round appear stationary, and the rest of the world as spinning. The forces acting on are also perceived differently depending on a reference frame. The centrifugal force, for example, arises naturally in the rotational frame of reference due to conservation of linear momentum.

[v] There are, of course, 4 quantum numbers defining 2 spin particles – the projection *and* the total spin quantum number. It is just that for a set of particles with well-defined spin, the total spin quantum number does not change with representation and therefore has been omitted.

Two electrons paired in an atomic orbital interact very strongly due to their proximity and a shared spatial wavefunction. Paired electrons have no overall magnetic moment and will be of little interest here. The core of the mechanism of magnetosensitivity will be shown to rely on unpaired electrons (radicals). There are two ways in which two electronic spins in such radical pairs (RPs) can interact: dipolar interaction and the J-coupling. The dipolar interaction is the mutual interaction of the magnetic fields of the two spins (akin to two bar magnets). The strength of this interaction scales approximately as one over the cube of the distance between the electrons pair species – $D \propto r^{-3}$. Furthermore, the contributions of the dipolar interaction between the electrons average out in solution. J-coupling is a direct interaction of the wavefunctions of the two electrons. It arises due to Pauli exclusion principle which, by means of wavefunction symmetry, dictates the allowed spatial wavefunctions of different electronic states.[w] The strength of the J-coupling decreases approximately exponentially with distance – $J \propto \exp(-\beta r)$.

Density matrix formalism

The density matrix formalism is a very common tool in spin quantum mechanics. It allows one to describe and generalise not only large spin systems but also their ensembles.[x] Density matrices capture the idea of mixed states which allow for descriptions of spin states and dynamics involved in the magnetosensitive reactions. It also yields itself to a convenient description of the systems' time-evolution, and leads into the superoperator formalism which allows one to describe such phenomena as spin relaxation and reactivity. Density matrix formalism will be used in listing 1.6 to calculate the time evolution of a magnetosensitive spin ensemble.

For a system with n possible spin states,[y] the density *operator* $\hat{\rho}$ is defined as follows:

$$\hat{\rho} = \sum_{n,m} c_n c_m^* \, |\psi_n\rangle\langle\psi_m| \tag{1.43}$$

where p_n is the probability of the system being state $|\psi_n\rangle$.[z] The density *matrix* is a n-dimensional matrix $\boldsymbol{\rho}$, where the $\boldsymbol{\rho}_{m\,n} = \langle m|\hat{\rho}|n\rangle$, analogously to the way the Hamiltonian was expressed as a matrix in Figure 1.2.

Lets consider a state of a single spin-½ particle. This spin can exist in any linear combination of its eigenstates – $|\alpha\rangle$ and $|\beta\rangle$. The overall wavefunction can therefore be described as a vector of (complex) coefficients c of the two eigenstates:

$$|\psi\rangle = c_1|\alpha\rangle + c_2\big|\beta\big\rangle = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} \tag{1.44}$$

where the coefficients of the eigenstates have to obey a normalisation condition $|c_1|^2 + |c_2|^2 = 1$. Since the states of the system are describes as vectors, the density matrix can be written as the outer product of the state with itself:

$$\boldsymbol{\rho} = |\psi\rangle\langle\psi| = \begin{pmatrix} |c_1|^2 & c_1 c_2^* \\ c_1^* c_2 & |c_2|^2 \end{pmatrix} \tag{1.45}$$

[w] Pauli exclusion principle states that the total wavefunction of the system must be antisymmetric with respect to interchange of coordinates of identical fermions, and symmetric for exchange of bosons. The J-coupling is therefore sometimes called the exchange interaction.

[x] The wavefunction, even that of a stationary state, has a phase associated with it – the phase of the solution to Equation 1.23a. To obtain an average property of a system, one could simulate an entire ensemble of independent spin systems and thus calculate the average numerically. Bar even the prohibitive computational cost of such an approach, the phases of the wavefunctions in each member of an ensemble would have to be different (as they are, by definition, independent). The wavefunction, therefore, vanishes under the ensemble average. Complex conjugation of the wavefunction in the definition of the density matrix means the phase components cancel and therefore the density matrix survives the ensemble average and therefore be used to ascertain the properties of an average spin system.

[y] For N electrons and M nuclei, the dimensionality of the spin system is $n = 2^N \times \prod_{m=1}^{M} \{2I_m + 1\}$, where I_m is the total spin quantum number of the M^{th} nucleus.

[z] The density operator is sometimes written as: $\hat{\rho} = \sum_n p_n |\psi_n\rangle\langle\psi_n|$, where $p_n = |c_n|^2$. There is always a set of basis where this is true but, for generality, here it has been as a double sum over all possible states.

Where $*$ indicates a complex conjugate. The diagonal elements of the density matrix are the probabilities of the system existing in a certain state $p_n = |c_n|^2$, while off-diagonal elements are the coherences between different states. Note that the normalisation condition of the probabilities means that the density matrix ought to have a unit trace. It is, however, often the case (e.g. the Pauli spin matrices) that a multiple of an identity matrix is added to the density matrix scaled such that the trace of the result is zero. This turns out not to affect any dynamics of the spin system,[aa] but it improves its numerical and in some cases algebraic properties.[22]

The time evolution of the density matrix can be calculated in a way analogous to that which led to Equation 1.28 – by differentiating the density operator with respect to time and substituting in the time-dependent Schrödinger equation (Equation 1.20), to give:

$$\frac{\partial \hat{\rho}}{\partial t} = -\frac{i}{\hbar}\left[\hat{\mathrm{H}}, \hat{\rho}\right] \tag{1.46}$$

This is the Liouville-Von-Neumann equation. Note the difference of sign between it and the Heisenberg equation (Equation 1.28). The Liouville-Von-Neumann equation can also be written in an 'integrated' form – where the solution at time t is given explicitly as a function of the initial condition. To do this, the state vector $|\psi\rangle$ has to be Taylor expanded about time $t = 0$ to give:

$$|\psi(t)\rangle = \sum_{n=0}^{\infty} \frac{\partial^n |\psi\rangle}{\partial t^n}\bigg|_{t=0} \frac{t^n}{n!} = |\psi(0)\rangle + t\frac{\partial |\psi\rangle}{\partial t}\bigg|_{t=0} + \frac{t^2}{2}\frac{\partial^2 |\psi\rangle}{\partial t^2}\bigg|_{t=0} + \ldots \tag{1.47}$$

For a time-independent Hamiltonian, all the derivatives of the wavefunction can be found by repeated application of the time-dependent Schrödinger equation (matrix form of Equation 1.20) to give:

$$|\psi(t)\rangle = \sum_{n=0}^{\infty} \frac{(-i\mathbf{H}t)^n}{n!} |\psi(0)\rangle = \exp(-i\mathbf{H}t)\,|\psi(0)\rangle = \mathbf{U}|\psi(0)\rangle \tag{1.48}$$

where the infinite matrix power series is the definition of a matrix exponential in analogy to the Taylor expansion of $\exp(x)$.[ab] The state vector at time t has, therefore, been expressed as a function of the initial conditions, related to it by the propagator matrix $\mathbf{U} = \exp(-i\mathbf{H}t/\hbar)$. By definition of the density matrix (Equation 1.45), it follows that:

$$\boldsymbol{\rho}(t) = \mathbf{U}\boldsymbol{\rho}(0)\mathbf{U}^* \tag{1.49}$$

The expectation value of an observable can be expressed in terms of the density matrix by considering its matrix representation. When considering pure wavefunctions, the expectation value of an operator is the overlap integral of the result of the action of the operator in question on a wavefunction, with itself: $\int \psi^*(x)\hat{Q}\psi(x)\,\mathrm{d}x$ where x indicates all spatial dimensions. In *matrix formalism* operator of interest is a matrix $\mathbf{Q}$, $|\psi\rangle$ is a column, and $\langle\psi|$ is a row vector. The expectation value becomes a double sum over all rows and columns of $\mathbf{Q}$, as follows:

$$\langle\hat{Q}\rangle = \langle\psi|\mathbf{Q}|\psi\rangle = \sum_{i,j} \psi_i^* Q_{ij} \psi_j \tag{1.50}$$

where Q_{ij} is an ij^{th} element of the matrix representation of $\hat{Q}$, and ψ_i is the i^{th} element of the vector $|\psi\rangle$. Both indices run over all possible states of the system. Since Q_{ij} and ψ_j are just numbers, they commute under multiplication. This, by definition from Equation 1.45 yields a density matrix element under the sum.

[aa] Since the unit matrix commuted with everything, it commutes with the Hamiltonian and therefore does not contribute to the time evolution of the density matrix (by Equation 1.46). Since all the observables in spin dynamics are traceless matrices, this also does not affect their expectation values (by Equation 1.52).

[ab] This is a very computationally expensive operation.[23;24]

Furthermore, by definition of what is an operator in a matrix form, $Q_{ij} = Q^*_{ji}$. Hence:[ac]

$$\sum_{i,j} \psi^*_i Q_{ij} \psi_j = \sum_{i,j} \psi^*_i \psi_j Q_{ij} = \sum_{i,j} \rho_{ij} Q_{ij} = \sum_i \sum_j Q^*_{ji} \rho_{ij} = \sum_j \left\{ \mathbf{Q}^\dagger \boldsymbol{\rho} \right\}_{jj} \tag{1.51}$$

The last term is a trace of a product of the density matrix with the adjoint of $\hat{Q}$.[ad] Therefore, overall, the expectation value of an observable can be expressed as the trace of the product of it with the density matrix of the state.

$$\langle \hat{Q} \rangle = \mathrm{Tr}\left[\mathbf{Q}^\dagger \boldsymbol{\rho} \right] \tag{1.52}$$

By differentiating the above equation, the Heisenberg equation (Equation 1.28) can be obtained based on the density matrix formalism.[ae]

Finally, if the system is in a pure state with a particular value of an observable associated with an operator, then its density matrix is equal to the corresponding operator. This is sometimes referred to as the operator $\hookrightarrow$ state duality. Consider a spin-½ particle in a pure α sate. As per Equation 1.44 and 1.45, its density matrix is:

$$\boldsymbol{\rho}_\alpha = |\alpha\rangle\langle\alpha| = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \underset{-k\mathbf{I}}{\rightarrow} \frac{1}{2} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} = \boldsymbol{\sigma}_z \tag{1.53}$$

where σ_z is referred to as the *Pauli spin matrix* for the z(+) state, $\mathbf{I}$ is an identity matrix and k is a constant chosen such that the resultant density matrix is traceless. Note the difference of a multiplicative factor of $\hbar$ between σ_z and $\mathbf{s}_z$ – the matrix representation of the spin-z state. This difference vanishes when working in atomic units. In a similar way, by setting $|\psi\rangle$ to the x and y states, one can derive the Pauli matrices for the spin x (σ_x) and y (σ_y) states:

$$\boldsymbol{\rho}_x = |x\rangle\langle x| = \frac{1}{2} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \underset{-k\mathbf{I}}{\rightarrow} \frac{1}{2} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \boldsymbol{\sigma}_x \tag{1.54}$$

$$\boldsymbol{\rho}_y = |y\rangle\langle y| = \frac{1}{2} \begin{pmatrix} 1 \\ i \end{pmatrix} \begin{pmatrix} 1 & -i \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & -i \\ i & 1 \end{pmatrix} \underset{-k\mathbf{I}}{\rightarrow} \frac{1}{2} \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} = \boldsymbol{\sigma}_y \tag{1.55}$$

Pauli spin matrices, alongside the operator $\hookrightarrow$ state duality property, have been used in Listing A.30 to construct the initial state, and the observable operator for the simulation of a magnetosensitive spin evolution.

[ac] The following property of matrix products has been used: $\{\mathbf{AB}\}_{ij} = \sum_k A_{ik} B_{kj}$. The ij^{th} element of a matrix product is the sum over the product of k elements of the i^{th} row of A and j^{th} column of B. In this case this relation is used backwards to collapse the sum over the middle indices.

[ad] The adjoint is not important for most observables since most of them are self-adjoint (Hermitian). An example of a non-Hermitian observable is the free induction decay quadrature signal – proportional to $\hat{s}_x + i\hat{s}_y$.

[ae] In order to do this, one has to use the property of a cyclic permutation under the trace: $\mathrm{Tr}[\mathbf{ABC}] = \mathrm{Tr}[\mathbf{BCA}] = \mathrm{Tr}[\mathbf{CAB}]$, where **A**,**B** and **C** are square matrices of the same size.

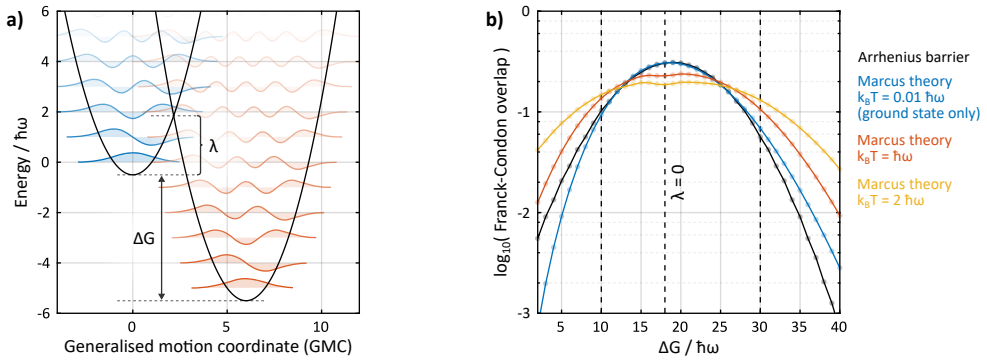


Figure 1.5: a) Electron transfer reactant (blue) and product (red) potentials and wavefunctions, harmonic along the generalised motion coordinate (GMC). The energy barrier λ and Gibbs free energy ΔG of the ET reaction are in indicated. The products are shifted 6 units along the GMC. **b)** Franck-Condon overlap of the normalised harmonic potential wavefunctions as a function of ΔG. The average overlap of the Boltzmann distribution of the states for different temperatures has been shown. The ground-state only overlap has been calculated according to Listing 1.2. The Arrhenius-like barrier has the form $\exp(-\lambda)$. Three points for which the overlap of the wavefunctions has been shown in more detail in Figure 1.6 are indicated by the dashed lines, and correspond to $\Delta G \in [10, 18, 30]\,\hbar\omega$.

1.2 Electron transfer reactions

The long-range transfer of electrons between different chemical species is fundamental for magnetosensitive reactions under investigation in this work. Electron transfer (ET) can be responsible for the formation of the magnetosensitive pairs of radicals as well as their quenching. Long-range ET is described very successfully by Marcus theory.

Time-dependent perturbation theory can be used to derive the Fermi golden rule for the electron transfer. This is done by considering an oscillating perturbation to an otherwise steady-state Hamiltonian (see e.g. Equation 1.30). If $\hat{\mathrm{H}}'$ is the time-dependent perturbation describing the electron transfer from an initial $|i\rangle$ to a final $|f\rangle$ state, the rate of transfer of an electron between the two states is proportional to the matrix element of the perturbation Hamiltonian:

$$k_{\mathrm{ET}} \propto |\langle f|\hat{\mathrm{H}}'|i\rangle|^2 \tag{1.56}$$

The motion of the electrons and the nuclei can be treated separately (Born-Oppenheimer approximation). The state wavefunctions can therefore be separated into a product of vibrational $|\upsilon\rangle$, electronic $|\xi\rangle$, and spin $|\sigma\rangle$ wavefunctions. Since the $\hat{\mathrm{H}}'$ couples only the electronic states, the electron transfer rate becomes:

$$k_{\mathrm{ET}} \propto |\langle \xi_{\mathrm{f}}|\hat{\mathrm{H}}'|\xi_{\mathrm{i}}\rangle|^2 \, |\langle \upsilon_{\mathrm{f}}|\upsilon_{\mathrm{i}}\rangle|^2 \, |\langle \sigma_{\mathrm{f}}|\sigma_{\mathrm{i}}\rangle|^2 \tag{1.57}$$

$\langle \xi_{\mathrm{f}}|\hat{\mathrm{H}}'|\xi_{\mathrm{i}}\rangle$ is the matrix element of the time-dependent interaction coupling $|i\rangle$ and $|f\rangle$, and $\langle \upsilon_{\mathrm{f}}|\upsilon_{\mathrm{i}}\rangle$ is the Franck-Condon factor – the integrated overlap of $|i\rangle$ and $|f\rangle$ vibrational wavefunctions.

The $\langle \sigma_{\mathrm{f}}|\sigma_{\mathrm{i}}\rangle$ in Equation 1.57 can be considered to be the spin selection rule. If the overlap integral of the final and the initial spin components of the wavefunctions is zero, so is the transition rate between these states. The overlap integral can be shown to be zero for states with different spin multiplicity (e.g. singlet and triplet), and any transitions between such states (e.g. phosphorescence) are very slow. The reason such transitions are possible in the first place, is because the assumption that $|\Psi\rangle$ can be expressed as a product of electronic-only, and vibrational-only wavefunctions is not quite true. There is always a very small degree of coupling between the vibrational and electronic components of the overall wavefunction $|\Psi\rangle$, and hence the overlap integral $\langle f|\hat{\mathrm{H}}'|i\rangle$ for such transitions is not actually zero.

Marcus theory of electron transfer builds on top of the Born-Oppenheimer approximation for the rate of electron transfer . The Marcus-like electron transfer is characterised as follows:

- Born-Oppenheimer approximation leading to Equation 1.57.
- ET reactions conserve spin (i.e. $\langle\sigma_f|\sigma_i\rangle$ is equal to 1 if the spin state does not change, and 0 otherwise)
- The interaction between the electronic orbitals falls off approximately exponentially with distance. $|\langle\xi_f|\hat{H}'|\xi_i\rangle|^2 \approx H_0^2 \exp(-\beta r)$ where H^2 is maximum electronic coupling (at close distance), r is the distance between the ET loci and β is a positive constant. This assumption is broadly true as the magnitude of the electronic wavefunctions decays approximately exponentially with distance from the nucleus. The value of β is approximately constant for a biological medium.[25]
- The vibrational motion of both species involved can be expressed as perfectly harmonic along a certain, generalised motion coordinate (GMC). This may be a combination of internal vibrations and rotations.[af] The characteristic oscillator frequency for motion along GMC is approximately constant for biological medium.[25]
- The reaction takes place at constant pressure and temperature. This means that Gibbs free energy is a good measure of the capacity of the chemical system to do work.

The implications of Marcus theory are, therefore concerning mainly the Franck-Condon overlap integral, and its approximation by a harmonic oscillator along the GMC. A general view of the overlap can be seen in Figure 1.5a. The product potential is displaced with respect to the reactant both along the GMC and the energy coordinate. The former reflects different molecular arrangement of the species after ET, and the latter the ET being thermodynamically favourable. The relative position of the two potential parabolas can be parametrised in many ways, but within the scope of Marcus theory this is done with: the Gibbs free energy difference ΔG between them, and the potential energy barrier height λ when travelling from reactants to products. k_{ET} is predicted to increase as the products get stabilised with respect to reactants since λ decreases. k_{ET} is predicted to peak when $\lambda = 0$. If the ET products become further stabilised with respect to the reactants, the rate of ET is predicted to begin dropping again as λ begins to increase again (The point where two quadratic potentials cross is now on the left-hand side of the product potential). The region of ΔG values where k_{ET} increases with increasing ΔG is referred to as the *normal* Marcus region. The region where the opposite is true is referred to as the *inverted* Marcus region.

If the λ barrier were to be treated in a purely Arrhenius fashion, the rate of the electron transfer is proportional to the exponential of the (negative) barrier height: $k_{ET} \propto \exp(-\lambda)$. Marcus' argument about reaction rates is, however, not only based on the height of the barrier, but the associated trend in the Franck-Condon overlap of the wavefunctions involved. 1D wavefunctions for a species in a harmonic potential are calculated numerically in Section 'Boundary value problems'. These results can be used here to calculate the Franck-Condon overlap for different values of ΔG and hence show the behaviour of ET rates as predicted by Marcus.

The method and code as shown in Listing 2.45 has been used to calculate the first 50 eigenstates of a particle in a harmonic well. The domain size used was $L = 15$ units along the GMC, and the exponential runaway truncation was used as described in Section 'Boundary value problems'. The characteristic frequency of the oscillator used was $\hbar\omega = 0.5\,E_h$. The following code relies on existence of the matrix of eigenvectors `Psi` and eigenvalues `E` for potential function `V(x)`. The Franck-Condon overlap for each of the eigenstates of the product with the ground eigenstate of the reactant can be calculated as follows:

```
Psi2 = interp1(xx+6,Psi,xx,'linear',0); % Product eigenstates
F = abs(sum(bsxfun(@times,Psi(:,1),Psi2))*dx);
```

Listing 1.2: Franck-Condon overlap integral of harmonic oscillator wavefunctions calcualted by direct multiplication of the base (`Psi`) and the shifted and interpolated wavefuncitons (`Psi2`).

[af] A classical example is to consider the expansion and contraction of the solvent shell around a single ionic species. The rates of oxidation/reduction follow the trend predicted by Marcus theory, since the electron transfer happens much faster than solvent relaxation as the central ion becomes smaller/larger respectively.

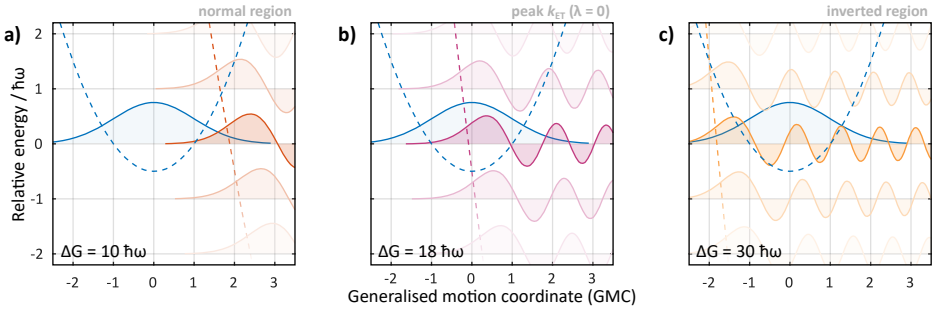


Figure 1.6: Details of the Franck-Condon overlap for selected ΔG values indicated in Figure 1.5. a) Marcus normal region. ET rate k_{ET} increases with increasing ΔG. b) Maximum Franck-Condon overlap , peak k_{ET} and no barrier for the reaction ($\lambda = 0$). c) Marcus inverted region. ET rate k_{ET} decreases with increasing ΔG since the Franck-Condon overlap decreases.

The quadratic potential for the products has been shifted, arbitrarily, 6 units along the GMC axis. The resulting Franck-Condon overlap as a function of ΔG, as well as the Arrhenius treatment of λ has been plotted in Figure 1.5b. The normal as well as the inverted region can be seen to be predicted in the same way by both approaches. Figure 1.6 shows the overlap between the ground-state reactant and higher vibrational product wavefunctions for three points: normal region, peak k_{ET} and inverted region.

This treatment assumes that the entirety of the population of the reactants is present only in the ground generalised vibrational level. This is very often true, but Markus theory holds regardless of this assumption. The calculation of average Franck-Condon overlap for Boltzmann-like distribution through the higher generalised vibrational levels has also been shown in Figure 1.5b. The normal/inverted region pattern can be see to also hold for such thermally-excited reactants. Note that the slope of k_{ET} is, in general, much less steep in the inverted region.[ag]

Markus' theory, albeit simple in its assumptions, has been successful at predicting the rates of biological ET reactions[25;26] – both the normal and the inverted region. Markus normal region often governs the intermolecular electron transfer and, by extension, energy transfer reactions. The inverted region behaviour is seen very commonly in intramolecular ET in proteins, where the electron transfer between neighbouring amino acid residues[ah] can have a very small reorganisational change (horizontal shift of the products potential along the GMC) accompanied by a very large ΔG. Evolution is thought to utilise the inverted region to tune the rates of ET in such proteins. Since the values of β and $\hbar\omega$ are approximately constant for biological systems, the rate of ET is reasonably parametrised by three parameters: ΔG, λ and r, all of which have a well-defined influence on the ET rates. Markus' theory, therefore, allows one to make predictions and rationalisations about the ET reactions which initiate the mechanisms of magnetosensitivity of biological systems.

[ag] To rationalise this consider the residual overlap between the eigenstates in Figure 1.6c, not present in Figure 1.6a.

[ah] Only a few naturally-occurring amino acids can stabilise a radical efficiently, and therefore be involved in electron transfer reactions. These are Tryptophan (W) and Tyrosine (Y), both with hydrophobic side chains, as well as Histidine (H) with a polar side chain. All three of these amino acids are aromatic. Cysteine (C) and Methionine (M) have also been known to stabilise radical species due to the presence of heavier atom (sulphur). Biological ET can also be mediated by appropriate cofactors, notably e.g. iron-sulphur clusters.

1.3 Reaction kinetics

The quantification of the rates of chemical reactions, as well as the interplay between multiple interacting processes[ai] is an important concept towards understanding the way the magnetic field effects can manifest themselves in the bulk properties of chemical reactions.

Arguably the simplest example of a chemical reaction is a unimolecular first-order reaction. First-order reactions are those whose rate is governed by the abundance of only one species. A unimolecular first-order reaction therefore has the form:

$$\mathrm{A} \xrightarrow{k} \mathrm{B} \tag{1.58}$$

Species A is being transformed into species B with rate quantified by k – the rate constant. The concentrations of species A and B are subject then to a differential equations:[aj]

$$\frac{\partial[\mathrm{A}]}{\partial t} = -k[A] \tag{1.59a}$$

$$\frac{\partial[\mathrm{B}]}{\partial t} = k[\mathrm{A}] \tag{1.59b}$$

whose analytical solutions are simply the exponential decay of A from its initial concentration $[A]_0$, and corresponding growth of the concentrations of B:

$$[\mathrm{A}] = [\mathrm{A}]_0\, e^{-tk} \tag{1.60a}$$

$$[\mathrm{B}] = [\mathrm{A}]_0 \left(1 - e^{-tk}\right) \tag{1.60b}$$

The units of the rate constant k for a first order process are therefore s^{-1}.

Numerically, this can be simulated with an differential equation solver in Matlab as follows:

```
y0 = [1 0]'; % Initially only A
k = 1; % Per seccond
K = k*[-1 0; 1 0]; % Rate constant matrix
f = @(t,y) K*y; % A->B
[t,y] = ode15s(f,[0 10],y0);
```

Listing 1.3: Simple A→B kinetics simulation.

Note the use of a stiff solver (`ode15s`). This is not strictly necessary for this specific problem, but is a good practice for simulation of chemical kinetics, as most chemical systems rely on both large and small relative concentrations of species and are therefore considered stiff problems (for more information see Section 2.3).[ak]

The (first-order) rate of a reaction is sometimes given in units of time, as opposed to reciprocal time. This stems from the convention where the rate constant itself is given as a reciprocal such that e.g. $k = 10^9$‘, s^{-1} will be reported as $\tau = k^{-1} = 1\,\mathrm{ns}$. An exponential with a (reciprocal) time constant of 1 ns will decay by $\left(1 - e^{-1}\right) \approx 63.2\,\%$ in 1 ns. τ is, therefore referred to as the timescale/time constant of the reaction.[al]

A reaction can behave like a first-order one even when there is more than one reactant. When the concentration of all the other reactants can be considered constant, their concentrations can be incorporated into an effective first-order k for the one species which undergoes change. This happens when one of the reactants is in vast excess and it is therefore, effectively, always available for the reaction.[am] Another way a pseudo-first-order reaction can occur, is if one of the reactants rapidly undergoes a rapid backwards

[ai] This part of the chemical kinetics has strong relations to graph-theory, with chemical species being the vertices, and the reactions being the edges of the graph.

[aj] This formulation is appropriate for bulk kinetics. An equivalent set of differential equations can be formulated for the probability of a particle being in state A or B at time t. Such a formulation is more appropriate for considering a stochastic reaction of a single species. For such a stochastic process it can be shown, for example, that the time at which the reaction occurs is $t_r = k^{-1}\ln(r)$ where r is a random number drawn from a uniform distribution on a unit interval.[27]

[ak] Consider, for example, a reaction where A→B→C with $k_{\mathrm{BC}} \gg k_{\mathrm{AB}}$. B is a transient intermediate whose concentration will be very small at all points through the reaction, but it nevertheless has to be simulated correctly, as it is the only pathway from A to C.

[al] As a rough guideline, for a single exponential decay, 1 time constant τ corresponds to a decay by ca. 63.2%, $3\tau \to 95\%$, $5\tau \to 99\%$ and $7\tau \to 99.9\%$.

[am] Note the implication of kinetic *availability*, rather than just an *excess*.

reaction, in comparison to the forward reaction. To illustrate, consider the following reaction, where A gets oxidised by B:

$$A + B \xrightarrow{k_{AB}} A^{\bullet+} + B^{\bullet-} \qquad B^{\bullet-} \xrightarrow{k_{B'}} B \tag{1.61}$$

These reactions can be better seen on a reaction scheme as shown in the insert of Figure 1.7a. The rate of change of [A] can be considered as, approximately, constant if $k_{B'} \gg k_{AB}$, and hence B gets reoxidised much faster than it oxidises A. Furthermore, the reaction of reoxidation of B is, too, pseudo-first-order as the reoxidant of B is considered to be in excess and hence not explicitly written. This is common when, like for example in this case, the exact rate is not critical. The only important thing about the reoxidation of $B^{\bullet-}$ to B is that is is much faster than reaction of A with B. The behaviour of the above system can, too, be simulated with a differential equation solver. The formulation of the chemical system as written in 1.61 is:

```
y0 = [1 1 0 0]'; % Initially only A and B
kAB = 1; kBp = 100; % Rate constnats
f = @(t,y) kAB*[-1 -1 1 1]'*y(1)*y(2) + [0 1 0 -1]'*y(4)*kBp; % ODE fun.
[t,y] = ode15s(f,[0 1e2],y0); % Solve
```

Listing 1.4: Electron transfer with reoxidation simulation.

Equal parts of A and B are the only species initially present. The bimolecular rate constant for the redox of A and B is set to $k_{AB} = 1\,\mathrm{s}^{-1}\mathrm{conc.}^{-1}$ (note the change of units since this is a bimolecular rate constant). With $k_{B'} = 100\,\mathrm{s}^{-1}$ the oxidation of A appears first-order with rate constant $k_A = k_{AB} \times 1\,\mathrm{conc.} = 1\,\mathrm{s}^{-1}$, and [A] falls exponentially in time. As k_{AB} becomes smaller and, eventually, approaches 0, the rate of decay of [A] decreases and approaches $(1 + k_A t)^{-1}$. This behaviour has been shown in Figure 1.7a.

Note that the system of differential equations given here is non-linear and therefore, as opposed to the first-order kinetics equation in Listing 1.3 or 1.5, it cannot be expressed as a (pure) matrix equation.[an]

The example above involves a cycle: $B \to B' \to B$ (see Figure 1.7a). It is such cyclical elements of chemical kinetics which lead to the more complicated behaviours of chemical systems. The reaction 1.61 does, however consist of a sink (species A'), and hence its terminal behaviour is for all the species to end up in states A′ and B. A chemical reaction proceeding entirely in a cycle (with no sources or sinks) has to involve a driving force, [ao] and hence is often depicted with the ordinate broadly representing the Gibbs free energy of the species involved. This driving force is often the energy delivered by a photon. This is the case in this work, and hence such circular kinetics will be referred to as a *photocycle*.

An example of a photocycle is:

$$A \xrightarrow{k_{AB}} B \qquad B \xrightarrow{k_{BCD}} \tfrac{1}{2}C + \tfrac{1}{2}D \qquad C \xrightarrow{k_{CA}} A \qquad D \xrightarrow{k_{DA}} A \tag{1.62}$$

The reaction $A \to B$ can be considered as the photoexcitation. $B \to \frac{1}{2}C + \frac{1}{2}D$ can be thought of as the dissociation of B, and finally $C \to A$ and $D \to A$ are species C and D returning to the ground state. The depiction of this photocycle can be seen in the insert of Figure 1.7b. Importantly, the states A (and, possibly B) are not 'pure states' – consisting one chemical species, but instead they are *compound states* – cumulative concentration of multiple species considered as the 'ground state'. Hence, too, the factor of ½ in the second Equation in 1.62 – to conserve molecularity (number of molecules present). The use of compound states

[an] This equation could be linearised by augmenting the y vector by a term p corresponding to the product of concentrations of A and B, $p = [A][B]$. p can then be propagated in time by the product rule $\dot{p} = [\dot{A}][B] + [A][\dot{B}]$. The system of differential equations becomes linear upon introduction of p because the rate of change of each of the constituent species can be expressed as a linear combination of their instantaneous concentrations. The operation of augmenting y is still, however, non-linear and has to be performed at each iteration.

[ao] The driving force has to be an external source of Gibbs free energy flowing through the system, as otherwise there would be no way of completing the cycle, as that requires at some point going up the energetic hill. In a closed system the Gibbs free energy of a reaction (or otherwise, an appropriate thermodynamical formulation of the systems energy) has to decrease. A photocycle has to therefore be an energetically open system, or violate the second law of thermodynamics upon which the postulate of free energy is built. An identical argument prevents the existence of perpetual motion machines – perpetuum mobile.

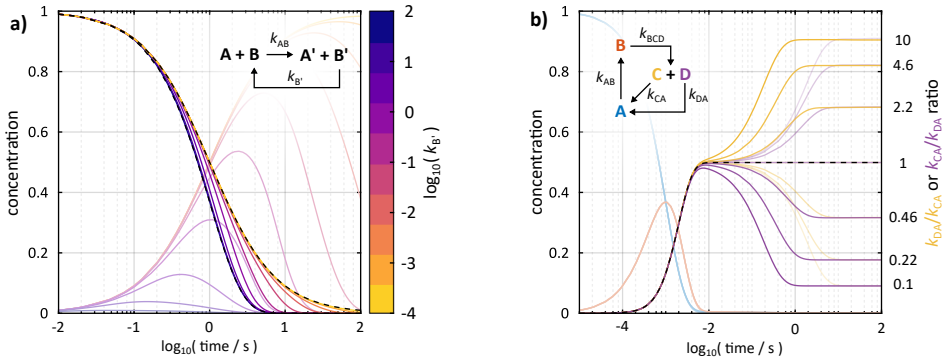


Figure 1.7: a) Reaction kinetics of an electron transfer reaction k_{AB} with a variable rate reoxidation step $k_{B'}$. Solid lines correspond to the decaying concentration of A. The faded lines in the background correspond to the transient concentration of B′. In the limit of $k_{B'} \to 0$, this reaction approaches the simplest possible bimolecular reaction $A + B \to C$. The photocycle is shown in the insert. **b)** Steady-state behaviour of a simple photocycle (shown in the insert). The solutions are plotted for varied ratio k_{CA}/k_{DA}. This demonstrates that the steady state concentrations of these species can be affected very strongly by small changes in the rate constants governing the long-timescale accumulation.[ap]

simplifies the equations for chemical kinetics, both due to a smaller number of species involved, as well as by, often, linearising the resulting system of differential equations (as it is in this case)

This photocycle can be simulated as follows:

```
y0 = [1 0 0 0]'; % Initially only A
kAB = 1e3; kBCD = 1e3; % Fast rate constants
kCA = 1; kDA = kCA/10; % Rate of decay of C and D back to A
K = [-kAB   0      +kCA +kDA ; % Matrix of rate constants
     +kAB  -kBCD    0    0   ;
      0    +kBCD/2 -kCA  0   ;
      0    +kBCD/2  0   -kDA];
f = @(t,y) K*y; % ODE function
[t,y] = ode15s(f,[0 1e2],y0); % Solve
```

Listing 1.5: Photocycle kinetics simulation

Initially all of the species are in the state A. The rates of k_{AB} and k_{BCD} are be kept constant and high, while the rates k_{CA} and k_{DA} are kept low. The plot of y in time can be seen in Figure 1.7b. When the time starts [A] drops as it gets converted to B, which in turn quickly disappears to give equal proportions of C and D. This establishment of pseudo-steady state can be considered as the fast part of the photocycle since it happens on the timescale of, approximately, $k_{AB}^{-1} + k_{BCD}^{-1}$. The slow part of the photocycle is, then, the establishment of the overall *steady state*. This happens on the timescale of the slower of the two reactions, C → A and D → A – $\min(k_{CA}, k_{DA})^{-1}$. Importantly, since there are no unbalanced sinks nor sources, the steady-state concentrations of all the species are non-trivial and depend strongly on the exact values of the rate constants involved, especially the lowest ones (aka the slowest reactions). To illustrate, the approach of the steady state of the chemical system above has been simulated for different ratios k_{DA}/k_{CA} – from 1/10 to 10. This can be seen in Figure 1.7b. The slower the overall rate of depletion of D, the larger its steady-state proportion.

The behaviour of the reaction rates can be very complex in itself. Moreover, factors like sample, diffusion and photodegradation which add slow, unbalanced sources and sinks respectively to the photocycle can and do complicate further detailed understanding of slow kinetics of photosystems. The detailed behaviours of the reactions will therefore not be analysed in this work, and instead the overall trends will give suggestion as to the underlying mechanisms. Since the single-exponential behaviour is a pseudo-steady-state limit of first-order reactions, it will be used for fitting of kinetics in future chapters as a general guideline rather than an accurate model of the photochemistry involved.

[ap] This can be thought of as large magnitude of the Jacobian of the steady-state concentration with respect to $\boldsymbol{k}$ – the vector of rate constants.

1.4 Magnetosensitivity of chemical reactions

The mechanism of magnetosensitivity of chemical reactions discussed here will be the *radical pair mechanism* (RPM) – a mechanism based on the interactions of a pair of radical species with the magnetic field. The formation of the RP, the evolution of its spin state and its eventual fate will be shown to all play a role in developing the magnetic field effect (MFE) – a macroscopic, measurable change in the state of a chemical system.

Formation of the radical pair

Radical pairs can originate from various chemical or photochemical pathways. Radical pairs are created in a spin-polarised state – thermodynamic non-equilibrium. The first step in their formation must, therefore, be the absorption of energy. In all cases considered here, the source of energy will be an absorption of a photon by the ground-state chromophore, to form a first excited electronic state. The photoexcited species can be either an electron donor or an acceptor: it can accept electron to the lower lying singly occupied molecular orbital (SOMO) as well as donate one from the higher energy one (SOMO*). The resulting pair of species (oxidised/reduced chromophore and and electron donor/acceptor) is formed with a conservation of spin angular momentum – in the same overall spin configuration (S or T) as the parent species.

In order for the electron transfer to occur, the molecular precursors of the radical pair must be in close proximity – there must be either an orbital overlap or electron transfer through the medium. This means that such RP will form in a strongly coupled state. In cryptochrome (Cry) proteins discussed in Chapter 6, the electron transfer happens between non-covalently bound flavin adenine dinucleotide (FAD) cofactor and a chain of tryptophan (W) residues. The RP is therefore created very rapidly from the first excited singlet state of FAD (singlet-born). In solution based magnetosensitive systems such as flavin mononucleotide/hen-egg white lysozyme (Chapter 4) the photoexcited molecular precursors are formed in solution and usually have plenty of time to evolve before meeting their ET partner. This allows the intersystem crossing (ISC) to compete with the ET and the RP in such systems usually originate from molecular triplet (triplet-born).[aq] The pyrene/1,3-dicyanobenzene RP originate from a single precursor (an exciplex) and therefore the RP is singlet-born. The details of the magnetosensitivity of this system will be discussed in Chapter 3 (Figure 3.6).

RP are formed in close contact, in a strongly coupled state, no spin-state mixing and therefore field-coherent spin evolution can take place. This can be seen by simulating an interaction of a pair of strongly coupled electrons in a magnetic field. This can be done by small alterations to the code in Listing 1.1, namely setting the nuclear magneton to the electron magneton, and the nuclear g-value to the electron g-value (+0.2), and the hyperfine coupling becomes, de facto, the J-coupling. The result of this simulation can be seen in Figure 1.9a. The 4 couples states (S_0,T_0 and $T_\pm$) can be seen to be the eigenstates of the system at all the field values (compare to dashed lines – spin-1 particle). S_0 and T_0 states are split by the J-coupling constant and their energies stay invariant with the magnetic field. T_+ and T_- states posses a magnetic moment and therefore interact with the magnetic field but, since they are in eigenstates of the system, they do not evolve in time.

In order to allow for the development of the MFE, the RP species have to become less strongly coupled. In the above cases this happens by excitation transfer through the body of the protein (along the chain of tryptophan residues), or by diffusion of the RP species away from one another in the solvent cage. In either case, if the interaction between the species is weak enough such that they are no longer in a pure ST eigenstate, they can undergo spin-mixing.

$$\mathbf{H} = \begin{array}{c} \\ \langle S,\alpha| \\ \langle S,\beta| \\ \langle T_0,\alpha| \\ \langle T_0,\beta| \\ \langle T_+,\alpha| \\ \langle T_+,\beta| \\ \langle T_-,\alpha| \\ \langle T_-,\beta| \end{array} \begin{array}{c} \begin{array}{cccccccc} |S,\alpha\rangle & |S,\beta\rangle & |T_0,\alpha\rangle & |T_0,\beta\rangle & |T_+,\alpha\rangle & |T_+,\beta\rangle & |T_-,\alpha\rangle & |T_-,\beta\rangle \end{array} \\ \begin{bmatrix} 0 & 0 & A_1+B_1 & 0 & 0 & -A_2 & 0 & 0 \\ 0 & 0 & 0 & -A_1+B_1 & 0 & 0 & A_2 & 0 \\ A_1+B_1 & 0 & 0 & 0 & 0 & A_2 & 0 & 0 \\ 0 & -A_1+B_1 & 0 & 0 & 0 & 0 & A_2 & 0 \\ 0 & 0 & 0 & 0 & A_1+B_2 & 0 & 0 & 0 \\ -A_2 & 0 & A_2 & 0 & 0 & -A_1+B_2 & 0 & 0 \\ 0 & A_2 & 0 & A_2 & 0 & 0 & -A_1-B_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & A_1-B_2 \end{bmatrix} \end{array}$$

$$A_1 = {}^{a}/_{4} \qquad B_1 = \frac{1}{2}(g_1 - g_2)\mu_B B_0$$

$$A_2 = \sqrt{2}A_1 \qquad B_2 = \frac{1}{2}(g_1 + g_2)\mu_B B_0$$

Figure 1.8: Hamiltonian matrix for the interaction of two electronic, and one nuclear spin, in mixed basis. ST labels represent the overall spin of the electron pair. $\alpha\beta$ labels represent the spin state of the nucleus. Only the electron Zeeman, and the hyperfine interaction have been included. The electronic spins are weakly coupled, and hence the Hamiltonian *is not represented in its eigenbasis.* The ST_0 and $T_\pm$ sub-manifolds are indicated in red and blue respectively. Hamiltonian reproduced from the work of Jonathan Storey.[28]

Spin mixing

To understand the spin mixing process in the RP, consider the Hamiltonian for a pair of radicals and a nucleus in a magnetic field shown in Figure 1.8. All-but-crucial interactions have been removed from this Hamiltonian for simplicity. Since the electrons are no longer strongly coupled, no dipolar or J-coupling is included. The nuclear Zeeman interaction is also omitted. This Hamiltonian encapsulates only the hyperfine coupling of one of the electrons to the nucleus (A-terms) and the two electron Zeeman interactions (B-terms). The state of the RP have been expressed in ST basis, while the α/β labels refer to the nuclear spin state.

The two sub-manifolds of interest are the ST_0, and the $T_\pm$ states indicated in blue and red respectively. In the absence of the hyperfine interaction, these manifolds would never mix. The $T_\pm$ states would stay as pure states and, since these are not the eigenbasis of the system, the ST_0 states would mix as expected (see Figure 1.4). The hyperfine interactions (A-terms in Figure 1.8) not only mix the ST_0 levels, but drive the mixing of the two sub-manifolds by $S \leftrightarrow T_+$ and $S \leftrightarrow T_-$ mixing (they appear on the off-diagonals of the ST_0 and $T_\pm$ sub-manifolds). The relative strength of this hyperfine mixing drives the magnetic field effect.

To see this, the Hamiltonian for a 2 electron and 2 nuclei analogous to one from Figure 1.8 has been used to propagate a spin density matrix starting from an initial spin-polarised S state, by the Liouville-Von-Neumann equation as described in Section 'Density matrix formalism'. The singlet probability can be obtained as the sum of the first two elements of the trace of the density operator (probabilities of the two singlet states $|S_0,\alpha\rangle$ and $|S_0,\beta\rangle$). The resulting singlet yield is the singlet probability integrated over its lifetime (here modelled as an exponential envelope with a time constant of 2 µs). The simulation of the singlet yield as a function of applied magnetic field can be done as follows:

```
t = linspace(0,100e-6,1e4+1); dt = t(2)-t(1); % Time vector
B0 = unique(sort([10.^linspace(0,log10(300),50)])); % B0 in Gauss
hfc = [3 27]; % Hyperfine couplings / G
for Bi = 1:numel(B0)
    S(Bi,:) = liouville_von_neumann_fun(t,hfc,B0(Bi));
end
E = exp(-t/2e-6); % Exponential lifetime envelope
S_yield = sum(bsxfun(@times,S,E),2)./sum(E); % Normalised singlet yield
```

[aq]The encounter between the RP species in solution is random, and therefore a statistical distribution of singlet and triplet states is formed. A field effect can also have its origins in non-equilibrium state populations formed at this stage of the RP creation. This effect is, however, usually much weaker than the RP mechanism.

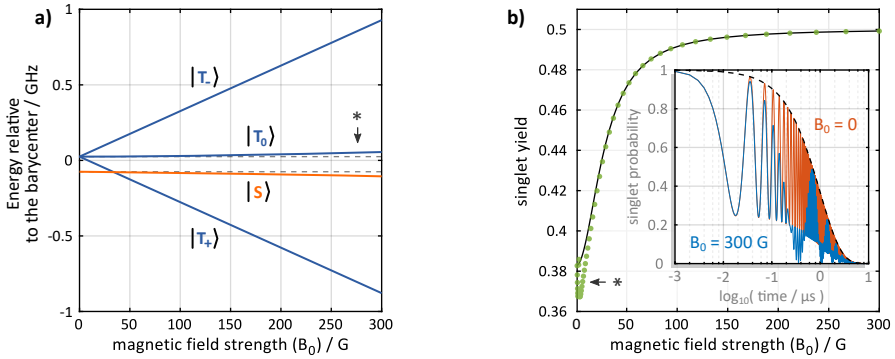


Figure 1.9: a) Energy level diagram of the coupled spin states of the radical pair as a function of applied external magnetic field. S and T_0 levels have no component of magnetic moment in the direction of the field and are unperturbed by it. $T_\pm$ levels poses a magnetic moment in the field direction and are thus split from the ST_0 manifold by the Zeeman interaction. S and T_0 states are split by an exchange interaction of –100 MHz to make the splitting easily visible.[as] The point where the Zeeman splitting of the coupled state becomes comparable to J-coupling is indicated with ∗. **b)** Singlet yield of a singlet-born radical pair with exponential decay lifetime of 2 µs, plotted as a function of applied magnetic field, as calculated in Listing 1.6. The black line is a fitted Lorentzian lineshape with full width at half maximum of $B_{1/2} = 28.3$ G. The low field effect has been indicated with ∗. The insert shows the time evolution of the singlet probability for zero field and $B_0 = 300$ G.

Listing 1.6: Singlet yield as a function of magnetic field

where the `liouville_von_neumann_fun` has been adapted from one published by the author,[29] and can be seen in Listing A.30. The simulation runs over 100 µs and up to 300 G. The resulting singlet yield as a function of the magnetic field strength is plotted in Figure 1.9b. This is referred to as the magnetically-altered reaction yield (MARY) curve, and its overall shape is well-modelled by a Lorentzian lineshape (black line in Figure 1.9b).

Note the departure from the pure Lorentzian lineshape at very low fields (indicated in 1.9b with ∗). This is not a bug, but a feature of particular interest for the MFE research. Certain off-diagonal elements of the Hamiltonian vanish at zero field, but even the presence of a very weak field makes them non-zero. This unlocks additional mixing pathways between the ST_0 and $T_\pm$ manifolds and increases the rate of the $S \leftrightarrow T$ mixing.[ar] The behaviour at low fields (on the order of hyperfine interaction) can, therefore, be opposite to that at high fields (much larger than hyperfine). This can result in a pronounced 'bump' in the MARY curve at low fields, which is referred to as the low field effect (LFE). The main contribution to the exact shape and extent of the LFE comes form the hyperfine interactions and it can thus be modulated by isotopic substitutions (e.g. H→D leading to decrease of the hyperfine coupling in one of the RP species). This feature appears, but in a highly irregular manner for the case of two electrons with a single nucleus (and isotropic hyperfine interaction).[30] Hence, in order to show a more typical behaviour, the 2 electron and 2 nuclei system has been simulated as opposed to, arguably simpler, 2 electron 1 nucleus one matching the Hamiltonian from Figure 1.8.

This mechanism of spin-mixing described so far is the hyperfine mixing. While the above has been a proper quantum-mechanical treatment (albeit omitting many interactions of the RP), it can prove useful to describe it, in addition, in a semi-classical way. Consider the vector model of a spin system. If the RP partners have different hyperfine interactions, they can be thought of as experiencing different local magnetic fields due to the magnetic dipoles of their molecular environment. If these fields are small compared to the externally applied magnetic field, the overall field at each radical (external + hyperfine) has the same orientation and approximately the same magnitude as the external magnetic field. The spins therefore

[ar] At zero field the total angular momentum of electrons and nuclei F and its projection onto the quantisation axis M_F must be conserved. The presence of a field relaxes these selection rules and only M_F is has to be conserved.
[as] For RPs in solution, exchange interaction usually destabilises S state with respect to T_0, hence the negative sign of J.

precess (Larmor precession) about common axis and the only state mixing allowed is the $S_0 \leftrightarrow T_0$ mixing (as can be see from Figure 1.3). This corresponds to the 'high' field effect (normal MFE). If the hyperfine fields are comparable to the external field, the overall field experienced by each radical is different in both magnitude and direction. The spins precess at different rates about different axes and mixing between all of the ST states is allowed. This corresponds to the LFE region. At zero field, the spins do note have a quantisation axis (it is arbitrary) and no Larmor precession, and hence any additional $S_0 \leftrightarrow T_0$ mixing takes place. The rate of $S \leftrightarrow T$ mixing is, overall, inversely proportional to the energy gap between the $T_\pm$ states and the ST_0 manifold (see Figure 1.9a).[at]

There are other mechanisms though which a RP spin state could mix. One such mechanism relies on the radicals having slightly different effective g-factors and thus Larmor precession frequencies (see vector model in Figure 1.3). The difference in Larmor frequencies scales with applied B_0. This can be seen to a very small degree in Figure 1.9a, where the ST_0 states begin to mix with the $T_\pm$ states at ca. 300 G, and therefore their energy begins to shift. The difference in electron g-values used in the simulation in Figure 1.9a ($\Delta g = 0.2$) is much larger than commonly seen in organic-based radicals (without heavy atoms). There the Δg-mechanism of spin mixing becomes significant at much higher fields. Since such systems are not considered in this work, this mechanism is of no significance here.

Many features of RP spin evolution are, of course, not encapsulated by the Hamiltonian from Figure 1.8, nor even by the Liouville-Von-Neumann equation itself. The RP spin polarisation has a finite lifetime. Even if the RP were to live forever (no reaction), eventually the spins would dephase, lose the spin coherence (T_2 relaxation) and relax back to a thermal distribution (T_1 relaxation). The radical pair also does not live forever. In the simulation in Listing 1.6 this has been accounted for with a separate exponential envelope, not included in the Liouville-Von-Neumann equation. The formation of the RP has also not been covered, and instead the initial state of the RP has been set to the S state. The shortfall of the Liouville-Von-Neumann equation is that it assumes a time-invariant Hamiltonian. More advanced and complete formulations of spin dynamics, which are able to simulate the aforementioned effects do exist,[au] but will not be discussed here. One important effect which will, however, be mentioned. Motion of one of the spins broadens the MARY curve with respect to its theoretical hyperfine-only width.[31] This motion could be a result of, e.g. geminate electron transfer between like species in solution (self-exchange) or motion of the electron over the neighbouring residues within a protein. The MARY curve, its exact shape and behaviour are therefore of central interest to the MFE research.

Recombination and accumulation

$S \leftrightarrow T$ mixing and magnetically altered reaction yield of one of the spin states is not, by itself, sufficient for magnetosensitivity. In order to develop MFEs, *the rates of reaction of different spin states have to be different*. It is not enough for the singlet yield to be altered by the magnetic field, if its fate is not distinct to that of a triplet. As mentioned in the previous section when integrating the singlet probability, the RP has a finite lifetime. Following any spin evolution, the RP partners can undergo a back electron transfer (BET), non-radiative (or, indeed, radiative) transition back to the ground state (geminate recombination). In the case of BET the exciton is retained, but merely returns to its precursor. This happens, for example, in the pyrene/1,3-dicyanobenzene system discussed in Chapter 3, were the RP returns to its parent exciplex state. In contrast, flavin-based systems investigated in this work undergo a non-radiative transition to the ground state. The exciton level is lost and the energy is released to the environment through vibrational relaxation. Both of these pathways must happen under conservation of total spin. Hence, since flavin is a closed-shell

[at] The hyperfine interaction can be considered as a small perturbation to the overall system (at least at higher fields). This perturbation mixes all the states, and it can be shown by 2nd order perturbation theory that this effect is proportional to the inverse of the energy difference between the energy levels considered – $\propto \Delta E^{-1}$. At low fields hyperfine interaction mixes all the states together. As the field increases, the energy gap between $T_\pm$ and ST_0 states increases and the corresponding rate of $S \leftrightarrow T$ mixing decreases.

[au] For example the superoperator formalism which allows for description of phenomena like spin relaxation (e.g. Redfield theory) and dephasing.

singlet, only the singlet RP recombines to the ground state. Any effects of the magnetic field on the singlet character of the pair will, therefore, directly affect the quantum yields of this reaction.

Both spin states of the RP can also undergo a spin-incoherent decay, either by spin decoherence (in Cry proteins), or escape of one of the RP species from the solvent cage (in solution).[av] This reaction pathway is available to all the spin states involved, and usually leads to creation of long-lived species. This is, in fact, another crucial step of developing a significant MFE on a chemical system. Sans exceptional circumstances,[7] the effect of the magnetic field on the reaction yield is very small. In order for the magnetic field to alter the macroscopic state of a chemical system, its effect has to accumulate over the course of multiple turnovers of the photocycle. This poses another restriction on the reaction rates of the species involved in the RPM – *the long-lived downstream products of the RP have to have an appreciable steady-state concentration.*

The requirement for both of these conditions can be demonstrated by simulating a slightly altered version of a photocycle from Equation 1.62, namely:

$$\mathrm{A} \xrightarrow{k_{\mathrm{AB}}} \mathrm{B} \xrightarrow{k_{\mathrm{BC}}} \mathrm{C} \qquad \mathrm{C} \overset{k_{\mathrm{CDm}}(t)}{\longleftrightarrow} \mathrm{D} \qquad \mathrm{C} \xrightarrow{k_{\mathrm{CE}}} \mathrm{E} \qquad \mathrm{D} \xrightarrow{k_{\mathrm{DE}}} \mathrm{E} \qquad \mathrm{E} \xrightarrow{k_{\mathrm{EA}}} \mathrm{A} \qquad \mathrm{C} \xrightarrow{k_{\mathrm{CA}}} \mathrm{A} \tag{1.63}$$

C is now a sole product of B. C and D undergo a field dependent mixing with rate of $k_{\mathrm{CDm}}(t)$, and return to the ground state A via a long-lived intermediate E. C can also return back to the ground state with rate k_{CA}. This simulation can be performed easily as follows:

```
y0 = [1 0 0 0 0]'; % Initially only A
kAB = 1e3; kBC = 1e3; % A->B and B->C fast
kCE = 10; kDE = kCE; % C->E and D->E slow and equal
kEA = 1; % E->A very slow
kCA = ⟨...⟩; % [5e2, 5e3, 5e4]
kCDm = @(t) 10*((sin(2*pi*log(t))+1)/2); % Rate of CD mixing
kCd = @(t) kCA+kCE+kCDm(t);  kDd = @(t) kCDm(t)+kDE;
K = @(t) [-kAB  0   +kCA      0      +kEA ; % Rate matrix
          +kAB -kBC  0        0       0   ;
           0   +kBC -kCd(t)  +kCDm(t) 0   ;
           0    0   +kCDm(t) -kDd(t)  0   ;
           0    0   +kCE     +kDE    -kEA];
f = @(t,y) K(t)*y; [t,y] = ode15s(f,[1e-5 1e2],y0); % Solve
```

Listing 1.7: Demonstration of the kinetics of MFE accumulation

The rates k_{AB} and k_{BC} were chosen to be fast (on the timescale of the simulation) since they mimic the photoexcitation and RP formation respectively. The species C and D mimic the different RP spin states. The rate of their mixing k_{CDm} is set to oscillate in logarithmic time.[aw] The magnitude of the oscillation seen on other states will therefore correspond to the magnitude of the MFE.[ax] The rate k_{EA} is very low, such that E represents the long-lived species which accumulate over multiple turnovers of the photocycle. The rate k_{CA} represents the geminate recombination of the RP back to the ground state.

If k_{CA} is varied such that it is slower, on the order of, and faster than k_{CE}, the requirements for an appreciable accumulated MFE can be demonstrated. This can be seen in Figure 1.10. If $k_{\mathrm{CA}} < k_{\mathrm{CE}}$ (Figure 1.10a), the fates of C and D (RP species) are approximately the same since they mostly decay to the ground state through E. Hence, a high steady-state concentration of long-lived photoproducts does accumulate, but it does not develop an appreciable MFE, since the field sensitive step (k_{CDm}) does not alter the pathway the reaction takes through E. On the flip side, if $k_{\mathrm{CA}} > k_{\mathrm{CE}}$ (Figure 1.10c), the fates of C and D *are* different. In this case, however, the quantum yield of E is very small since the predominant pathway through the reaction goes straight from C to A. The steady-state concentration of the accumulated E, and the field effect on it are

[av] These free radicals can then recombine to form so called F-pairs. These will be formed in a statistical spin distribution of $^1/_4$S and $^3/_4$T. In certain cases the F-pairs can contribute to the MFE.

[aw] An oscillation in logarithmic time is of the form $f(t) = \sin\left(\log_{10}(t)\right)$. Such an oscillation can be considered a down-chirp.

[ax] Note that, since the frequency of the oscillation of k_{CDm} changes, the oscillations in other states develop only on timescales above which they can respond to it.

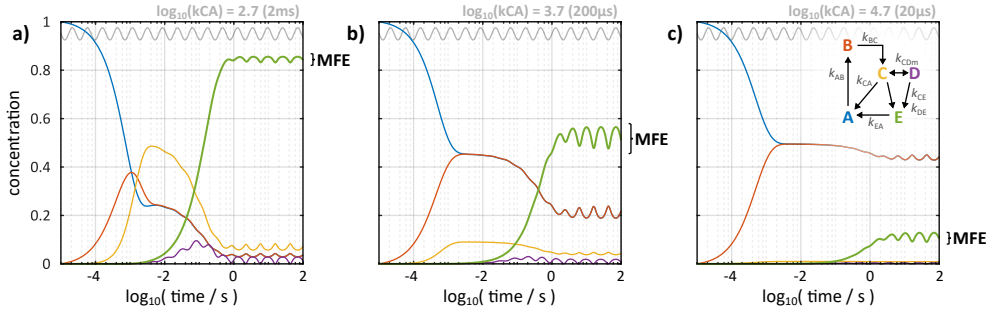


Figure 1.10: Demonstration of the kinetics of MFE accumulation, as calculated in Listing 1.7. The three plots show the behaviour of the photocycle from Equation 1.63 for three different values of the rate k_{CA} – proxy for the geminate recombination of the radical pair. The grey line oscillates at the frequency of k_{CDm}, and is shown for reference (since it oscillates in logarithmic time, on the semilog plot it does indeed appear as a simple sine wave) **a)** $k_{CA} = 5 \times 10^2\,\mathrm{s}^{-1} < k_{CE}$ **b)** $k_{CA} = 5 \times 10^3\,\mathrm{s}^{-1} \approx k_{CE}$ **c)** $k_{CA} = 5 \times 10^4\,\mathrm{s}^{-1} > k_{CE}$. The photocycle is shown in the insert.

therefore low. Finally, if $k_{CA} \approx k_{CE}$ (Figure 1.10b) both condition of 1) asymmetric fates of C and D and 2) appreciable yield of E are satisfied. The MFE on E can be seen to accumulate to a significant degree. Note that the rate of $C \leftrightarrow D$ mixing is always lower than k_{CA}. The effect of the magnetic field on a single turnover of the photocycle is therefore always low, and only its accumulant is observed.

The geminate recombination rate can be seen to have a strong effect on the size of the resulting field effect. For the case of FMN/HEWL system discussed in Chapter 4, this rate is mediated by the balance of diffusive separation of the RP species in the solvent cage and the coulombic attraction between the net-negative FMN and net-positive HEWL radicals. In cryptochrome systems investigated in Chapter 6 the RP species are situated within the protein body. The recombination of the RP has a very large drop in free energy, and therefore it falls within the Marcus inverted region (see Section 1.2). Evolutionary tuning of this rate could, therefore, happen by adjusting the precise protein environment of the RP and therefore the rate of recombination of the RP.

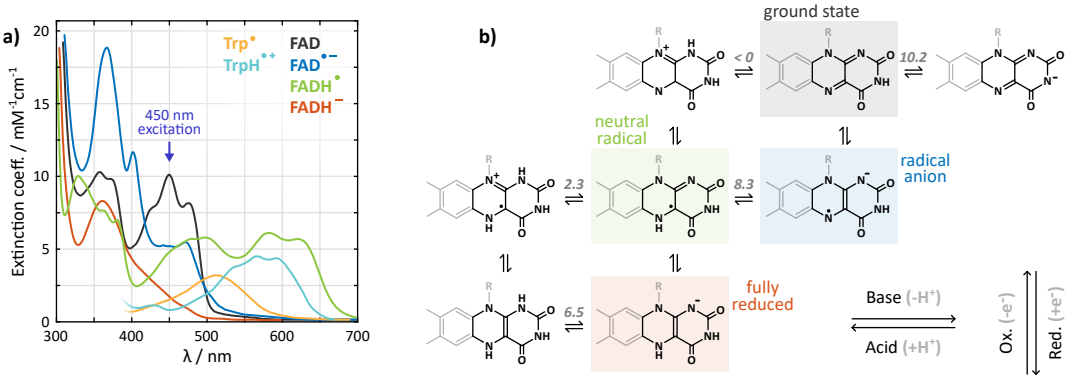


Figure 1.11: a) Absorption spectra of selected redox stated of FAD and Trp. The spectra of flavin species are reproduced from the work of Liu *et al.*[32] The spectrum of $Trp^{\bullet}$ and $TrpH^{\bullet+}$ are similarly reproduced from the work of Solar *et al.*, and Miller *et al.* respectively.[33;34] **b)** Redox and acid-base equilibria of flavins. The value given above the protonation/deprotonation equilibria correspond to pKa values.[35;36] The R-group and an aromatic ring of the isoalloxazine moiety do not play a (major) role in the reactivity shown here, and therefore have been greyed out.

1.5 Flavin-based systems

Finally, the specifics of magnetosensitivity of the flavin systems studied in this work will be introduced.

Types of flavins Flavins are a family of molecules based on an isoalloxazine group – tricyclic aromatic moiety with low-lying photoexcited states. The excitation from ground state (singlet, S_0) to first (S_1) and second (S_2) electronic excited states span the 400–500 nm and 300–400 nm bands and are therefore accessible with most common visible-range light sources. This property also gives flavins their characteristic yellow colour.[ay] The structure of flavin can be seen in Figure 1.11b. The nature of the R-group attached to N10 determines the exact nature of the flavin, and some of its non-chromophoric properties:

- The simplest flavin (bar the isoalloxazine itself) is the *lumiflavin* (LF) – with R-group being simply a methyl. Lumiflavin is a common photodegradation product of flavins.
- If R-group is a ribityl chain, the flavin is referred to as a *riboflavin*. Riboflavin will not be used in this work, but has been used in the magnetoreception research in the past and does exhibit magnetic field effects when paired with a suitable electron donor.
- If a phosphate group is added to the end of the ribityl chain, the resultant flavin is referred to as the *flavin mononucleotide* (FMN). This flavin is the basis of the most commonly used solution-based magnetosensitive system, and will be studied extensively in this work (in Chapters 4 and 5).
- An associate of the FMN with an adenine nucleotide yields the *flavin adenine dinucleotide* (FAD). FAD is the form of flavin in which it most often appears as a biological cofactor. Cryptochrome molecules studied in Chapter 6 associate an FAD cofactor which forms the basis of their magnetosensitivity.

Redox states Flavins are known not only for their photoactivity, but also for their coupled redox and acid/base chemistry. The diagram showing different redox and protonation states of FAD can be seen in Figure 1.11b. A ground state (GS) FAD usually exists in a single protonation state, with single labile proton with pKa ≈ 10.2 and no way to delocalise a net-positive charge if it were protonated. FAD can, however, easily accept a single electron, as the negative charge is stabilised by delocalisation into two neighbouring ketone groups. The radical is also stabilised by delocalisation into the aromatic ring of the isoalloxazine. A significant proportion of the radical spin density does, however, reside on N5 as shown in Figure 1.11b, which makes the hyperfine coupling of the N5 nitrogen nucleus with the electron one of the main contribution to the RP spin dynamics.[37] The singly reduced FAD is referred to as the *radical anion* ($FAD^{\bullet-}$). $FAD^{\bullet-}$ can be protonated with a pKa ≈ 8.3, to form a *semiquinone radical* species ($FADH^{\bullet}$). The protonation occurs on N5 which may destroy the spin coherence and therefore yield the downstream RP

[ay] In fact, the name 'flavin' comes from Latin '*flavus*' meaning 'yellow' or 'gold'.

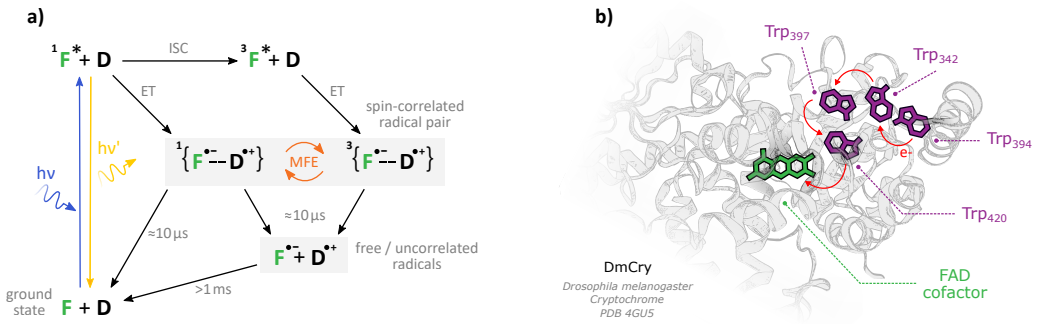


Figure 1.12: a) Generalised photocycle of flavin-based RP chemistry. The ordinate broadly represents the energy of the represented species. The photoexcitation and the resulting fluorescence have been indicate with $h\nu$ and $h\nu'$. Note that, depending on the system, the intersystem crossing (ISC) / electron transfer (ET) competition may put this photocycle in different regime. Compare this photocycle with one shown in the insert of Figure 1.10c. **b)** Electron transfer chain through the tetrad of Trp residues in *Drosophila melanogaster* cryptochrome, *Dm*Cry WT.[38] The Trp residues as well as the FAD cofactor are shown in their approximate positions. The structure rendering produced based on the protein databank (PDB) entry 4GU5.

product. Whether the protonation of FAD happens depends, of course, on the pH in the vicinity of the flavin. In the FMN/HEWL system (see later), as well as in one of the cryptochromes investigated (*At*Cry), this is indeed the reaction which terminates the spin mixing and dictates the lifetime of the RP. In conditions of high reduction potential, the semiquinone radical can accept another electron to form the *fully reduced* form of FAD. This is not usually desired. In biological systems flavins act as both one- and two-electron redox cofactors. The latter is, however, probably better thought of as a double one-electron cofactor since the fully reduced state of FAD *can not* be reached directly without a potentially very highly energetic intermediate or a concerted/coupled electron-hydrogen transfer. The natural reactivity of flavins is, therefore, based on radical species. Note the values of pKa given are approximate values for FAD in solution. The protein environment, or change in the R-group can influence the redox chemistry of the specific flavin involved.

Spectral properties Different redox states of flavins have different spectral properties. This can be seen in Figure 1.11a where the absorption spectra for the main redox species discussed above are shown. The shape of the GS FAD shown is characteristic of the FAD in the cryptochrome binding pocket,[az] and its presence and definition have been used to qualitatively asses the condition of the protein samples measured. The radical anion absorption can be seen extending slightly more towards the red end of the spectrum as compared to the ground state, as well as having a pronounced absorption at 400 nm. $FMNH^{\bullet}$ is the most characteristic of all the redox species, extending far into the red due to the extensive conjugation through the entire isoalloxazine ring present. Finally, the fully-reduced form of flavin is almost entirely transparent in the visible region and has not been observed in this work.

All of the flavin-based radical pairs investigated here also involve tryptophan-centered radical species. Ground-state tryptophan does not absorb in the visible range, but its neutral and protonated radical species do. These have also been shown in Figure 1.11a.

Photocycle The generalised photocycle which governs the flavin-based RP chemistry can be seen in Figure 1.12a. The cycle gets initiated by the $S_0 \rightarrow S_1$ transition of the flavin (F) usually pumped with 450 nm light (indicated by $h\nu$). The excited state (^{1}F*) can return directly to the ground state radiatively (fluorescence, indicated by $h\nu'$) or become quenched by an electron donor (D) to form a singlet-born RP. If the rate of quenching by the electron donor is to low, it becomes outcompeted by the intersystem crossing (ISC) of ^{1}F*

[az] The residues neighbouring the isoalloxazine split the energies of the, otherwise degenerate, $S_0 \rightarrow S_1$ optical transitions corresponding to the three internal coordinates of the isoalloxazine ring. Effectively, the tensor for the interaction of the electric field of the incoming photon with the transition dipole moment of the isoalloxazine becomes anisotropic.

to form the excited molecular triplet state (^{3}F*).[ba] This is usually the case in solution for *intermolecular* RP formation. The following ET then results in a triplet-born radical pair.[bb] The subsequent spin-evolution occurs as described in Section 1.4. Only the singlet RP can undergo geminate recombination, while both singlet and triplet states can form secondary products. As mentioned, this can often be the protonation of the radical species, as well as their escape and diffusion away from the solvent cage (or both). The formation of the secondary products of the RP can also simply be the loss of correlation of the RP species due to their interactions/relaxation with the environment, not necessarily any chemical change. The order-of-magnitude timescales for the geminate recombination of singlet RP, as well as formation of the secondary products in cryptochrome proteins have been indicated in Figure 1.12a. Note, by comparing Figure 1.12a with the photocycle discussed in Section 1.4 (insert of Figure 1.10 c), that the rates of the processes involved are well-tuned for the development of MFEs.

Importantly, note that the photocycle shown in Figure 1.12a is *simplified* – other reactions, not indicated in Figure 1.12a, are also common. For one, Figure 1.12a does not include any photodegradation pathways. Furthermore, photoexcited flavin can also return to the ground state through vibrational relaxation and through other (potentially long-lived) intermediates not involved in the magnetosensitive branch of the photoscheme. Even further considerations include the resonant energy transfer from (and to) the flavin moiety by a neighbouring chromophores (e.g. another flavin molecule),[bc] reactions of the oxidised donor species, as well as more detailed chemistry of the long-lived components of the photocycle. The latter is, in fact, least well-understood part of flavin magnetosensitivity. It has, however, come under close investigation in past years, not in small part due to the investigations performed and results obtained as part of this work.

FMN/HEWL The model cryptochrome system investigated thoroughly in this work is the FMN and hen-egg white lysozyme (HEWL) (100 µM and 1 mM unless otherwise indicated). In Chapter 4, investigations of the interaction of FMN with crystallised HEWL are performed, whereas in Chapter 5 the FMN/HEWL system is used as a model system of flavin-based magnetosensitivity. The net-negative change of the phosphate group on FMN allows it to loosely associate with the net-positive surface of HEWL. The electron donor and RP partner is either of the two tryptophan residues on the surface of HEWL (W62 and W123).[39] This yields strong triplet-born MFE signals. The details of the interaction of FMN with HEWL are not known, but there exists evidence for a selective association of FMN with a pocket on the surface of HEWL.[bd]

Cryptochromes The biologically-relevant magnetosensitive systems investigated in this work are the cryptochrome (Cry) proteins. The cryptochromes occur in 4 variants in birds. Only certain variants bind the FAD cofactor, and their full array of biological functions is still under investigation. The main known roles of the cryptochromes (Cry1 and Cry2) is the regulation of the circadian cycle in animals, and phototropism in plants.[41] It has been suggested that Cry4 is responsible for animal magnetoreception. The formation of the RP in cryptochromes occurs by a successive *intramolecular* ETs through a triad/tetrad of Trp residues.[be] This has been shown in Figure 1.12b on the example of the fruit fly cryptochrome. The first two ET steps are usually very rapid (ps to ns) followed by one/two slower ones (>10 µs).[16;38;42] Since the initial ET is rapid, the RP formed is singlet-born. This light-induced formation of magnetosensitive RPs has been shown in the

[ba] The rate of intersystem crossing is also increased in the presence of any heavy nuclei like e.g. sulphur in cysteine amino acid residues

[bb] Singlet-born RPs can also exist in solution if the RP species are strongly constrained to one another's vicinity, for example by being captured in a micellar microreactor, or having originated from a common bound precursor as it is the case for the pyrene/1,3-dicyanobenzene system investigated in Chapter 3.

[bc] The flavin-flavin interactions have been shown to give rise to a family of magnetic field effects. This is presently under careful investigation in the Timmel group

[bd] The triplet-born field effects of FMN/HEWL could not originate in slow ET from the excited molecular singlet (*F^1), but in to the interaction of any cysteine residues which are present close to both W62 and W123 with the flavin moiety and the resulting increase in the rate of intersystem crossing. C64 and C76 are close to W62, while C6, C30 and C115 are close to W123 (based on PDB entry 2VB1).[40]

[be] It is suggested that any resulting MFE is transduced to the conformational changes of the C-terminus of the protein to then trigger a neural signal.

past in non-migratory species (*At*Cry and *Dm*Cry), but never before in migratory species. The investigation of Cry4 proteins from such migratory species (European robin) can be seen in Chapter 6.

Chapter 2

Computational methods for experimental physical chemistry

A significant portion of work done for the purpose of this thesis involved a mixture of computational and mathematical methods. This chapter outlines the main ones of these, with strong focus on their applications in experimental chemistry and data analysis. This is not intended to be a tutorial in these subjects, but merely an outline of what the author considers the most important, high-level concepts and possible pitfalls.

2.1 Time-frequency analysis

Fourier transforms (FT) are ubiquitously present, yet the details of it are often poorly understood in experimental science. A Fourier transform is a brilliant idea of an 18$^{\text{th}}$ century French mathematician Joseph Fourier, which says that any function can be expressed as an infinite sum of different frequency and phase sine waves. In general, the Fourier transform is an operation which takes in any complex analytical function $f(t)$, and maps it onto a *unique* (also complex) function $F(\nu)$, such that $F(\nu) = F\left(t^{-1}\right) = \mathrm{FT}\left\{f(t)\right\}$. The magnitudes and phases of the waves defining $f(t)$ are defined by the absolute value and the argument of $F(\nu)$ respectively. There exist, correspondingly, an inverse Fourier transform (iFT) which maps (also uniquely) the FT output $F(\nu)$ onto the corresponding input $f(t)$, such that for any sufficiently well-behaved function $f(t) = \mathrm{iFT}\left\{\mathrm{FT}\left\{f(t)\right\}\right\}$.[bf] This is the 'purest' form of the FT which is of much use in pure mathematics and number theory.[43]

However, any real world signal processing, analysis, as well as most numerical simulations exist in the discrete domain – the signals in question are measured/calculated at discrete points and represented as a series of (x,y) samples. Moreover such signals are *finite*, and *not necessarily periodic*. The discussion here will therefore be limited to a discrete analogue of the Fourier transform (DFT).[bg]

In general the FT input-output pair will be defined on inverse domains. However, the domain of the input function of a DFT is most commonly time and the domain of the output is frequency (inverse time). Hence the Fourier analysis, as well as related techniques, are often referred to collectively as the time-frequency analysis. All the examples in this section will use timeseries signals for simplicity, such that the signals will be defined on time t and their FT on frequency ν domain (hence the '$f(t)$' and '$F(\nu)$' notation above).

Unless otherwise specified, the names 'Fourier Transform', 'FT', 'DFT' will all be used interchangeably to refer to the operation of discrete Fourier transform. Similarly, the name of the algorithm used to compute the DFT: 'Fast Fourier Transform' or 'FFT' (see Section 'Fast Fourier transform') will explicitly refer to the operation of discrete FT. All the signals will be of finite length, uniformly-sampled, real and finite in magnitude.

Discrete Fourier transform

Consider a *continuous* function $f(t)$. The magnitude and the phase of the periodic component at frequency ν contained in it will be the overlap integral of $f(t)$ with an oscillation of that frequency. A general form of such oscillation is a complex exponential $\exp(-2\pi i \nu t)$, and the corresponding overlap integral is the

[bf] There is no one definition of what 'well-behaved' means. In this context it refers to *integrable, bounded* function with a *finite number of discontinuities*.

[bg] Note that in chemistry the acronym DFT often refers to 'density functional theory' – a computational technique in quantum mechanical modelling.

definition of the (continuous time) Fourier transform:

$$F(\nu) = \int_{-\infty}^{+\infty} f(t) \exp(-2\pi i \nu t)\, \mathrm{d}t \tag{2.1}$$

Let's consider what would happen if $f(t)$ was to be discretised in time with sampling period of δt over N sampling points. f_k will denote the value of f at the k^{th} sampling point, where k is the time-sample index running from 0 to $N-1$. In general it is simplest to think of such $f(t)$ as an ordered set (vector) of value, together with the value of δt. $f(t)$ can, however, also be thought of as a 'continuous' function of time which is zero everywhere, except at the sampling points where it is a delta function.[bh] The peak integral of the delta at the k^{th} sampling point is equal to f_k. This train of delta functions can be plugged into the above equation for the continuous FT. Since the value of the integral is non-zero only at the positions of the deltas (sampling points of $f(t)$), the FT equation simplifies to:

$$F(\nu) = \delta t \sum_{k=0}^{N-1} f_k \exp(-2\pi i \nu) k \tag{2.2}$$

This is a continuous function of ν which is periodic in $1/\delta t$. Because of that, all of the information about $F(\nu)$ is contained in a single period. Moreover, no information is lost if this period of $F(\nu)$ is sampled with at least N samples.[bi] Similarly to time sampling, lets define the frequency-sample index j going from 0 to $N-1$ with frequency sampling period $\delta\nu$. The FT equation can therefore be written as:

$$F_j = \delta t \sum_{k=0}^{N-1} f_k W_N^{jk} \tag{2.3}$$

where F_j is the value of the j^{th} sample of $F(\nu)$, and W_N is defined as:

$$W_N = \exp\left(-\frac{2\pi i}{N}\right) \tag{2.4}$$

W_N is called the 'N^{th} root of unity' since its a unit vector on a complex plane which, when raised to the power of N gives a purely real unit vector. An analogous analysis leads the equation for the iFT:

$$f_k = \delta\nu \sum_{j=0}^{N-1} F_j W_N^{jk} \tag{2.5}$$

Since both time and frequency domain have the same number of points, it further follows that:

$$\delta\nu\, \delta t = \frac{1}{N} \tag{2.6}$$

It is important to note that, as much as the symmetry between these two equations is arguably quite beautiful, most implementations of the DFT take the factor of δt from the forward FT and move to the iFT

[bh] Note the difference between a Dirac delta function, a Kronecker delta and a unit impulse. Dirac delta has an infinite magnitude, but unit integral $\int_{-\infty}^{\infty} \delta(t)\, \mathrm{d}t = 1$. A Kronecker delta is a function of two variables i and j (non necessarily, but usually, integers) which is zero, unless $i = j$. A unit impulse has zero integral, but a unit magnitude at one point. The unit impulse can be thought of as continuous Kronecker delta where one of the inputs is 0. These are not, of course, rigorous mathematical definitions.

[bi] It is one of the implications of the Shannon information theory (which itself if closely liked to the Fourier analysis) that for a general case of N bits of information, there does not exist a smaller set of bits which can be used to represent all of the information of the original N bits. This places a lower bound on how many samples of $F(\nu)$ one can take. Since $F(\nu)$ comes from a finite length signal (f_k) it is band-limited. This means it does not contain frequencies higher than certain value and can be accurately represented with as little as N points. This is a lower bound of N. Nothing prevents one from oversampling the DFT, but no additional information is gained. Oversampling is equivalent to gluing on copies of the original signal onto the back of itself and performing a normal FT.

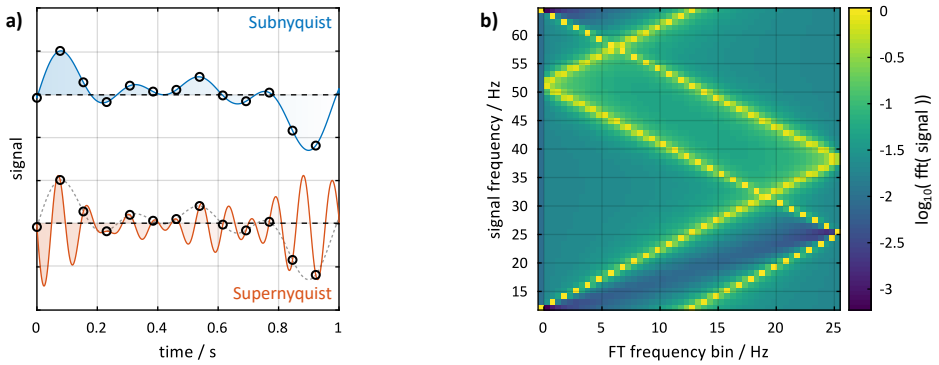


Figure 2.1: a) Sub-Nyquist and super-Nyquist reconstruction of a signal **b)** 'reflection' of a tone off the Nyquist frequency limit. Nyquist reflection. Spectrum of a signal with frequencies at $0+\nu_x$ and $\nu_s/4+\nu_x$ plotted as a function of ν_x. The peaks sharpen and blur and and out as they get in and out of phase with the bins of the `fft` (see 'Frequency and amplitude measurements').

such that the forward and inverse DFT are:[bj]

$$F_j = \sum_{k=0}^{N-1} f_k W_N^{jk} = \mathrm{DFT}\left\{f_k\right\} \tag{2.7a}$$

$$f_k = \frac{1}{N}\sum_{j=0}^{N-1} F_j W_N^{-jk} = \mathrm{iDFT}\left\{F_j\right\} \tag{2.7b}$$

This is done with no loss of generality as the change in the scaling of F_j is compensated for in the iDFT. It it therefore important to be mindful of this scaling as the relation f_k holds only for a FT pair defined together. The above equations are as implemented in Matlab and therefore will be used here.

The reversibility of the Fourier transform can be confirmed by comparing a signal with the transformed, and transformed back version of itself:

```
1 x = randn(1,1e3); % Random input
2 delta = max( x-real(ifft(fft(x))) )/eps; % = 5; Maximum error
```

Listing 2.1: FFT reversibility.

Where the use of `real()` is necessary to assert that result *is* purely real (see also Listing A.1). The maximum difference between the signals is on the order of machine precision for floating-point arithmetic and arises due to accumulated round-off errors.[bk] This is a remarkable degree of error cancellation for ca. 2×10^4 operations ($2\times N\log_2(N)$ for $N=10^3$).

Note that this still means that repeated application of the `ifft(fft(...))` will lead to eventual degradation of the signal due to accumulated floating-point errors (see Listing A.1). It is usually possible to avoid chaining the FT operations and instead operate just in time or frequency domain.

Sampling theorem

One of the fundamentals of discrete-time signal analysis is the Nyquist-Shannon sampling theorem. Lets consider a bandlimited signal – a signal whose components are limited to some band of frequencies below ν. For such a signal to be appropriately sampled (such that no information is lost in the process), it needs to

[bj] This is done for the reasons of speed. It is faster to perform one set of N floating point operations (division by $1/N$) than two (multiplication by $\delta\nu$ and then by δt). The DFT is used more commonly than iDFT, hence the additional mathematical operation is moved to the iDFT. Furthermore, this makes the DFT defined in this way completely independent of the exact value of the sampling frequency.

[bk] Most of current numerical computing hardware operates on double-precision floating point arithmetic. This scheme uses eight bytes to represent each number (hence 'double'). Only certain numbers can therefore be represented.[44] `eps` is the distance from 1 to the next number `eps` $=2^{-52}\approx2\times10^{-16}$.

be sampled at a frequency of, at least, 2ν. This is the fundamental statement of the sampling theory. For a given sampling frequency $\nu_s = {}^1/_{\delta t}$, the highest signal frequency which can be accurately represented is called the Nyquist frequency $\nu_n = {}^{\nu_s}/_2$. The idea of sampling theory can be illustrated by constructing two signals which, when sampled, appear alike. In general, any frequency $\nu > \nu_n$ will appear to look like a lower frequency $\nu_n - \nu$. Therefore, the two signal can be constructed as follows:

```
N = 13; t = linspace(0,1,N+1); t(end) = []; dt = t(2)-t(1);
fs = 1/dt; % Sampling frequency (13 Hz)
f = [0.79 2.11 3.05 4.11]'; % Signal frequency components / Hz
y1 = @(x) sum( sin(2*pi*x .* f) ,1); % Sub-Nyquist signal
y2 = @(x) sum( sin(2*pi*x .* (fs-f)) ,1); % Super-Nyquist signal
```

Listing 2.2: Demonstrations of sampling theorem.

Here a sum of 4 arbitrary single tones has been chosen to show that the sampling theory talks about signals in general, not only single tones. The two signals `y1` and `y2` have been plotted in Figure 2.1a). As the frequency of the signal increases slowly from below to above ν_n, it will appear to 'reflect' from the edge of the spectrum. This behaviour has been shown in Figure 2.1b) where the DFT spectrum has been calculated for a signal as a function of increasing frequency of its components. This 'reflection' is also most commonly referred to as aliasing.

The implications of sampling theory are therefore quite severe: no sampled signal can ever be trusted, since any part of it can come from some higher frequency which just appears like that which has been measured due to aliasing. In practice though this can be dealt with and sometimes even leveraged for high frequency signal detection.[bl] Most sensors comprise of some kind of analogue to digital converter (ADC) which typically implement an anti-aliasing low-pass (LP) filters. The job of the LP filter is to attenuate any high frequency components such that they do not alias down to below the Nyquist frequency ν_n and distort the signal. The effect of such a distortion can be shown with downsampling a noisy signal. Simply taking every M^{th} point from the signal is equivalent to sampling a noisy signal without anti-aliasing filter. The alternative is, of course, to first filter the signal which in this case has been done with Matlabs in-built `downsample` function:

```
N = 2^10; t = linspace(0,1,N+1); t(end) = []; dt = t(2)-t(1);
y = sin(2*pi*t.*2) + randn(size(t)); % Signal + noise
naive = y(1:64:N); % Take every 64'th point
aaliased = decimate(y,64); % Filter first, then decimate
```

Listing 2.3: Demonstration of signal downsampling.

The downsampled signals, as well as the resulting `fft` spectra are shown in Figure 2.2.

An important feature of the aliased peaks in the `fft` is that, as opposed to properly sampled signals, they move as a function of the sampling frequency ν_s. This can be used to identify the aliased part of the signal and sometimes even reconstruct the super-Nyquist spectrum.[bm] Moreover, assumptions about the physical nature of the underlying signal can be made. A temperature of a water reservoir, for example, is unlikely to change fast due to the high heat capacity of water. Such temperature measurement, therefore, will not contain many high frequency components except for the electrical noise of the sensor, and so can safety be assumed to be properly sampled.

[bl] for example, sub-Nyquist frequency sampling is, broadly speaking, the principle of structured light surer-resolution microscopy techniques. The low (spatial) frequency structured light demodulates the image (signal) and allows one to computationally reconstruct its high resolution (super-Nyquist) components. See Section 'Convolution theorem' and Chapter 3 for discussion on modulation/demodulation.

[bm] This is, for example, the basis of structured-light illumination superresolution microscopy. The image is spatially sampled with light of different striped patterns (spatial frequencies) in order to measure spatial frequencies which would, originally, be limited by the diffraction limit of light (spatial Nyquist limit).[45]

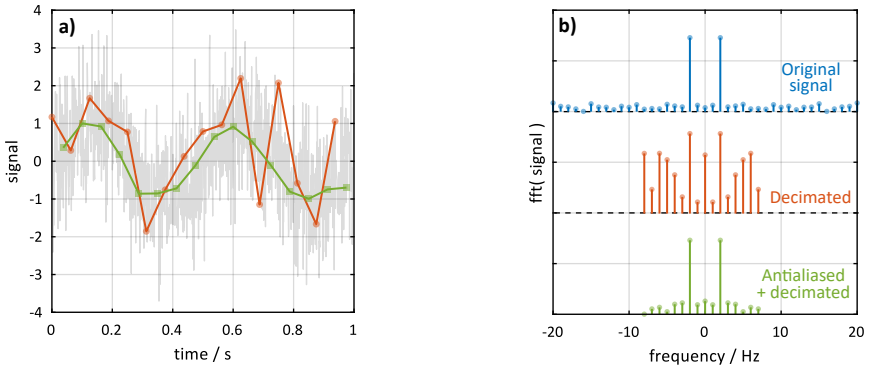


Figure 2.2: a) A single noisy tone signal, downsampled with (green) and without (red) anti-aliasing filter. **b)** Frequency spectra of the original signal (blue), and the same signal downsampled with (green) and without (red) anti-aliasing. Note that, since the downsampled signal as lower sampling frequency, the frequency spectra of the downsampled signals have a lower Nyquist frequency.

Frequency axis

The `fft` algorithm is oblivious to the actual real-world frequencies of the signal it processes. This is because the DFT equation as stated in 2.7a is, independent of the value of δt. The frequency axis of F has to, therefore, be constructed independently of the `fft` algorithm, by using the knowledge of the sampling time δt. This is somewhat involved, partially due to the inner workings of the `fft` algorithm, and partially due to minutiae of the DFT itself.

Firstly, in order to be able to represent all the possible phases (complex angles) of the oscillations comprising the signal, the DFT makes use of the negative frequencies. For DFT of real signals the *magnitude* of the `fft` in positive and the negative frequencies is identical but the *phase* (complex angle) is opposite. To put it otherwise, the real component of the `fft` is symmetric and the imaginary component is antisymmetric with respect to the zero-frequency. The zero frequency is often referred to as DC. It is a term originating from electrical engineering – 'direct current', and it represents the vertical offset of the signal. The rest of the spectrum is correspondingly referred to as the AC – 'alternating current'.

Secondly, there is a difference between an odd and even number of samples N of the DFT. The N roots of unity used to represent the signal (see equation 2.4) are uniformly distributed around the unit circle, and the 0^{th} root which represents the DC component is always present (see Figure 2.3c). This means that the root lying exactly opposite the DC (the Nyquist frequency) is present only when N is even. The frequency axis has to therefore be sampled differently based on the parity of N. The demonstration of this can be seen in Figure 2.3a and 2.3b where a 41 point signal is shown not to have the Nyquist frequency bin.

Thirdly, as the number of sampled points (N) increases, the amplitude of the DFT (as stated by equation 2.7a) goes to infinity as it approaches the delta function. In order to get an amplitude spectrum, the output of the `fft` has to be divided by N (the factor of $1/N$ has to be explicitly brought back from the definition of `ifft`). The details of the `fft` amplitude measurements will be elaborated on further, in Section 'Frequency and amplitude measurements'.

Finally, the output of the `fft` is shuffled such that $\lfloor N/2 \rfloor$ points on the left are swapped around with the $\lceil N/2 \rceil$ on the right. In this way the DC point always ends up as the first element of the resulting vector.[bn] The `fft` output can be put back the usual way with the use of `fftshift()`. Note though that after the `fftshift()`, the DC will not necessarily be the middle sample. For even values of N the middle sample does not exist and the DC will end up at $\lceil N/2 \rceil^{\text{th}}$ sample. The 1^{st} sample will then be the Nyquist frequency.

The frequency axis for an `fft` can therefore be constructed as follows:

[bn] This is just how the output of the `fft` ends up, and it is not shuffled back for the sake of speed. `ifft` takes its input in the same, shifted form.

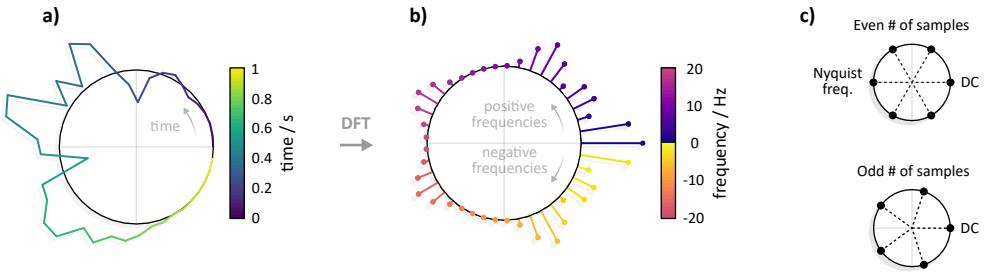


Figure 2.3: Circularly periodic view of a signal from the perspective of Fourier transform. **a)** Time-domain signal plotted on a circularly periodic axes. This signal has been windowed, and hence first and last points are both 0, and the entire signal matches up to itself after a full revolution. **b)** DFT of the signal from a). The colour scheme matches the typical output of an `fft()` algorithm, in which the left and right halves of the spectrum are swapped around. **c)** Roots of unity for even and off number of points. The DC sample lies exactly on $1+0i$ since $W_0 = \exp(0) = 1$. Note that the bin corresponding to Nyquist frequency does not exist when the number of sampling points is odd. The spectrum in b) has 41 points (odd), and hence no Nyquist bin.

```
y = ⟨signal⟩; t = (0:(N-1)) * dt; % Time axis
Y = fftshift( fft(y) )/N;
v = linspace(0,1-1/N,N) + (mod(N,2)/N-1)/2; v = v/dt; % Frequency axis
```

Listing 2.4: Construction of the fft frequency axis.

The frequency axis is made by taking a uniformly spaced vector from 0 to 1 excluding the final point which corresponds to the N roots of unity. It is then shifted such that the $\lceil N/2 \rceil^{\text{th}}$ sample becomes zero, and divided by the time-sampling period. If only half of the frequency axis is needed (e.g. for peak amplitude measurements), it can instead be constructed as follows:

```
y = ⟨signal⟩; t = (0:(N-1)) * dt; % Time axis
Y = fft(y)/N;
Yh = Y(1:ceil((N+1)/2)); % Half spectrum
Yh(2:ceil(N/2)) = Yh(2:ceil(N/2))*2; % Adjust amplitude of middle samples
vh = linspace(0, (1-mod(N,2)/N)/2, ceil((N+1)/2)); vh = vh/dt;
```

Listing 2.5: Construction of the fft half-frequency spectrum and axis.

The amplitude of the half-spectrum had to be doubled since it is normally distributed over two halves of the frequency axis. The DC and Nyquist samples are, however, unique and should not be doubled in any case. For real y, this half-spectrum is still, technically lossless,[bo] since `y` can still be reconstructed from `Yh` due to symmetries of the DFT:[bp]

```
Yl = Yh; Yl(2:ceil(N/2)) = Yl(2:ceil(N/2))/2; % Left half
Yr = fliplr( conj( Yl(2:ceil(N/2)) ) ); % Right half
Y2 = [Yl,Yr]; % Reconstructed two-sided fft
y2 = real(ifft(Y2*N));
```

Listing 2.6: Reconstruction of signal from single-sided spectrum.

Note that the length of the measured signal N translates to higher frequency resolution $\delta\nu$, and shorter sampling time δt (higher sampling rate ν_s) translates to higher frequency range. This is related to time-frequency uncertainly principle where both frequency and location in time cannot be known simultaneously and precisely. An infinite single-tone will have a precise, well defined frequency but will not have a well-defined position in time. By converse a very short, seemingly single frequency pulse (e.g. a light packet from a pulsed laser) will in fact have a very broad frequency spectrum.

[bo] Sans a single bit of information – the parity (sign) of N.
[bp] This technique is used, for example, in single-sideband modulation (SSB) radio.[46]

Fast Fourier transform

It is almost impossible to talk about Fourier transforms and not spend a moment on the most ubiquitous algorithm for computing it. Any real world calculation of the Fourier Transform will be done with the aid of an algorithm developed by James Cooley and John Tukey in 1965.[47]

The fast Fourier transform (FFT) algorithm relies on the fact that the computation of an N-point Fourier transform can be done by splitting the work amongst two interwoven $N/2$-point transforms.[48] These calculations are independent and so can in turn be split into two $N/4$-point ones; and so on recursively.[bq] This is in contrast to a naïve approach of looking at the overlap of the input function with each of the possible oscillations (basically eq. 2.2). For a length N signal the naïve approach takes roughly N^2 operations, whereas the FFT algorithm takes only $N \times \log_2(N)$. This difference amounts to orders of magnitude increase in speed for values of N as small as 200 (see Figure 2.4). Moreover, the numerical roundoff errors which always occur in floating point arithmetic tend to cancel out much better when using the FFT algorithm. This becomes important for applications where repeated use of the `fft` is required, as well as for large N. This can be seen with a simple test:

```
y = @(x) cos(2*pi*2*x); % 2Hz signal
Ns = unique(round(10.^linspace(1.5,4,200))); % Sampling points
rmsi = @(x) sqrt(median(imag(x).^2)); % Root median square imaginary part
for Ni = 1:numel(Ns)
    N = Ns(Ni); t = linspace(0,5,N+1); t(end) = [];
    % Fast Fourier transform
    tic, fast = fftshift(fft(y(t)));
    t_fast(Ni) = toc*1e3; d_fast(Ni) = rmsi(fast);
    % Naive approach
    dt = t(2)-t(1);
    v = linspace(0,1-1/N,N) + (mod(N,2)/N-1)/2; v = v/dt; % Frequency
    tic, naive = sum(bsxfun(@times,y(t),exp(2*pi*1i*v'.* (t-t(1)))),2)';
    t_naive(Ni) = toc*1e3; d_naive(Ni) = rmsi(naive);
end
```

Listing 2.7: Timing and accuracy benchmark of the FFT algorithm.

The input signal has been chosen as a cosine wave with an exact integer multiple of oscillations in the sample period (see Section 'Fencepost problem'). This means that, in theory, the output ought to be purely real, with 2 single-sample peaks of height $N/2$ at exactly ±2 Hz. Since the FFT treats the input `x` as complex with zero imaginary component, the roundoff error can also appear in the imaginary part of the output. This has been used in the code above to estimate the error in the Fourier transform as the root-median-square of the imaginary component of the output.

Fencepost problem

One of the common pitfalls of Fourier analysis has to do with what is commonly termed the 'fencepost problem'.[50] The problem can be illustrated with a simple example: if one is to put up 30 m of straight fence with posts 3 m apart, they need 11 posts and not, as one could initially think, 10. This is because each *section* needs a *fencepost* on its left-hand side except for the right-most one which needs an additional post on the right-hand side. If we sample N points of a signal of interest with sampling period of δt, then we have therefore sampled only a span of time equal to $(N-1)\delta t$. This, however, is not what the `fft` 'sees'. An important implicit feature of the DFT is that it assumes the signal to be periodic, such that the point *after* the last one is taken to be the first point. This is like the fencepost problem, but when the fence is, instead, to be built in a circle and the left-hand side of the first section serves as the right-hand side post of the last. The two ways of sampling can be easily programmed as follows:

[bq] This in principle works only for N which is a power of 2. There exist other versions of the FFT algorithm working on the same principle which allow for any value of N. This 'coarse to fine' approach is very common in algorithmic science.

[br] Such behaviour is, for example, the power management software slowing down a CPU to preserve battery usage or the interrupt handler processing giving the priority to the web-browser thread.[49]

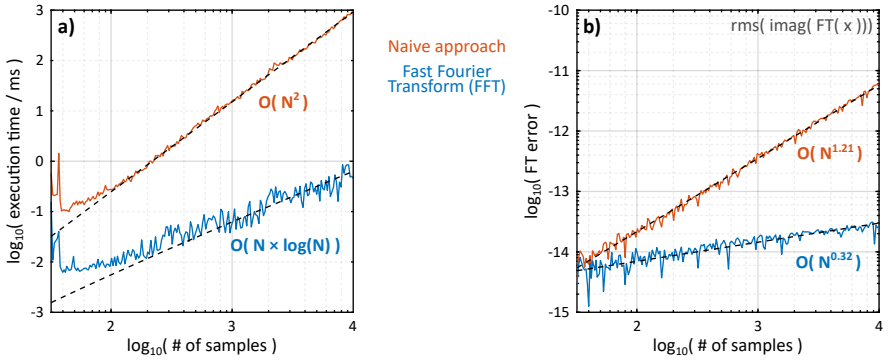


Figure 2.4: Performance of the FFT algorithm, as calculated in listing 2.7. The FFT can be seen to be **a)** faster and **b)** more accurate than a naïve approach of calculating overlap integrals. The benchmarks are fitted wit lines and the respective orders functions are given on the plots. Note that for low number of samples both methods deviate from their order functions. This is because the at small number of samples the algorithms are not limited by their processing speed but the data transfer overhead and manipulation time. The noise in the data occurs due to hardware-dependent behaviour.[br]

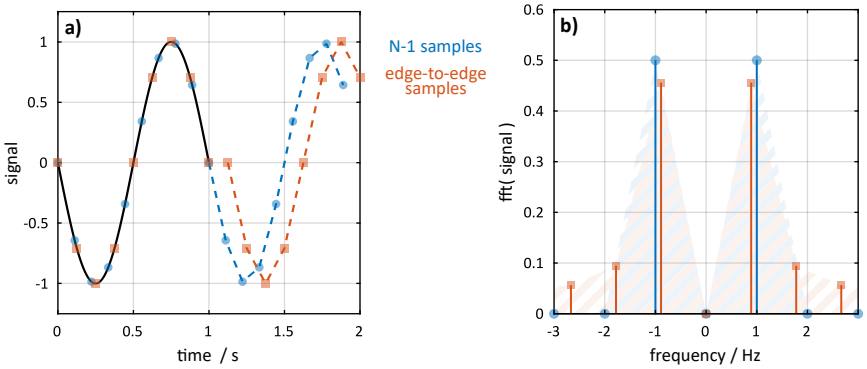


Figure 2.5: Correct ($N-1$, blue) and incorrect (edge-to-edge, red) sampling of a single tone signal. **a)** Since the `fft` assumes the signal to be periodic (see Figure 2.3a), concatenating together 2 copies of the signal allows one to get a better idea of what does the `fft` actually 'sees'. The $N-1$ sampled signal matches with itself, while the edge-to-edge one has a sharp step, at least from the perspective of the FFT. **b)** Frequency spectra of the signals from a). Since the signal frequency is an integer multiple of the sampling period, the spectrum ought to be a height ½ peak at ±1 Hz. This is indeed the case for the $N-1$ sampled signal, while the sharp step in the edge-to-edge sampled signal gives rise to high-frequency signals, and apparent drop in the frequency of the observed signal.

```
1 N = 7; % Number of sampling points
2 y = @(x) -sin(2*pi*1*x); % 1Hz Signal
3 t1 = linspace(0,1,N+1); t1(end) = []; % N-1 timesampling
4 t2 = linspace(0,1,N); % Edge-to-edge tiesampling
```

Listing 2.8: $N+1$ and edge-to-edge timesampling of a signal.

The signal `y` sampled at both of these time axes as seen by the `fft` has been plotted in Figure 2.5. With improper time sampling, the frequency of the oscillation (the main peak) is perceived to be lower than it really is. Moreover, there are additional high frequency components which originate from the sharp corners which arises in the improperly sampled signal. The difference is subtle but important, especially when small frequency components or shifts are to be measured.

Frequency and amplitude measurements

A common practice in signal processing is to use window functions – functions which multiply the signal in the time-domain before it goes though the DFT. The window functions are usually unit peak amplitude, symmetrical and close to zero at the edges. The use of windows allows one to mitigate the signal edge effects

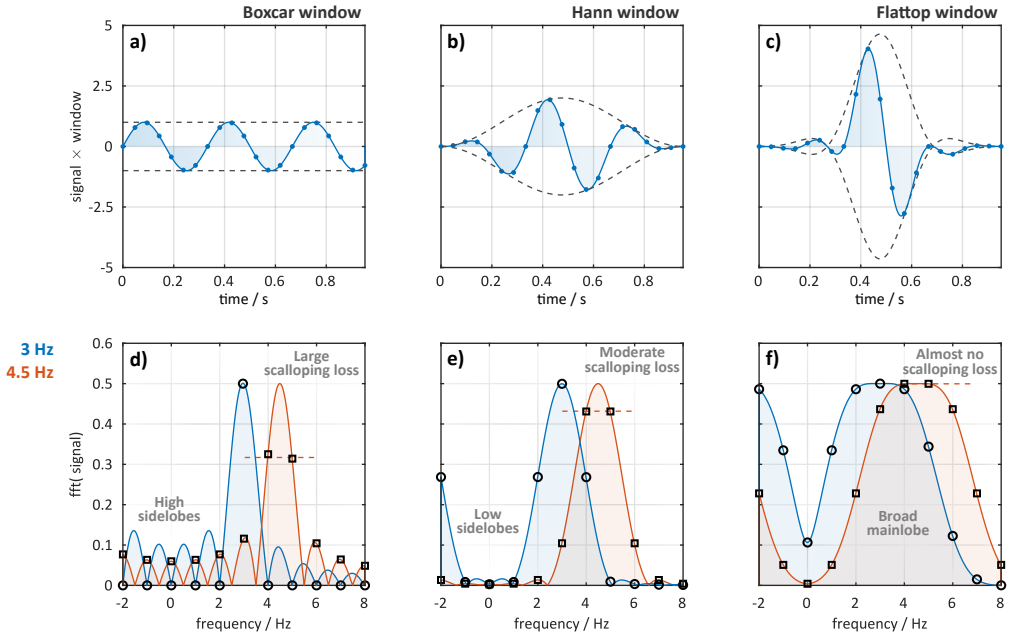


Figure 2.6: Effects of windowing of 3 Hz and 4.5 Hz signals on scalloping and sidelobe magnitude. The 3 Hz windowed signals are shown for the **a)** boxcar, **b)** Hann and **c)** flattop windows. The frequency spacing is set up such that the 3 Hz signal falls exactly on a bin of the DFT, whereas the 4.5 Hz signal falls exactly between the bins. **d)** The boxcar window performs poorly in terms of both scalloping losses and sidelobe magnitude. **e)** The Hann window drops the sidelobes but still shows some scalloping on the main signal peak. This is a good general-purpose window. **f)** Flattop window has, by design, almost no scalloping losses, but it is very wide in frequency. The signal can, therefore, be seen to partially overlap with the neighbouring –3 Hz peak.

of truncated signal as described in the previous section, and is used in signal frequency and amplitude estimation.

The amplitude of the signal has to be adjusted when applying a window such that the amplitude of the components of interest stays constant. This can be done by multiplying the signal by the widow, and then by the scaling constant s:

$$s = \frac{\int_\Omega I \, \mathrm{d}t}{\int_\Omega \mathrm{d}t} \tag{2.8}$$

where Ω is the region of the entire window. For an N-sample window, taking the mean of value of the window as the value of s is broadly equivalent. For many windows though, this integral can be done explicitly.

When the signal of interest has an integer multiple of oscillations in the sampling time, its DFT will be a single peak. This, however, is only because the other samples of the DFT will fall in the troughs of the sideband ripple of the peak. The DFT sample position is fixed: N samples between $\pm\nu_n$. If, the signal is a non-integer multiple of oscillations, its peak will therefore not be sampled at its peak and will therefore appear distorted (see Figure 2.6 a,d). This is most pronounced for when the length of the acquired signal is en exact multiple-plus-half of the period of the oscillation. The peak will then fall exactly half-way between the DFT bins. This peak distortion is referred to as 'scalloping'.

The most commonly used windowing function is the Hanning (Hann) window (`hann(N)` in Matlab). It is a single lobe of the $\cos^2(x)$ function, appropriately scaled and shifted. Hann window strikes a good balance between maintaining high frequency resolution while decreasing the scalloping losses. This prevents the signal interfering with its aliased shoulder and allows the components of the signal closer to the Nyquist frequency ν_n to be more reliable (see Figure 2.6 b,e). The Hanning window is a good general-purpose windowing function, but there exist specialised windows for specific applications.

The flattop window (`flattopwin(N)` in Matlab) is designed especially for amplitude estimation, and should be used whenever exact peak magnitudes need to be read. This window is relatively narrow in time, and therefore broad in frequency. The FT of the flattop window has a broad main lobe such that for all cases at least the top 2 samples have almost equal magnitude.[bs]. This minimises scalloping losses to almost negligible and allows for correct amplitude estimation, but at the price of poor frequency resolution (see Figure 2.6c and 2.6f) A simple simulacrum of the flattop window will be constructed in Section 'Global vs local convergence' with multi-target global optimisation.

The Gaussian window is the usual window of choice for applications where a precise frequency measurement is required. This window (`gausswin(N)`) is exactly that – a Gaussian lineshape with unit amplitude which is shifted such that its maximum is in the middle of the sample. The usual choice of the width is such that the window spans 3 standard deviations either side of the center. The reason this window is most used in frequency estimation, is because the FT of a Gaussian lineshape is also a Gaussian. This means that the resulting peaks, regardless of the scalloping losses, can be fitted with a Gaussian to estimate the center peak frequency.[bt] This is not true for most other windows, where the shape of the peak will change depending on scalloping.

Lastly, the Hann-Poisson window is an interesting case of a window with no side lobes. The main lobe of the FT of this window is a Lorentzian with ever-falling wings. This is especially useful for when the FT of a signal contributes to the hypersurface in an optimisation algorithm (see Section 2.2). Since there are no sidelobes, there is no local minima for the algorithm to potentially get stuck on.

With the content of the above in mind, a Matlab function to take a one-sided FT of a signal and construct the accompanying frequency axis is shown in listing A.3.

Convolution theorem

Closely related to the idea of a FT is that of *convolution* – an integration of a product of two functions f and g as a function of the shift of one of the functions. The result h is a new function of the relative shift. This can be put formally as:

$$\int_{-\infty}^{+\infty} f(\tau)g(t-\tau)d\tau = f*g = h(t) \tag{2.9}$$

where the $*$ is commonly used as a symbol for convolution, and τ has been used as a dummy variable of integration. This operation can be thought of as taking one signal, flipping it left-to-right and sliding it past the other one. At each time point, the result is the integral of the overlapping area of two signals. It is most common to talk about one signal called the 'kernel' acting on another 'input' signal to produce an output. The kernel is usually a narrow function (in relation to the input), and the output can be thought of as a modifier version of the input. This can be used to model many phenomena. For example, a boxcar kernel will model the effect of the out-of-focus lens blur,[bu] while an exponential kernel can be used to model the effects of the instrument response function of a detector.

The convolution theorem itself states that the Fourier transform of a product, is the convolution of the Fourier transforms:

$$\mathrm{FT}\left\{f(t)\times g(t)\right\} = \mathrm{FT}\left\{f(t)\right\} * \mathrm{FT}\left\{g(t)\right\} = F(\nu) * G(\nu) \tag{2.10}$$

where F and G are the Fourier transforms of f and g respectively.

[bs] Flattop window can be constructed by linear combination of 4 single frequency oscillations with arbitrary scaling constants, followed by numerical optimisation to maximise the flatness of the top of the main lobe.

[bt] Note that, in frequency estimation, any shifts of the frequency of the signal of interest, as well as the instability of the sampling frequency clock play a role.

[bu] Technically the kernel for the out-of-focus lens blur (Bokeh blur) is a disc and the operation is a 2D convolution with an image, but in the case of a 1D image this reduces to a boxcar function.

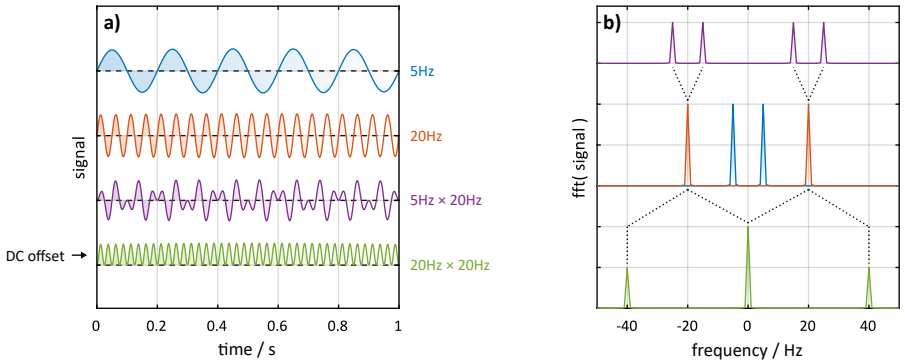


Figure 2.7: Effect of convolution of two single-tone signals. **a)** Two pure tones: 5 Hz and 20 Hz and two pairwise mixed tones: 5×20 Hz and 20×20 Hz. Note the 0 Hz component (DC offset) of the 5×20 Hz signal. **b)** Frequency spectra of the signals shown in a). The mixing of two tones can be seen to give rise to sidebands of one of them on the frequency of the other one. For the self-mixing case this leads to the 0 and double frequency components.

Convolution of discrete finite signals (vectors) is, analogously, defined by the sum over the product of the elements of the zero-padded vectors. This means that for a general case of two vectors of length n and m, such that $m \leq n$, the resultant vector will have length $n+m+1$, but only the middle $n-m+1$ entries will be 'valid' in as much as the zero padding will not contribute to their value. Because of that one often talks about a *circular* convolution where, as opposed to the zero padding, the signals in question are thought to wrap around and effectively get padded out with copies of themselves. The convolution theorem still holds for the case of DFT if the operation of convolution is replaced with its circular variant (denoted by $\circledast$):

$$\mathrm{DFT}\{f \times g\} = \mathrm{DFT}\{f\} \circledast \mathrm{DFT}\{g\} = F \circledast G \tag{2.11}$$

The convolution theorem for DFT can be easily verified with the following:

```
1 N = 1e3; x = randn(N,1); y = randn(N,1); % Two random signals
2 XY1 = fft(x.*y); % DFT of a product
3 XY2 = cconv(fft(x),fft(y),N)/N; % Circular convolution of DFTs
4 delta = max(abs(XY1-XY2))/(N*eps); % = 1/4; Max difference
```

Listing 2.9: Convolution theorem verification

The `delta` turns out to be approximately 1/4 of `N*eps` – the expected difference based purely on the number of the floating point operations.

One of the phenomena which benefits greatly from being considered in terms of a convolution is that of signal mixing. When two frequencies f_s and f_c (with $f_s \leq f_c$) mix, they get multiplied together in the time domain. By a series of trigonometric identities, one can see that the result is a the sum $f_s + f_c$, and the difference $f_s - f_c$ of two frequencies. This is not obvious though, unless once considers that mixing of two signals is the convolution of their FTs. If one considers the spectrum of the f_s signal – two peaks of magnitude 0.5 at $\pm f_s$, sliding past the corresponding spectrum of the f_c signal, one can easily see that the result will be two *satellite* peaks of magnitude 0.25 at $\pm f_s$ (see Figure 2.7). The lower frequency component is usually referred to as the *baseband* signal (hence subscript 's'), and the higher one as the *carrier*. Mixing with high frequency carrier does not actually change the information content of the signal, but merely shifts it up to higher frequency range.[bv] The process of mixing with a carrier is often referred to as amplitude *modulation* (AM).[bw] Since modulation produces frequency components at the sum and the difference frequency, half of any component of the signal at the carrier frequency will end up at DC, and the other

[bv] This is the basis of operation of an analogue radio. The audio signal is mixed with the carrier frequency and broadcast into the ether. The receiver is 'tuned' onto a station when its internal oscillator matches the stations' carrier frequency, and therefore is able to demodulate the audio signal. For example, the carrier frequency of BBC Radio Oxford is 95.2 MHz.

[bw] There exists a plethora of modulation schemes. Starting from the most obvious amplitude (AM), frequency (FM) and phase modulation (PM) and ending at pseudorandom noise modulation (PNM) and frequency hopping spread modulation (FHSM).

half at twice the carrier frequency (see Figure 2.7).[bx] This second mixing can be used to *demodulate* the signal. All one needs to do, in fact, is to reject the $2f_c$ component. This can be done with an application of a low-pass (LP) filter.

The reason one might consider modulating and then demodulating their signal, is due to presence of additive noise. The baseband signal may be, for example, measured fluorescence intensity (DC signal), and the noise may be the drift in the signal due to sample degradation or the alignment of the optics, oscillations in the (flowed) sample's fluorescence due to the action of the pump or light scattering by air bubbles in the sample volume (see Chapter 3). Notice that most of the sources of noise are either low frequency (drifts) or periodic (action of the flow pump). Nominally, the signal of interest is a DC signal, and therefore strongly influenced by the low frequency noise. A careful choice of carrier would, however, allow one to move the signal away from most of the sources of noise, and subsequent demodulation would allow one to recover the original signal of interest. Notably though, the modulation would have to be done before any source of additive noise; otherwise it would have no effect. In this case this is (mostly) possible by modulation of the fluorescence excitation light. This can be done successfully so long as the fluorescence intensity, all other phenomena of interest (e.g. the magnetic field effect) as well as the detector itself, respond to the modulation of the excitation light fast enough.[by] This modulation will take action before some of the noise components will degrade the signal and therefore result in a higher signal-to-noise (S/N) after the demodulation. Note that, in this case, the modulation is not a mixing with a pure carrier, but with a carrier plus a DC component.[bz] This will end up not mattering as long as the demodulation is done with a purely-AC carrier (AC coupled).

An example of this type of modulation/demodulation scheme can be easily shown with the following:[ca]

```
N = 400; t = linspace(0,1,N+1); t(end) = []; dt = t(2)-t(1); % Time axis
v = linspace(0,1-1/N,N) + (mod(N,2)/N-1)/2; v = v/dt; % Frequency axis
w = hann(N)'; w = w./mean(w); % Hann window
signal = sin(2*pi*t*5); % 5 Hz baseband signal
carrier = sin(2*pi*t*80); % 80 Hz carrier
noise = 1.5*randnd(-2,1,N)+0.5; % Random pink noise
unmodulated = signal + noise;
modulated = signal.*carrier + noise;
demodulated = modulated.*carrier;
[b,a] = butter(6,10*dt*2); % 6-pole IIR Butterworth filter, 10 Hz cutoff
lpfiltered = filtfilt(b,a,demodulated).*2; % Zero-phase filtering
[h,~] = freqz(b,a,N,'whole'); % Calculate the filter frequency response
filterwindow = fftshift(abs(h)); % Shape of the filter window
```

Listing 2.10: Modulation/demodulation of a signal.

In this example a 5 Hz signal is recovered from low frequency noise with a use of an 80 Hz carrier. The carrier frequency *and phase* have to be known a priori in order to recover the signal correctly. The low frequency (pink) noise generation is done with a non-standard Matlab function `randnd` which can be found in Listing A.5.[51] Figure 2.8 shows the (windowed) time and frequency profiles of the `unmodulated`, `modulated`, `demodulated` and `lpfiltered` signal, as well as the original `signal` itself.

The filtering in the example above is done with a simple Butterworth LP filter. Proper filter design and implementation is very involved and beyond the scope of this work. The specifics of the filter design

[bx] The low frequency components in the self-demodulated (bandlimited) signal can be considered the signals' envelope. The frequency of the envelope is sometimes referred to as the *beat* frequency. This beating can be often observed when when two high frequency which ought to be identical, are not. The small difference between them creates a low-frequency beat which can be easily observed (e.g. grids between the overhead signs on the motorway) or heard (e.g. cooling fans in two computers).

[by] The delayed response of the system in question to modulation can, also, be leveraged as a diagnostic for these effects. For example, the Enhanced Magnetic Field Effect (EMFE) on fluorescence intensity could be measured by varying the frequency of the magnetic field modulation and measuring the magnitude of the resulting signal.[2]

[bz] The fluorescence does not get modulated from +1 to –1 of an arbitrary unit since light intensity cannot be negative. At most it can be modulated between +1 and 0. Hence the additional DC component in the carrier.

[ca] The word 'modem' – a networking device used for data encoding and decoding, is a portmanteau of *mo*dulation and *dem*odulation, since historically that is indeed exactly the process used.

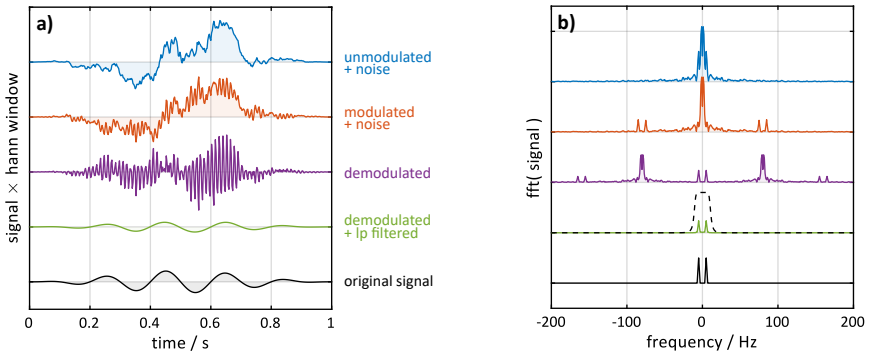


Figure 2.8: Demonstration of signal modulation and subsequent, and its relationship to low-frequency noise influence of noise. **a)** Time traces of example signals, multiplied by a Hann window. It is almost impossible to distinguish the original signal (black) within the plot of same signal corrupted by low-frequency noise (blue). If the original signal is modulated by a high frequency carrier (80 Hz) prior to the addition of the noise (red), demodulated (purple) and LP filtered (green), the original signal can be recovered. Compare the recovered signals' magnitude and shape with the original signal. **b)** Frequency spectra for the corresponding timetraces in a). The low-frequency noise band can be seen to be modulated up to the carrier frequency (purple) and then rejected by the Lp filtering (green). Note that half of the signal power also gets unconverted by 80 Hz (red→purple) and therefore lost in the LP filtering step.

depend heavily on the application and implementation (digital signal processing or an analogue circuit). Briefly, when talking about filters (both electronic and optical) one talks about low-pass (LP), high pass (HP), bandpass (BP) and bandstop (BP) or 'notch' filters. They, respectively reject high, low, all-but-a-band and just-a-band of frequencies. The features of interest when considering different filters are the *rolloff* – how fast does the particular filter transition from one regime to the other, and the *passband* and *stopband ripple* – the characteristic flatness of the transparent and opaque spectral regions of the filter respectively. This, of course, in addition to cutoff frequencies/wavelengths appropriate for the filter in question.

Electronic filters can be categorised by how long their output remains non-zero after a unit impulse input. There are finite (FIR) and infinite impulse response (IIR) filters. All analogue filters are, at heart IIR since, once applied, the input forever lingers within the residual current of the filter. The digital filters in turn can be designed to be of either IIR or FIR type.

Analogue filters are used when certain fixed filtering operation needs to be performed. They do not require any computational power, and sometimes they do not require power at all (passive filters). Analogue filters are however slave to the IIR design which means they inherently require certain form of feedback which can lead to instability (e.g. a constructive resonance in an RLC circuit). Moreover, their phase response can be very nonlinear in frequency. This means that they introduce complex lag into the signal which has to be considered in certain applications. Such analogue electronic filters are usually implemented with cascades of resistor-inductor-capacitor (RLC) circuits (passive filters) or with operational amplifier circuits (active filters).[cb]

Digital filters are very flexible, and are an excellent choice when the signal in question is already digitised. They are, in fact, often a good enough choice to warrant an analogue-to-digital (ADC) converter, followed by a microprocessor to implement the filter followed by an digital-to-analogue converter. Such filters are however obviously limited to the processing speed of their floating point chip and the speed of the ADC and DAC converters.[cc] Digital filters can also implement the FIR architecture as it is not possible to ensure that the output of the filter goes *exactly* to zero after certain time away from an impulse. Due to lack of feedback FIR filters are inherently always stable, and can achieve excellent phase response. They are however much

[cb] Optical filters are created mostly by doping a substrate (usually glass) with a chemical species which absorbs light in a specific band (absorptions filters), or by deposition of thin dielectric layers with varying indices of refraction on the surface of the substrate. This causes selective interference extinction of the incident light (interference/dichroic filters).

[cc] There exists a family of integrated circuits known as the field-programmable gate arrays FPGA. The internal structure of the logic gates of these circuits can be reprogrammed to implement, for example, a digital filter. FPGAs find their uses in applications where there is a need for both flexibility and processing speed.

more computationally expensive which is why digital IIR filters are also used. If the filtering is not done in real-time (RT), the problem of complex phase response can be addressed with so called zero-phase filtering (`filtfilt()` in the code above).

An important observation is that the action of a filter is to, effectively, multiply the signal in the Fourier domain by some window (see Figure 2.8). This means that it can be expressed as a convolution of the corresponding time signal with iFT of the filter window. The narrower the filter window, the wider its iFT. Filtering a signal by simply setting the values of the FFT bins to 0, is equivalent to convolving it in the time domain by a $\mathrm{sinc}(x) = \sin(x)/x$ which, in turn, has a long ripple. This can lead to signal distortion and proper carefully designed filters ought to be used instead.

Gabor transform

The signal of interest will not always consist entirely of the same information throughout. Sometimes the frequency content of interest will be localised in time to some part of the signal. If its location (in time) is known, it can be simply trimmed and then analysed. This is, however, not always possible, especially when analysing an unknown or complicated signal. The concept of windowing can be leveraged to analyse such signal. If the window is made narrow, and allowed to shift along the time axis, the resulting signal could be analysed as a function of its position and location of specific frequency components could then be obtained. This is a variant of the FT and is sometimes referred to as the *Gabor transform*. It can be formally put by a small modification of eq. 2.1

$$F_G(\nu, \tau \,|\, d) = \int_{-\infty}^{+\infty} \exp\left(-4\log(2)\left(\frac{t-\tau}{d}\right)^2\right) f(t) \exp(-2\pi i \nu t)\, dt \tag{2.12}$$

where d is the full-width-at-half-maximum (FWHM) of the sliding Gaussian window. F_G is now a function of both the frequency components of the signal as well as the time shift of the window. There is also an implicit parameter – the width of the Gaussian window used. On that matter, any other scaled window (Hann, flattop, etc.) which can be slid across the signal could be used instead.

A narrow window has a well defined region but equally results in a broad frequency signal. A wide window can result in sharper frequency response but less temporal signal localisation. The choice of the window width will therefore affect the extent of the time $\leftrightarrow$ frequency resolution trade-off. Simple Gabor transform (in this case for two different widths of the window) can be programmed as follows:

```
N = 201; t = linspace(0,1,N+1); t(end) = []; % Time axis
% Make an interesting signal with changing frequency content
signal = sin(2*pi*[t(t<1/3)*20 t(t>=1/3 & t< 1/2)*60 t(t>=1/2)*40]);
signal = signal.*(sin(2*pi*t*3).^2) + 0.2*randn(size(signal));
% Gabor transform
d1 = 0.05; d2 = 0.4; % Widths of the window
w = @(c,d) exp( -(4*log(2).*((t-c)./d).^2 )); % Gaussian window
ts = linspace(0,1,N); % Time shifts
for ti = 1:numel(ts)
    Y = fft( signal.*w(ts(ti),d1) )/N; YY1(ti,:) = Y(1:ceil((N+1)/2));
    Y = fft( signal.*w(ts(ti),d2) )/N; YY2(ti,:) = Y(1:ceil((N+1)/2));
end
```

Listing 2.11: Gabor transform of a signal with changing frequency content

A signal with distinct periodic frequency packets has been chosen. The results are plotted in Figure 2.9. A window narrower than the width of a packet gives a distinct frequency peak for each of the packets, but results in relatively poor frequency resolution. A choice of wider window allows better frequency discrimination but, because this window effectively averages two neighbouring frequency packets, it can miss single packets and it shows satellite peaks due to packet-to-packet amplitude modulation (see Figure 2.9c).

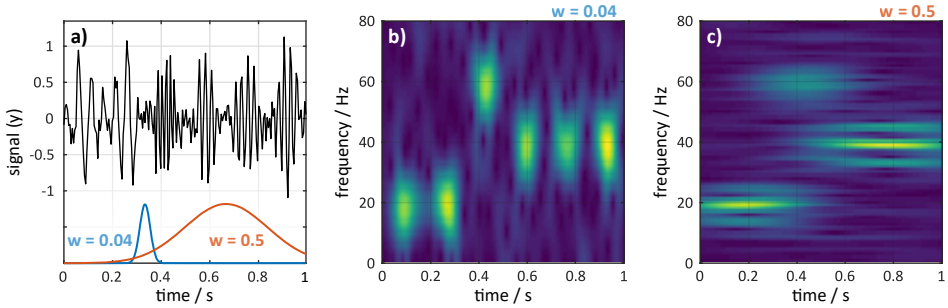


Figure 2.9: Gabor transform of a signal with changing frequency content, as calculated in Listing 2.11. The frequency bursts are at $2 \times 20\,\text{Hz}$, $1 \times 60\,\text{Hz}$ and $2 \times 40\,\text{Hz}$. **a)** Input signal with alternating bursts of high frequency. The two windows used are shown below for comparison. The width parameter (full-width at half-maximum) for each of the windows is indicated on the graph. **b)** Narrow-windowed Gabor transform. Peaks are well-localised in space, but have very broad frequency distribution, since each window 'sees' only a very short part of the signal. **c)** Broad-windowed Gabor transform. Peaks are well-localised in frequency, but very poorly localised in space. The single 60 Hz peak can be seen to almost disappear. Note the 6 Hz sidebands which is the frequency of the envelope of the wavepackets.

2.2 Numerical optimisation

A quadratic (a parabola) is a second degree polynomial which has a general form:

$$y = a\,x^2 + b\,x + c \tag{2.13}$$

Any set of coefficients $[a, b, c]$ *uniquely* defines such a parabola, and any point of interest on it can be found explicitly by deriving the appropriate expression in terms of the three coefficients. The two (sometimes complex) solutions to $y = h$ are:

$$x = \frac{-b \pm \sqrt{b^2 - 4a(c-h)}}{2a} \tag{2.14}$$

When $h = 0$ these solutions are called *roots*. The turning point (minimum/maximum) of the parabola can be found either by differentiation and setting the derivative to 0, or by realising that at the turning point there is only one solution, which means that the term $\sqrt{b^2 - 4a(c-h)}$ must be equal to zero. In either case the turning point coordinates in the $x\,y$-plane are:

$$(x_m\,,\,y_m) = \left(-\frac{b}{2a}\,,\,-\frac{b^2 - 4a\,c}{4a}\right) \tag{2.15}$$

This is an *analytical solution* for the turning point – given any set of coefficients $[a, b, c]$ one can use the above equation to immediately know its position. Such analytical expressions are incredibly useful as they are *exact*, *computationally fast* and often offer insight into the general behaviour of the point of interest. They, however, exist only for polynomials up to degree 4,[cd] as well as for *some* simple functions. A good example of a simple function without an analytical solution is the roots of $\text{tanc}(x) = \tan(x)/x$.[52] The minima of all other functions can instead be found with algorithmic methods to yield *numerical solution(s)* up to an arbitrary precision. This is sufficient for almost all scientific application since no fundamental physical constant is known to more than 12 significant figures, and most algorithm implementations are accurate to 16 by default.[53;54]

One might notice that an exact definition of what comprises a *solution* has not been given above. This is because the problems of *rootfinding* and *optimisation* can be easily related and, in practice, differ only marginally. *Provided a function has a root*, to find a root of a function is to find a minimum of its square (or absolute value, or in general its *norm*). If the function has no roots in the optimisation range, the optimisation will find the point where the functions' value is closest to zero.[ce] Similarly, provided the

[cd] The Abel-Ruffini theorem states that there is no *general* closed form solution solution for roots of polynomials of degree 5 and higher. It does not say that a *particular* polynomial (or even a family of polynomials) cannot be solved.

[ce] In general, to find a point where a function f *is closest* in value to b is to solve an optimisation problem: $\min_x ||f(x) - b)||_2$.

function has an inflection point, it can be found by finding the root of its derivative. The precise nature of the inflection point has to then be determined based on higher derivatives of the function in question.[cf] If the function has no infection points in the rootfinding range, its derivative will have no roots and precise nature of the termination of the rootfinding algorithm depends on its implementation.[cg] Minimisation and maximisation can also be easily related since to maximise f is to minimise $-f$ (or, otherwise, any *monotonically decreasing* transfer function on f with *negative gradient* over its entire domain).

Such methods will be explored in this section – the way they work and how they can be used to solve many problems which arise in both data analysis and experimental science. The areas which will be explored are: the choice of the algorithm, the question of local vs. global minima, regularisation and, once found, the uncertainty in the solution.

Problem structure

Finding function minima may seem exceptionally narrow in terms of applications, but it turns out that many problems can be phrased as finding a minimum of some kind of function. Protein folding and molecular geometry optimisation problems attempt to minimise the Gibbs free energy of the species of interest as a function of the position of its constituent components (and sometimes the solvent). The Schrödinger equation can be solved by minimising the distance between the solution and the boundary conditions as a function of some unknown parameters, as well as since the eigenvalue problem itself can be solved by optimisation. Furthermore, some highly efficient modern linear algebra methods for large-scale simulation (preconditioned conjugate gradient solvers) rely on an optimisation technique. A common application of optimisation is also to fit a model to a dataset. This is, conceptually, one of the simplest cases of optimisation to think about. Many examples in this section will, therefore, use fitting as the premise for use of the optimisation algorithms.

Data fitting can be thought of as minimisation of the energy of some kind of attractive potential function ϕ between the model and the data D, as a function of the model parameters $\boldsymbol{p}$. This function is referred to as the *loss function* l, and in the simplest case one can simply consider it to be equal to $l\left(\boldsymbol{p}\,|\,D\right)=\phi\left(\boldsymbol{p}\,|\,D\right)$. Note that $\boldsymbol{p}$ indicates a *vector* of parameters, such that for an n-parameter model the fitting procedure can be thought of as exploring a space spanned by this n-dimensional vector. For a 1-parameter model, this is a line, 2 parameters span a surface and, in general, n parameters span an n-dimensional manifold,[ch] usually referred to as the *hypersurface*, in an $(n+1)$-dimensional space. The exact shape of that surface and the chosen algorithms ability to explore it affect the quality, and even sometimes the validity of the fit, as well as the speed with which it is obtained.

Such a (hyper)surface can be calculated (for the purpose of illustration) by evaluating the loss function over a grid of parameter values.

```
m = @(x,w,h) (1./(1+(x/w*2).^2)-1).*h; % Generalized Lorentzian
x = linspace(-0.9,0.9,30);
y = m(x,1,5) + 0.5*randn(size(x)); % Data (model + gaussian noise)
phi = @(a,b) sqrt(sum( ( y-m(x,a,b) ).^2 )); % Potential function
lossfun = @(p) log( phi(p(1),p(2)) ); % Loss function
% Evaluate loss function for a grid of possible values of the parameters
ws = linspace(0,5,1e2); hs = linspace(0,10,1e2);
```

[cf] For ill-differentiable functions a *subderivative* may need to be employed. Consider a function $f(x)$ which is differentiable everywhere but at a point $x=p$. When $x\neq p$ the subderivative and the derivative are equivalent. At $x=p$ the derivative of f is not undefined, but its subderivative is any line $l(x)$ such that is coincident with the point $(p,f(p))$ and whose value is always less than or equal to f.

[cg] A well-behaved function f (one well approximated by its Taylor expansion) on the interval $[a,b]$ there will relationship between the parity of the signs of $f(a)$ and $f(b)$ and the number of roots the function can have. If the product $f(a)f(b)<0$, f will have an odd number of roots, if $f(a)f(b)>0$ the number of roots will be even (including 0 – no roots) and the root will lie at either (or both) ends of the interval. This is a corollary of what it referred to as a *fundamental theorem of algebra*, which can be summarised by the following statement: *'If a function value is positive at a and negative at b, then somewhere in between it must have crossed the $x=0$ line'*.

[ch] Note that n refers to the dimensionality of the parameter space, *not* the data.

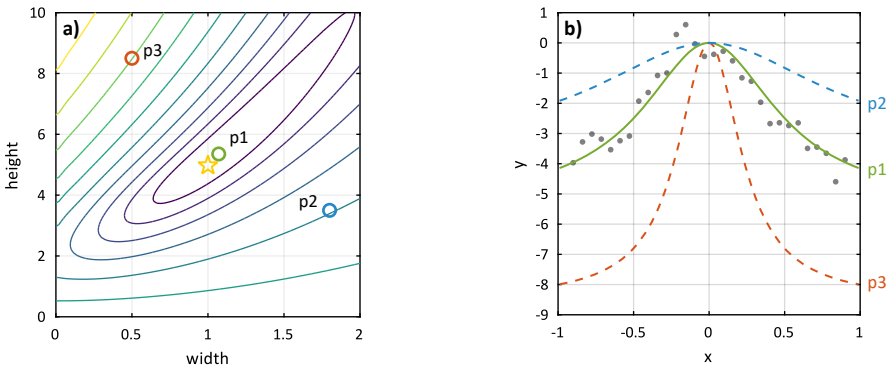


Figure 2.10: a) Lorentzian fitting loss function hypersurface, and **b)** selected fit functions, as calculated in Listing 2.12. The minimum of the loss function in the parameter space is shown in green. This is the best fit. The actual optimum is indicated with a star. Note that the well of the loss function does not align with the axes of the plot – basis vectors of the parameter space.

```
L = zeros(numel(ws),numel(hs)); % Preallocate
for ai = 1:numel(ws)
    for bi = 1:numel(hs)
        L(ai,bi) = lossfun(ws(ai),hs(bi));
    end
end
[~,I] = min(L(:)); [wI,hI] = ind2sub(size(L),I); % Find minimum
pf = [ws(wI) hs(hI)]; % Fitted parameters
```

Listing 2.12: Loss function hypersurface for a Lorentzian fit

The artificial data has been created by adding normally distributed random noise to the model. Only a 2-parameter model has been chosen such that the associated hypersurface can be easily visualised. The potential function has been chosen as the *2-norm* of the difference $\boldsymbol{\delta}$ between the data $\boldsymbol{y}$ and the model $\boldsymbol{m}$ and vectors:[ci]

$$\phi = ||\boldsymbol{\delta}||_2 = ||\boldsymbol{y} - \boldsymbol{m}||_2 = \sqrt{\sum_{i=1}^{n} \left(y_i - m_i\right)^2} \tag{2.16}$$

The loss function is, in this case, the log of the potential. Since log is a monotonically increasing function, the optimum of the $\log(\phi)$ is also the optimum of ϕ. The minimum $\boldsymbol{p}_f$ and the associated fits are shown in Figure 2.10.

This can, in fact, be considered to be the simplest method of optimisation: the *brute-force method* – just evaluate the function of interest over a grid of parameters and pick the best one. It is a simple method which does work great, but only for *well-behaved*, computationally *cheap* loss functions over a *finite domain* in *very low dimensional* space and when only *low accuracy* result is needed. Since the domain of $\boldsymbol{p}$ is sampled with finite number of samples m, l must be band-limited (no sharp and small features, see 2.1). The loss function is evaluated an exorbitant number of times (m^n) which makes this method take prohibitively long time for expensive (long evaluation time) loss function.[cj] The precision with which the position of the minimum is established is furthermore inherently restricted to how finely is the parameter grid sampled and whether the range over which it is sampled happens to encompass the position of the minimum. This is quite restrictive in terms of applications and most, if not all, fits are performed much more effectively with more sophisticated algorithms.

[ci] The other common choices are the 1-norm $||\boldsymbol{\delta}||_1 = \sum|\delta_i|$ and and the ∞-norm $||\boldsymbol{\delta}||_\infty = \max|\boldsymbol{\delta}|$. They differ mostly in how much do the small entries in the vectors contribute to the result (the most in 1-norm and the least in ∞-norm).

[cj] As the dimensionality of the space increases, the number of uniform sampling points needed to span the entire space increases exponentially. The higher the dimension the sparser the space. This can be seen by comparing the volume V of an n-cube to its subset, for example an n-ball. As n goes to $+\infty$ so does the ratio $V_{\text{cube}}/V_{\text{ball}}$. This makes a high dimensional space hard to explore and is often referred to as the '*curse of dimensionality*'. [44;55]

Every optimisation algorithm (solver) will end up evaluating l at some number of points. The loss function is therefore an inherent part of every problem. There is, however, one more vitally important component – the *starting 'point'*. This can be either explicit – literally a point (or a set of points) in n-space, or implicit – the range over which the initial 'point' is randomly generated or, as it is in the case of the brute-force method above, just a range of points over which to search. It is important to remember that every algorithm has to, and does, start somewhere. For very simple problems, this may not matter much, but much more commonly a choice of good starting point will affect the quality of the solution.[ck]

The loss function l and the *starting 'point'* are the two components inherent to any optimisation problem. They can be thought of as, and is in fact closely related to, the boundary conditions of a differential equation (see section 2.3). Depending on the solver one can also specify multiple types of constraints on the fit parameters, as well as the loss function. These can however be often incorporated into the loss function and are not part of the *general* problem setup.[cl]

Gradient descent

Instead of brute-force search, most optimisation problems are solved with iterative methods. Possibly the simplest one of such methods is the *gradient-descent*. The optimisation starts at some point $\boldsymbol{p}_0$ in $\boldsymbol{p}$-space, evaluates the gradient of the loss function l and takes the a downhill step of size $-\alpha l$ where α is the step size. This is repeated until some stopping condition, e.g. number of function evaluations or the step size, is met. In general (for the case when $\boldsymbol{p}$ spans more than one dimension), the 'downhill' direction is generalised by the Jacobi matrix $\boldsymbol{J}$ (matrix of first derivatives) of l:[cm]

$$\boldsymbol{J}(\boldsymbol{p}) = \left[\frac{\partial l}{\partial p_1}\ \frac{\partial l}{\partial p_2}\ \dots\ \frac{\partial l}{\partial p_n}\right] \tag{2.17}$$

A basic gradient-descent algorithm can be programmed as follows:

```
f = @(x,y) log(1 + abs(x).^(2 + sin(x+y))) + y.^2/20; % Exact function
J = @(x,y) [⟨∂f/∂x⟩ ⟨∂f/∂y⟩];
tol = 1e-4; alpha = 0.5; % Tolerance and step size
p = [8.3 -8.9]; % Starting point p0
for j = 1:1000 % Limit number of iterations to 1e3
    s = -alpha*J(p(1),p(2)); p = p + s; % Downhill step
    if norm(s)<tol; break; end % Stopping condition
end
```

Listing 2.13: Basic gradient descent algorithm.

The function `f` has a single (global) minimum at $f(0,0)=0$. As stated, the algorithm converges within ca 150 iterations to within the `tol` of 10^{-4}. This corresponds to equal number of evaluations of the Jacobian (note that the algorithm does not actually evaluate the function itself). The analytical Jacobian is rather complicated and so it was omitted here since its exact form is irrelevant.[cn]

[ck] Simple problems are, basically, those with single, well defined minimum and an attraction basin spanning the entire domain; for example numerically finding a minimum of x^2. Even then, depending on the algorithm and the language in which it is implemented, a particularly obtuse choice of the starting point could lead to a numerical overflow and halt of the algorithm.

[cl] For example if a certain parameter must be smaller than 7, the loss function can be programmed to return $+\infty$ (or just a very large value) if that parameter exceeds 7. Note that the solver used must be able to handle ∞-valued loss functions.

[cm] There can be a few slightly different formulations of the Jacobi matrix, depending on the exact implementation and context. Equation 2.17 is a 1D vector since l is scalar-valued. If l were to be a vector-valued function instead, $\boldsymbol{J}$ would be a matrix such that $J_{i,j} = {}^{\partial l_i}/_{\partial p_j}$, where l_j is the t$^{\text{th}}$ element of $\boldsymbol{l}$. In the context of numerical approximation of the Jacobian a scalar-valued l can also give rise to a 2D Jacobian since it will be evaluated on a grid on N points along x, such that $J_{n,j} = {}^{\partial l(x_n)}/_{\partial p_j}$. One ought to be careful to be certain about the exact form of the Jacobian matrix used. The same considerations apply to the Hessian, as well as higher order matrices.

[cn] Depending on the exact implementation of the function which return the Jacobian it may 1) be faster, 2) work or not on vector/matrix/complex inputs and 3) be well defined on the entire domain. The implementation used in the code above, e.g. fails if evaluated exactly at $(0,0)$ since it evaluated to $-\inf/0 = \texttt{NaN}$ while it ought to be $J(0,0)=0$. In practice function evaluating the Jacobian ought to deal with such special cases.

The output of the gradient-descent algorithm at each iteration has been plotted in Figure 2.11a. It is easy to see how the convergence might depend on the starting point as well as the step size α. The path the algorithm takes is akin to that a marble would take when placed at point $\boldsymbol{p}_0$ and moving under potential l.[co] The path oscillates back and forth in the valleys of l as the gradient descent If the staring point was in a different valley of f, or closer to $(0,0)$, the convergence could have been faster. If the value of α was lower, the oscillations in the valley of l would have been smaller, but so would the steps which go straight downhill.

To improve the speed of convergence for a specific family of problems, the value of α could be pre-optimised. This process is also optimisation problem where the runtime of the algorithm is optimised as a function of its parameters.[cp] Alternatively, the value of α could be dynamically adjusted at each iteration. This would normally be done with another (small and efficient) solver over a set domain of possible values (e.g. $0.1 \leq \alpha \leq 2$). Other parameters e.g. the step angle with respect to the downhill direction, could also be introduced and optimised. Note that the process of *hyperparameter* optimisation can come at the cost of additional function evaluations.

There are plenty of other popular methods based around the idea of evaluating the derivatives of l.[56;57] Most notable due to its extremely rapid convergence is the *Newton–Raphson method* (or just Newton's method) which attempts to find the roots of the derivative of l. This can be shown to be identical to approximating $l(\boldsymbol{p})$ by an N-dimensional parabola tangent to l at $\boldsymbol{p}$ and required the evaluation of the Hessian $\boldsymbol{H}$ (matrix of second derivatives) of l.[cq] This method does require l to be twice continuously differentiable and for it to have a positive curvature (Hessian must be positive semi-definite – all eigenvalues of $\boldsymbol{H}$ are ≥ 0). Newton's method is mentioned here since the gradient descent can be considered as a special case of the Newtons method and, as such, it is the latter one which is usually implemented in popular scientific computing libraries.[cr]

There is, however, a problem with the above approach for applications like data fitting. The Newton's method requires the calculation of a Hessian and/or the Jacobian of l. In the code above, this has been done analytically with help of Mathematica, but in general such approach is not possible. The derivative can however be approximated with a *finite difference* scheme (see 2.3). This incurs additional function evaluations (more so for higher order schemes) but allows one to use Newton's methods for any (doubly) smooth function. The code above could be modified with, for example, the following central difference scheme:

```
f = @(x,y) log(1 + (x.*sign(real(x))).^(2 + sin(x+y))) + y.^2/20;
h = 1e-4; % Finite difference step size
J = @(x,y) [(f(x+h,y)-f(x-h,y)) (f(x,y+h)-f(x,y-h))]/(2*h);
```

Listing 2.14: Central difference scheme.

The numerical Jacobian is relatively cheap to compute, but the Hessian can be significantly more expensive, especially for large number of optimisation parameters (quadratic vs. linear scaling with the dimensionality of the parameter hyperspace). The most commonly used numerical optimisation methods, therefore, use an approximate Hessian in place of the exact (numerical) one, while still benefiting from the superlinear convergence of the Newton's method. Such Hessian approximant is initially calculated, but thereafter updated each iteration based only on the change in the function gradient. Depending on the Hessian update model, certain desirable properties can be ensured – the Hessian approximant is often necessarily symmetric and positive semi-definite.[56] This family of algorithms is referred to as the *quasi-Newton methods*.[cs] An

[co] It is however *not* the same since in gradient descent the *position* of the point is affected by the gradient of the l surface, whereas in a physics simulation it would be the points' *velocity*.

[cp] This process is referred to as the *hyperparameter optimisation* – a term is most commonly used when referring to solvers used in neural network learning.

[cq] Newton's method has originally been developed as a rootfinding algorithm where, given a starting point x_0 and a loss function $l(x)$, the algorithm iteratively 1) evaluates the gradient of l at x_j, 2) the point x_{j+1} where the tangent of $l(x_j) = 0$.

[cr] Newton's method with Hessian equal to the identity matrix $\boldsymbol{H} = \boldsymbol{I}/\alpha$ is equivalent to gradient descent with step size α.

[cs] Sometimes other algorithms with with approximate Jacobian/Hessian, are referred to as quasi-Newton. While this may be technically correct, the term 'quasi-Newton' ought to really be used only for algorithms which update the Hessian approximant each iteration rather than computing it anew.

implementation of a quasi-Newton algorithm, together with step-size α optimisation is implemented in Matlab within the unconstrained multivariate optimisation – `fminunc`. The optimisation above can be performed simply as:

```
options = optimoptions('fminunc'); % Default options
options.Algorithm = 'quasi-newton'; options.OptimalityTolerance = 1e-4;
[pf,fval] = fminunc(@(p) f(p(1),p(2)),[8.3 -8.9],options);
```

Listing 2.15: Optimisation with `fminunc`

The solution converges within ca. 15 iterations which is, roughly, 10x faster than the simple gradient descent.

Pseudo-gradient methods

Depending on the specific implementation of the loss function, it may not be locally differentiable, or the gradient may be zero. The finite difference method is guaranteed to return a result, but it may not specify a good direction to take a step in. More importantly, when used in fitting the loss function is based on a real, noisy data. The shape of l, especially around the optima, will therefore also have elements of noise imposed on it which makes the evaluation of a local gradient a poor choice. There however exists a whole family of optimisation methods which do not evaluate point-gradients but instead rely on a systematic probing of the value of l to arrive at the optimal point.

One of the most common of such methods is the *Nelder-Mead algorithm*. For an n-dimensional space of parameters, it chooses $n+1$ (distinct) points P in the vicinity of the starting point. Fully-connected $n+1$ points in n-space are the simplest n-dimensional polytope called a *simplex*. 2D simplex is a triangle, 3D a tetrahedron, 4D a 5-cell, etc. The algorithm then evaluates the loss function at the corners of the initial simplex, picks the worst point P_h (point with the worst l) and calculates the center of mass (centroid) $\bar{P}$ of all but the worst point. It then proceeds to modify the simplex in order to travel towards the optimum. The worst point P_h is reflected about $\bar{P}$ and the step is accepted if it means it is no longer the worst – *reflection*. Otherwise the worst point is moved half way towards $\bar{P}$ and, again, the step is accepted if it improves the function value – *contraction*. As a last resort, the simplex is expanded about the best point $P_l = \boldsymbol{P}@\min(\boldsymbol{F})$ – *expansion*. At each iteration P_h and $\bar{P}$ are computed anew. This is, in fact a slightly simplified version of the original algorithm,[58] but it works well for the purpose of demonstration.[ct] This algorithm can be programmed as follows:

```
f = @(x,y) log(1 + (x.*sign(real(x))).^(2 + sin(x+y))) + y.^2/20;
ff = @(p) f(real(p),imag(p)); % helper f - called with complex coordinates
tol = 1e-4; reflection = 1; contraction = 1/2;
p = 8.3+1i*(-8.9); % Starting point p0
P = p + exp(1i*(0:2)*2*pi/3); F = ff(P); % Initial simplex (triangle)
for j = 1:1000 % Limit number of iterations to 1e3
    [~,l] = min(F); [~,h] = max(F); % Find highest and lowest points
    Pc = P; Pc(h) = []; Pc = mean(Pc); % Find centroid of h-less P
    Pn = (1+reflection)*Pc - reflection*P(h); Fn = ff(Pn); % Reflect
    if any(Fn < F)
        P(h) = Pn; F(h) = Fn; % Accept if any better
    else % Contract
        Pn = contraction*P(h) + (1-contraction)*Pc; Fn = ff(Pn);
        if Fn < F(h)
            P(h) = Pn; F(h) = Fn; % Accept if better than F(h)
        else
            P = (P+P(l))/2; F = ff(P); % Expand about P(l)
        end
    end
```

[ct] The original implementation of the Nelder-Mead algorithm also included another case called *shrink*.[58] This case got triggered if the contraction failed, but before the expansion step. Shrink computes a new simplex which maintains P_l, and moves all the other vertices by $(P_i - P_l)/2$. This step was introduced to account for certain edge cases when the algorithm would move away from the optimal point in such a way as to make it come back in a very large number of iterations.

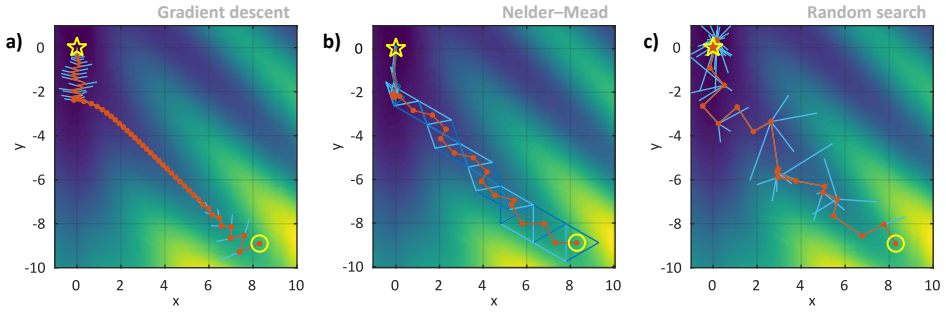


Figure 2.11: Comparison of **a)** gradient descent, **b)** Nelder-Mead simplex and **c)** random search algorithm traversing the same hypersurface, starting from the same point. The blue and cyan lines in indicate the size of the gradient in a), the shape of the simplex in b) and the attempted step c).

```
    % Check for convergence
    if max( abs(diff([P P(1)])) ) < tol; break; end
end
pf = mean(P);
```

Listing 2.16: Simplified Nelder-Mead algorithm.

For simplicity of the vector calculus involved, complex numbers have been used to represent the (x, y) coordinates. The main part of the algorithm is the reflection/contraction/expansion decision tree. The convergence criterion evaluates the length of the longest side of the simplex and compares it to the tolerance. The convergence of the algorithm above can be seen in Figure 2.11b.

Since the value of the function is still compared between points in a relative vicinity, a certain form of gradient is still (indirectly) evaluated. The Nelder-Mead algorithm (implemented in Matlab as `fminsearch`), as well as another *pseudo-gradient* method, *pattern search* (`patternsearch`)), which relies on an n-dimensional cross-shaped stencil are both excellent for poorly-differentiable loss functions which can be found in fitting problems.

Another approach to local gradient-free optimisation could be realised with help of random sampling. At each iteration a new, random step in n-space is taken. If the loss function value at the new step is lower, the step is accepted. The step is drawn from a normal distribution. In order to improve the runtime, the spread of the normal distribution can be reduced each time the algorithm has not accepted a step for a certain number of iterations.

```
f = @(x,y) log(1 + (x.*sign(real(x))).^(2 + sin(x+y))) + y.^2/20;
tol = 1e-4; step_size = 1; stall_limit = 10;
p = [8.3,-8.9]; % Starting point
fp = f(p(1),p(2)); stall = 0; % Initialize
for j = 1:1000
    pn = p + step_size.*randn(1,2); fn = f(pn(1),pn(2)); % Trial step
    if fn =< fp % Accept step
        p = pn; fp = fn; stall = 0;
    elseif stall >= stall_limit % Standing in one place for too long
        step_size = step_size/2; stall = 0;
    else
        stall = stall + 1;
    end
    if step_size < tol; break; end
end
```

Listing 2.17: Simple random search optimisation algorithm.

There exists strong evidence that algorithms from the *random search* family are the only reasonable way of exploring very high dimensional space $n \gg 100$.[53;59] Note, however, that a random search will not converge to the exactly the same point in two consecutive runs – it is not *deterministic*. This becomes significantly more pronounced if the loss function has multiple convex regions for the algorithm to 'get stuck' in.

Global vs local convergence

It is very important to always consider the phenomena of *local convergence*. The random search described above is an easy example thereof, but deterministic algorithms can get stuck in *local minima* too. If the search for an optimal point stops in a local minimum, it can lead to invalid and/or misleading fit results. Note that the starting point of the the deterministic algorithms is often random (within some domain), or it is chosen arbitrarily by the user. The consideration of the nature of the minimum found is therefore equally important for both kinds of algorithms, yet none described so far attempt to address it.

Different local minima can have certain physical meanings which can be explored. Careful study of local minima can also provide insight into the way the particular fit model (or otherwise, a minimisation problem) is posed and could help improving it. Local minima are very rare when performing *fitting* with $n \leq 2$ parameters,[cu] but they occur readily when more parameters are fitted. The prime cause of local minima is when more than one parameter of the model can be used to express some feature of the data. As the dimensionality of the parameter space increases, so does the number of possible degrees of freedom for each of the data features and therefore the likelihood of two (or more) of the parameters being able to account for it.

A periodic component of a model, either in the form of a trigonometric function or a modulo function, can also be a cause of local minima. This is, possibly, the easiest way of illustrating the concept of local minima. The loss function in the following example is periodic in the direction of the second parameter. There exists a global minimum at $f(2.5,1) = -1$, and a series of local minima vertically above and below it. Each minimum has its own *basin of attraction* which is the region of $\boldsymbol{p}$ where the solver coverages to that minimum. If the evaluation of l is relatively cheap, the shape of the basins of attraction can be explored by exhaustive search: recording which minimum has been reached as a function of the starting point in $\boldsymbol{p}$ space. In this case just the p_2 value can be used to easily distinguish the minima:

```
loss = @(p) -cos(2*pi*(p(2)-1)/2)+abs(p(2)-1) + (p(1)-2.5)^2;
p1s = linspace(-4,4,100); p2s = linspace(-4,4,100);
for p1i = 1:numel(p1s)
    for p2i = 1:numel(p2s)
        pf = fminsearch(loss,[p1s(p1i),p2s(p2i)]);
        B(p1i,p2i) = pf(2);
    end
end
```

Listing 2.18: Basins of attraction of the Nelder-Mead algorithm.

The plot of l as well as corresponding matrix `B` representing the basins of attraction can be seen in Figure 2.12a and 2.12b. The shape of these basins will obviously be dependent on the loss function, but also on the details of the solver used. The same exhaustive search has been performed for the random search algorithm with step size of 0.3 (see Listings 2.17 and A.4), and the results are shown in Figure 2.12c.

The region around each of the minima is obviously strongly attractive and thus forms a well defined basin. Departure of the starting point far from the minima makes the position of the obtained minimum highly uncertain, especially as the minimum moves up the high gradient of l and therefore is allowed a lot of leeway before reaching the valley. The random solver basins are much more fuzzy due to the stochastic nature of the solver. Notably though, there are regions of $\boldsymbol{p}_0$, where the deterministic `fminseach` behaves similarly to the random solver. In those regions the solvers' behaviour is *chaotic* (see section 'Chaos') with respect to the starting point – sensitive to infinitesimal perturbations in the exact value of $\boldsymbol{p}_0$. This is why the choice of the starting point of a solver is very important for problems with multiple local minima.

[cu] Note the emphasis on 'fitting'. Local minima are trivial to *engineer* even in low dimensional functions. Consider, for example, $f(x) = 2 - 1/(1+x^2) - \cos(x)$ which has a single global minimum at $(0,0)$ and an infinite number of local minima at integer multiples of 2π.

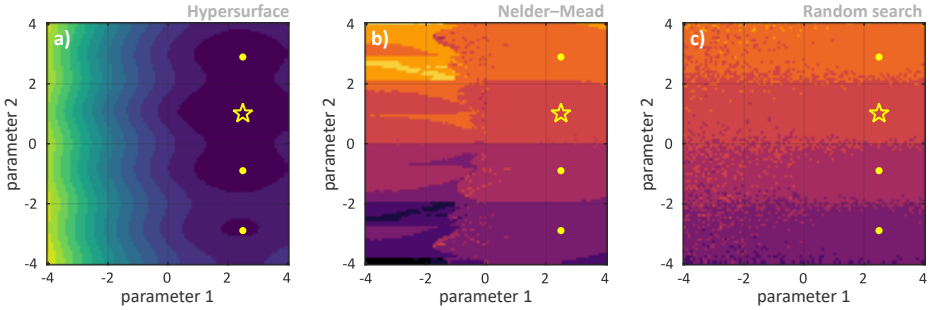


Figure 2.12: Basins of attraction of a hypersurface with multiple local minima. **a)** The hypersurface, with multiple minima indicated. The global minimum is indicated wit the star. **b)** Basins of attraction of the Nelder-Mead algorithm, as calculated in Listing 2.18. **c)** Basins of attraction for the random search algorithm on the same hypersurface. Calculated by replacing `fminsearch` in Listing 2.18 with `random_search` from Listing A.4.

Ideally, of course, the basin of attraction of the global minimum would span the entire domain of $\boldsymbol{p}$. The random search algorithm above used a smaller step size in order to illustrate the point. If the step size is increased to 1, the average size of the step is on the order of the distance the distance between the local minima, and the solution hops between different basins until (on average) it reaches the global minimum. This can be thought of as giving the initial point higher 'temperature', and the algorithm could even be modified to, sometimes, accept worse points (akin to a Markov chain) in order to encourage exploration. This is the basis of global search methods like '*simulated annealing*' (`simulannealbnd()` in Matlab) and '*basin hopping*' (`GlobalSearch`). Alternatively one could use an approach similar to the code above used to visualise the basins of attraction, but with a coarser grid, in order to hopefully have one of the points land in the basin of the global minimum. This is referred to as the *mulitstart* method (`MultiStart` in Matlab).

Whatever the method, one can only *hope* that the optimal point found is the global minimum of the function. There are no solvers which can find a global minimum of any function in finite time.[cv] Some solvers could even stop without reaching the minimum at all if they, for example, exceeded the iteration limit, or the attempted step size is smaller than a set tolerance limit.[cw] In general though, one would speak of the *putative minimum* – the best minimum that could be found given the method used, that is usually asserted to be the global minimum. A good way of ensuring that one has, indeed, found the global minimum is to use a sufficiently exploratory random algorithm, and run it multiple times (with different random seeds). The result of each of the runs can then be analysed and the putative minimum can be found.

The example below shows a use of such an approach in order to construct a flattop filter window (2.1). The window will be a simple linear combination of two basis functions – cosine components. The magnitudes of these components will be the fit parameters $\boldsymbol{p}$. An N-point unit window can be constructed from the parameters as follows:

```
function [w,p,s] = p_to_window(p,N)
%% Construct a scaled window from its components
n = 0:(N-1); % Window axis
p = [p(1)-p(2) p]; % Set edge samples to 0
b = bsxfun(@times, cos((0:2:4)'*pi*n/(N-1)), sign(mod(1:3,2)-1/2)');
w = p*b; % Window
% Scale the window and the components to unit magnitude
s = max(abs(w)); % Scaling factor
w = w/s; p = p/s; p = [p(2)-p(3) p(2:3)];
end
```

Listing 2.19: Create a 3-parameter window scaled to unit height.

[cv] Plenty of completely useless algorithms converge to the global minimum as the number of iterations approaches infinity (hence 'useless'). Random search with $\delta\boldsymbol{p}$ drawn from a normal distribution in n-space, and with a constant step size is a great example.

[cw] Most solvers have multiple possible stop conditions. When convergence to a well-defined minimum is required, it is therefore important to evaluate the actual criteria by which the iteration has been halted.

The design requirements of the flattop window are described by the loss function. In this case, we want to optimise the mainlobe flatness (difference between the 1st and 2nd sample), as well as the sidelobe magnitude, and sidelobe flatness. Since the window scaling is an arbitrary parameter (the window produced is always scaled to a unit window), it could be ignored, but for the sake of better convergence (better defined minimum) its distance from unit can also be added to the loss function. The loss function is defined as follows:

```
function phi = window_lossfun(p)
%% Loss function for creating a simple flattop window
if any(p<0); phi = inf; return; end
N = 1024; [w,~,s] = p_to_window(p,N); % Get the window
W = log10( abs(fft(w))/N ); W = W(1:20); % First 20 samples of the FT of w
phi = ... Loss function
    max( [log10(sqrt((W(2)-W(1)).^2)), -2] ) + ... Top flatness
    max( [median(W(3:end)), -6] ) + ... Sidelobe magnitude
    max( diff(W(6:end)).^2 ) + ... % Sidelobe flatness
    (s-1).^2; % Window scaling factor distance from unity
end
```

Listing 2.20: Loss function for creating a simple flattop window.

This is, in fact, a problem in multiobjective optimisation which is a subject of its own. In this case it does suffice to simply optimise the sum of all the terms of interest since they have been constructed to have similar magnitudes, and therefore similar contributions towards the overall value of l. The values of some of the components have been 'clamped' (with the `max()` function) so that they are not optimised past certain point and do not dominate the value of l. The values of the fit parameters have also been restricted to $|\boldsymbol{p}| > 0$ by setting $l = \infty$ if any of the components of $\boldsymbol{p}$ are negative.

The loss function can now be optimised multiple times with random starting points (uniform in the 0,1 range):

```
N = 1024; n = 0:(N-1); % Window axis
for j = 1:100
    [pf,fvals(j,:)] = fminsearch(@window_lossfun,rand(1,2));
    [w,pf] = p_to_window(pf,N);
    pfs(j,:) = pf;
end
[fvals,I] = sort(fvals); pfs = pfs(I,:); % Sort results according to fval
```

Listing 2.21: Multistart global optimisation.

The sorted function values for all the runs can be seen in Figure 2.13a. The plateaus correspond to loss function minima (or otherwise to points where the solver stalls). One can understand a lot about the problem its inherent features and/or the way it is posed, by analysing the solutions corresponding to the local minima. The FT of the windows corresponding to the 1st (global minimum), 3rd and 4th minima are plotted in Figure 2.13b. One can see that, when approaching the 3rd minimum, the solver compromised the top flatness for the sidelobe magnitude. Similarly, the solver gets stuck at the 4th minimum, most likely because reducing the single frequency component (3rd FT bin) would involve heavily compromising the mainlobe flatness due to spectral leakage.

There are advantages to performing this type of global minimum analysis by oneself, as it can warrant a much deeper understanding of the results as well as the problem itself. There does however exist a plethora of global solvers and methods.[17;57;60] The above approach (implemented in Matlab as `MultiStart`) is robust but not very consistent nor efficient in how it performs its search. If the number of function calls is of the essence, a good approach it to use a surrogate optimisation algorithm (`surrogateopt()`)[cx]. This type of

[cx] A different approach has been the specific problem of calibration of the BBCEAS field burst. Briefly, the loss function has been formulated such that the problem became that of rootfinding, rather than optimisation, and a golden section algorithm has been used. This was done because LabVIEW does not have extensive optimisation libraries, and an implementation of a golden section algorithm is significantly simpler than that of a surrogate solver (Section 5.5).

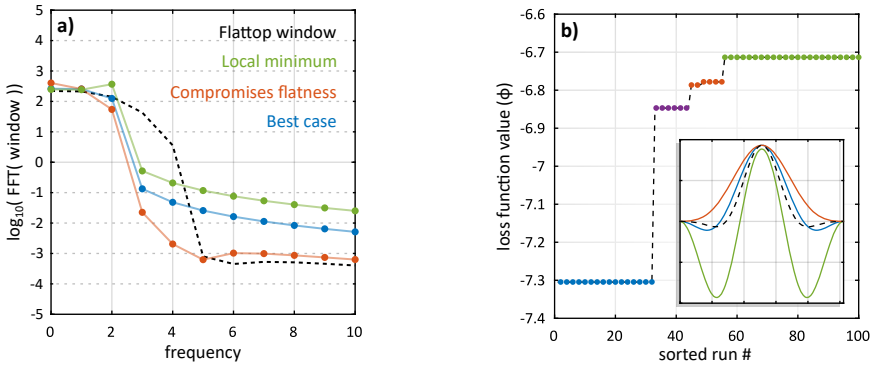


Figure 2.13: Flattop window construction by optimisation. **a)** Frequency spectra of selected windows found by the multistart optimisation of the loss function from Listing 2.20. Dashed black line shows an actual flattopwindow as defined by `flattopwin` in Matlab. **b)** Sorted value of the loss function for 100 runs of the mutistart solver. The selected regions are indicated in different colours. Normalised windows corresponding to each region can be seen in the insert. The purple region has not been shown in a) nor in the insert, as it is visually similar to the neighbouring cases and offers little insight.

algorithm samples the (expensive) loss function only a few times, and creates a surrogate model which then determines the new sampling points according to where the position of the minimum is most likely according to the surrogate. A popular choice, otherwise, is the simulated annealing (`simulannealbnd()`) algorithm which samples l by first taking large steps, and decreasing their size as the optimisation progresses, akin to how the temperature in the steel annealing process (hence the name).[17]

There also exist specialised 'exploratory' algorithms like the genetic evolution algorithm (`ga()` in Matlab) where a population of vectors of parameters $\boldsymbol{p}$ evolves, mutates and cross-breeds according to their fitness (inverse of the loss function), or the particle swarm optimisation (`particleswarm()`) where a literal n-dimensional swarm of sampling points explores the hypersurface of the problem. The multi-start method can also be adapted to get progressively repelled by the local minima in order to explore a large part of the problem space. This method is referred to as basin hopping (`GlobalSearch` in Matlab). These type of exploratory solvers are usually used when the parameter space is very high-dimensional, and the goal is to usually have a good, but not necessarily the absolute best solution. A good usage example is the 2006 NASA project: evolved antenna where such an algorithm was used to optimise the antennas' radiation pattern.[61] Exploratory algorithms are also finding used in neural network evolution and protein folding.[62–64]

Shape of the hypersurface

Norms The convergence of a fit, uniqueness, and the nature of the solution can all be affected by changing the shape of the hypersurface. The gradient descent method, for example, tends to converge to a solution much faster if a 2-norm is used for the potential function instead of a 1, or ∞ norm. This is because, when using the 2-norm, the gradient of l *increases* as one travels further away from the optimum, and so the solver takes larger steps.[cy] This is not the case with the other types of norm – the 'walls' of l around the solution have a constant gradient, and so the solver takes equal sized steps and can take a very long time to converge if started far away from the solution. The choice of the norm of the residuals can, in fact be much wider than the typical norms. One could, for example, minimise the the root-*median*-square, the standard deviation, interquartile range, or even the maximum peak magnitude of the FT of the residuals. The L2 norm should, however, be used unless there is a very good reason to believe another norm function would be more applicable.

[cy] In contrast to other norms, 2-norm hypersurface is also continuously differentiable. This is, in principle, required for gradient-descent methods but, in practice, usually does not matter as the solver can still evaluate the sub- or pseudo-derivatives of l and take a step.

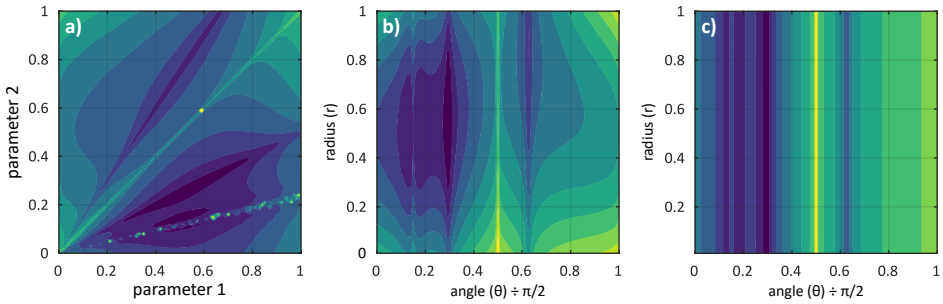


Figure 2.14: Hypersurfaces for the flattop window construction problem. **a)** Exact hypersurface, as optimised in Listing 2.21. **b)** The same hypersurface as in a), but now expressed in radial coordinates. Note that this aligns the wells of the optima with the basis vectors of the hypersurface. **c)** Hypersurface for the unregularised problem where the window scaling factor is allowed to vary arbitrarily. This is done by removing the last objective of the loss function in Listing 2.20. Each minimum is now an infinitely long well in the hypersurface.[da]

Coordinates Another way in which the shape of the hypersurface can be altered to facilitate the solution of the problem it to rephrase l in therms of different set of parameters. The entries of $\boldsymbol{p}$ define a Cartesian space in n-dimensions. Nothing is preventing one from exploring that space with for example a polar set of coordinates $n-1$ angles and a distance from origin for an arbitrary choice of origin. For example for two parameter problem, as opposed to minimising $l(p_1, p_2)$, one could equivalently minimise $l(r\cos(\theta), r\sin(\theta))$ where $r = \sqrt{p_1^2 + p_2^2}$ and $\theta = \mod\left(\operatorname{atan}\left(p_2/p_1\right), \pi/2\right)$.[cz] The hypersurface for the flattop window construction problem discussed above, as well as its polar representation has been shown in Figure 2.14a and 2.14b respectively. Certain types of problems lend themselves to being represented in a polar coordinate system. This is especially useful when using a solver which, somehow, relies on the orientation of the parameter space. For example, certain solvers take steps, genetic algorithm cross-breeds/mutates individuals and the finite-difference method evaluates the gradient along the axis of $\boldsymbol{p}$. Note that, by constraining the range of angles θ over which the solver can work, we implicitly constrain the elements of $\boldsymbol{p}$ to be positive.

Bounds The parameter space can be better explored if any of the parameters of the model can take values in a finite, or otherwise bounded, range. Any prior knowledge on the solution can, and should, be implemented into the optimiser constraints and model regularisation (see later). For example, there is no point allowing negative time constant τ when modelling an exponential *decay* with $f(x) = \exp(-x/\tau)$. This can be implemented either by altering the model to e.g. $f(x) = \exp(-x/|\tau|)$, or by constraining the optimiser to work only with positive values of τ. The former approach will yield the correct solution, but sometimes with inverted sign (since the model does not differentiate between them). The latter approach requires the use of a solver which allows for boundary conditions like, for example, `fmincon` in Matlab.[db] Alternatively one could use the fact that the exponential function maps the entire number line to positive real numbers to instead fit the log of $\tau_l = \log(\tau)$ to the model $f(x) = \exp(-x/\exp(\tau_l))$. Since $\log(\tau)$ will be always positive (for real τ), the boundary condition is implemented in a functional way.

Overparametrisation With certain formulations of a problem, one can find that many of the solutions correspond to the same solution (*ill-posed* problem). An example of this could be be fitting a model which is always scaled up to the data (which corresponds to fitting the shape, not the magnitude of the model). Consider the following example where a sinusoidal oscillation with zero phase is fitted to the data as a function of its frequency and amplitude.

```
1 x = linspace(0,1,1e2); m = @(x,f,a) a.*sin(2*pi*f*x); % Model
```

[cz] Technically a full representation in polar coordinates uses a `atan2(x,y)` function which correctly maps the position of the entire Cartesian plane, not only the positive quadrant.

[da] The hypersurface is, effectively, overparametrised. Note that this implies that a solution in lower dimensional space (less fit parameters) is possible. In this case, since the wells of the minima are aligned with one of the basis vectors (radius in Figure 2.14c), its value can be simply set to 1.

[db] Such solvers usually modify the loss function by adding to it barrier functions with progressively sharper and sharper cut-off, and analyse how does the position of the minimum change. This is done in case the minimum appears *on* the boundary.

```
y = 2*m(x,2,2) + randn(size(x)); % Data
fit_scaled = @(p) m(x,p(1),p(2)) .* (m(x,p(1),p(2))'\y'); % Scaled model
loss1 = @(p) mean( (y-fit_scaled(p)).^2 ); % Loss funciton
pf = ga(loss1,2,[],[],[],[],[0 0],[4 2]); % Solve with ga
```

Listing 2.22: Example of an ill-posed problem.

The frequency is a meaningful parameter of the model – it expresses a degree of freedom of the data. However, Since the fit function scales the model to the data, the second parameter of the model is, effectively, mute as the solution is independent of its value (for $a > 0$ anyway). The solution of the least squares fit problem with `ga()` yields a global minimum in f ($f \approx 2$), but a different solution in a every time. This can be considered as a trivial case of *overparametrisation* (see Section 2.2) – allowing for too many degrees of freedom in the model. There are a few ways in which one can modify the problem in order to achieve a consistent solution.

```
% One parameter model (a fixed at 1). p now has only one element
loss2 = @(p) mean( (y-fit_scaled([p 1])).^2 );
% Unscaled fit function
fit_unscaled = @(p) m(x,p(1),p(2));
loss3 = @(p) mean( (y-fit_unscaled(p)).^2 );
% Regularized problem
alpha = 0.1; % Regularization parameter
loss4 = @(p) mean( (y-fit(p)).^2 ) + alpha.*(p(2)-1).^2;
```

Listing 2.23: Better formulation of the problem from Listing 2.22.

The easiest thing to do in this case would be to simply get rid of the useless parameter (`loss2`), or use and unscaled fit function (`loss3`). The former is often preferable as it results in a simpler (1-parameter) model where the scale is fitted implicitly by the Matlab's backslash operator – `\` (abbreviation for `mldivide()`). This is a useful trick which can be used to fit linear algebra components of the model (see 2.26). These two methods are, however, not always available as the model may be overparametrised without any of the parameters being so obviously responsible. The problem can, however, be *regularised* – modified to express a preference for a certain type of solution and therefore force the solver into a single minimum. In the case above the `loss4` function has been modified by adding an attractive potential for p_2 to a value of 1. The optimal value of the *regularisation parameter* `alpha` can be found explicitly for a given problem, or an *ad hoc* value can be used.[dc] α is often small in order not to disrupt the real solution. In this case, due to the simplicity of the problem, any (positive) value of α works. This type of regularisation has also been used in the flattop window construction example in Section 'Global vs local convergence'. There the `window_lossfun` has been regularised such that there was a preference for the resultant window to have a unit scaling, even though any solution would work as it could easily have been rescaled afterwards. Figure 2.14c shows the loss function hypersurface for this problem *without* the regularisation.

Tikhonov regularisation The term 'regularisation' usually refers to expressing a preference for certain smoothness of solution when talking about hard *inverse problems* in linear algebra, the most common of which is the *Tikhonov regularisation*. Any linear algebra problem can be expressed, ultimately, as $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$ where some matrix operator $\mathbf{A}$, acts on an unknown (column) vector $\boldsymbol{x}$ to give a known resultant (column) vector $\boldsymbol{b}$. To illustrate regularisation of such problem, lets consider a problem of taking a numerical derivative of a noisy dataset. To formulate the problem in terms of linear algebra lets consider the action of a cumulative integral operator $\mathbf{A}$ on the answer $\boldsymbol{x}$ (the derivative of our data) to give us the data $\boldsymbol{b}$, shifted such that the first sample is equal to 0 (to mimic the action of the *cumulative* integrator). If $\mathbf{A}$ was invertible, the solution would be simply $\boldsymbol{x} = \left(\mathbf{A}^{\mathrm{T}}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathrm{T}}\boldsymbol{b}$ (or `x = A\b` in Matlab), however $\mathbf{A}$ is *singular* or close to singular,

[dc] There are formal ways of establishing the 'best' value of the regularisation parameter(s). In general, regularisation attempts to decrease the solution *variance*, at the price of introducing *bias*. If the two can be evaluated as a function of the regularisation parameter, its optimal value lies when their product is minimised. This is referred to as the bias-variance trade-off problem.

depending on the method of integration chosen to construct it.[dd] We can attempt to find a solution by minimising the (mean squared) distance between $\mathbf{A}\boldsymbol{x}$ and $\boldsymbol{b}$:

```
N = 100; k = linspace(0,2,N)'; dk = k(2)-k(1);
b = 0.6*cos(2*pi*k.^(2.3) + pi*rand) + 0.1*randn(size(bo)); b = b-b(1);
% Construct the cumulative integrator according to Simpson's rule
A = spdiags([ones(N,1) 2*ones(N,N-1)],[0 -1:-1:-(N-1)],N,N); % NB Sparse
A(:,1) = ones(N,1); A(1,:) = zeros(1,N); A = A.*dk/2;
% Formulate the optimisation problem
loss1 = @(x) mean( (A*x-b).^2 );
x0 = randn(size(bo)); % Random starting point
xf = fminunc(loss1,x0);
```

Listing 2.24: Optimisation solution to the numerical derivative problem.

The solution is, however, unstable – it varies heavily upon small perturbation of the data, and it tends to be very discontinuous. This problem is referred to as *ill-conditioned*. Tikhonov regularisation implements a preference for a 'smooth' solution by adding a regularisation term $\mathbf{T}\boldsymbol{x}$ to the loss function.[de] The problem then becomes:

```
% Construct a finite difference operator
D = spdiags([ones(N,1) -ones(N,1)]/2,[1,-1],N,N);
D(1,[1,2]) = [-1,1]; D(end,[end end-1]) = [1,-1]; D = D./dk;
% Formulate the regularised problem
%alpha = 0.04; T = alpha*speye(N);
alpha = 0.002; T = alpha*D;
loss_regularised = @(x) mean( (A*x-b).^2 ) + mean( (T*x).^2 );
x_regularised = fminunc(loss_regularised,x0);
```

Listing 2.25: Regularised solution to the numerical derivative problem.

$\mathbf{T}$ is called the Tikhonov matrix and it is either an identity or a finite difference operator – loss function preference for smaller L2 norm or smooth solution vector (x) respectively. Figure 2.15a shows the ideal solution (derivative of data without noise), the regularised solution as formulated above, as well as the result of the simple finite difference operator on x. The finite difference operator acts akin to a high-pass filter. It therefore amplifies the effect of the noise (left half of the solution) but captures the higher frequency oscillations (right half). In contrast the regularised solution is smooth, but this is achieved due to the Tikhonov operator acting akin to a low-pass filter. The solution therefore underestimates the high frequency components of the solution (right half). Note that the solution using a loss function is, in effect, a very overparametrised N-dimensional model. This is a hard and time consuming problem for a deterministic optimiser and it effectively prohibits using this specific method for any larger datasets. Luckily, the key feature of Tikhonov regularisation is that it is expressed as a *linear operator*. This allows one to reduce a fitting problem to a (and much more efficient) linear algebra one – inverting a matrix $\mathbf{A}^{\mathrm{T}}\mathbf{A}+\mathbf{T}^{\mathrm{T}}\mathbf{T}$ (which now *is* invertible). This can be done e.g. with the backslash operator, and works well even for very large N due to the use of sparse matrices for $\mathbf{A}$ and $\mathbf{T}$.

```
x_linalg = (A'*A + T'*T) \ (A'*b);
```

Listing 2.26: Regularised least-squares fit.

Robustification A concept, specifically in the context of fitting, similar to regularisation is the *robustification* – making the fitting less sensitive to noise in the data, especially to 'outliers'. The concept of robust fitting can be illustrated with simple linear regression.

```
x = linspace(-1,1,100); m = @(p) p(1)*x + p(2);
y = m([3,1]) + 0.3*randn(size(x));
```

[dd] A different way of tackling a problem of taking a derivative of a noisy data would be, for example, to use a linear *local estimator* (loess) – fit a line to each set of n points (for odd n) and use its gradient as the estimate of the derivative of the data. One of the first, and still used, examples of such technique is the Savitsky-Golay local regression smoothing and differentiation method.[65]

[de] The regularised solution can be shown, under certain assumptions, to be the most probable solution. Hence, a link with the maximum likelihood estimation can be made.

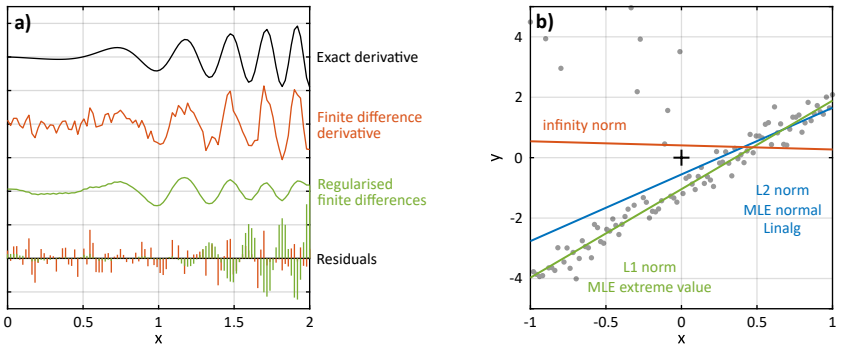


Figure 2.15: a) Unregularised (red) and regularised (green) finite difference solution to the problem of numerical differentiation of noisy data. The residuals of both solutions, with respect to the exact solution (black) are shown at the bottom. The unregularised solution reproduces the high-frequency components of the exact signal, but introduces a lot of noise. The regularised solution acts akin to an LP filter. It attenuates the noise and reproduced the low-frequency components well, but it introduces bias in the high-frequency components of the solution. **b)** Unregularised (red and blue) and regularised (green) linear regression of a dataset with outliers present. The exact methods used for each fit have been given on the plot. Visually indistinguishable fits have been grouped together and displayed as one line.

```
3 y(randi(50,1,10)) = 5*rand(1,10); % Add outliers
```

Listing 2.27: Linear regresion data with outliers.

The loss functions for the following methods are:

```
1 loss_L2 = @(p) mean( (m(p)-y).^2 ); % L2 norm
2 loss_L1 = @(p) mean( abs(m(p)-y) ); % L1 norm
3 loss_Linf = @(p) max( abs(m(p)-y) ); % Linf norm
4 % MLE with normal and extreme value noise models
5 loss_MLEnorm = @(p) -sum(log( normpdf(m(p(1:2))-y,0,p(3)) ));
6 loss_MLEext = @(p) -sum(log( pdf('Extreme Value',m(p(1:2))-y,0,p(3)) ));
```

Listing 2.28: Linear regresion models.

The plot of `x` vs. `y`, as well as the fits using the above methods have been shown in Figure 2.15 b). Simple 2-norm (`loss_L2`) is sensitive to the outlier points and results in underestimate of the gradient. 1-norm (`loss_L1`) is, in this case, a better choice as it concentrates only on the the *majority* of the residuals (rather than all of them). On the other hand the ∞-norm (`loss_Linf`) focuses only on the largest residual, and therefore is 'infinitely' sensitive to outliers. The problem could be expressed in terms of linear algebra by inverting a *Vandermode* matrix (a matrix of powers of `x`) which turns out to be equivalent to 2-norm fit. Since the answer of this problem ends up being a vector of polynomial coefficients, it does not regularise easily (unless there is a prior knowledge of any of the polynomial coefficients).

An approach which *does* allow robustification is, however, the *maximum likelihood estimation* (MLE). As opposed to minimising least squares potential between the fit and the data, MLE *maximises* the probability of the data given a normalised probability distribution function (pdf) of the residuals. A simplest case is to assume the residuals to come from a normal distribution with zero mean and an unknown standard deviation (`loss_MLEnorm`). The set of model parameters is augmented by the spread (standard deviation) of the normal distribution and minimised as a function of all of them.[df] The residuals are evaluated for a trial set of model parameters as well as the standard deviation (std) of the distribution. The probably of each of the residuals is calculated and the joined (assumed to be) independent probability – the product of all the probabilities is evaluated. Due to limitations of floating point arithmetic (not being able to deal well with very small numbers) it turns out to be much better to, instead of a product of probabilities, maximise the sum of the logarithms of the probabilities.

Normal distribution MLE turns out to be, again, equivalent to 2-norm optimisation. MLE does, however, allow one to use a better model of the noise – one which includes the outliers, in order to get a more robust

[df] Note that its important for the distribution to be normalised, as otherwise its parameters (e.g. standard deviation) will not converge.

fit. In this case, since all the outliers are assumed to be known to be positive, an *extreme value distribution* – a normal distribution with a heavy tail on one side, has been chosen (`loss_MLEext`). Overall, 1-norm and the extreme value distribution MLE both give the best results. The MLE approach does allow significantly more flexibility in terms of the choice of the probability density function.

Eigenvalues As mentioned, one can also express a preference *against* certain solutions. This can also be considered as a regularisation technique and it is useful when one intends to find many minima of the same loss function. For example, one can find all the eigenvalues of a matrix by finding one, and then modifying the loss function against it. The loss function for the eigenvalue problem $\mathbf{AV} = \mathbf{DV}$ is for the determinant of **A**–**D** to be zero (**A**–**D** has to be singular), where **D** is a matrix with eigenvalues on the diagonal, and V is a matrix of (column) eigenvectors. For a random 3×3 matrix **A**, the eigenvalue problem can be solved as follows:

```
A = rand(3,3); A = 1.5*eye(3) + A*A'; % Positive semidefinite matrix
% Linear algebra solution (with LAPACK)
[V,D] = eig(A); D_eig = diag(D)';
% Optimisation solution
lossfun_1 = @(L) log10( det(A - L*eye(3)).^2 ); % Initial problem
d1 = fminsearch(lossfun_1,0); % Find an eigenvalue
lossfun_2 = @(L) lossfun_1(L)+log10((L-d1).^-2); % Discourage 1st solution
d2 = fminsearch(lossfun_2,0); % Find another eigenvalue
lossfun_3 = @(L) lossfun_2(L)+log10((L-d2).^-2); % Discourage 2nd solution
d3 = fminsearch(lossfun_3,0); % Find the last eigenvalue
D_optim = sort([d1 d2 d3]); % Vector of sorted eigenvalues
```

Listing 2.29: Eigenvalues of a random 3×3 matrix by optimisation.

The loss functions for each stage of the problem have been plotted in Figure 2.16. A has been chosen as a positive semidefinite matrix to ensure positive eigenvalues. A more general solution for N eigenvalues can be seen in Listing A.6. Analogous approach would work for any matrix, but more parameters could be needed as the eigenvalues could be imaginary. This technique could also be applied to the basin hopping algorithm in order to encourage exploration of the problem hypersurface. Note, that the shape of the *discouragement function* (in this case the $\log_{10}\left((L-d_n)^{-2}\right)$ term) has to be well-matched to the problem in question.

Uncertainty of the optimum

When on the optimisation algorithm stops, it is a reasonable and very prudent to ascertain why. The solver could have hit any of the safety ceilings set upon it – limit on the number of iterations, function evaluations, execution time etc. The solver could encounter an unrecoverable error and, as opposed to crashing, it exited graciously, reported the error and gave the most recent valid solution. The most obvious answer, however, is that the optimality tolerance of the loss function has been met. This could mean that either the size (2-norm) of the last step taken in the $\boldsymbol{p}$ space is too small (*step tolerance*), or that the decrease in the value of the loss function itself is too small (*function tolerance*).[dg] When such an optimal point is reached, one could reasonably wonder how well defined it is, and therefore how certain its position. This question is especially important in optimisation problems when applied to fitting since the uncertainty in the fitted parameters represents the uncertainty of the fit and often translates directly to uncertainty in the parameters of interest.

The simplest method of evaluating the shape of l around the minimum $\boldsymbol{p}_f$ is to use its quadratic approximation l_q – a Taylor expansion of $l(\boldsymbol{p})$ about $\boldsymbol{p}_f$. The quadratic approximation of l at $\boldsymbol{p}_f$ is

$$l_q(\boldsymbol{p}) = l(\boldsymbol{p}_f) + \mathbf{J}_f(\boldsymbol{p}-\boldsymbol{p}_f) + \frac{1}{2}(\boldsymbol{p}-\boldsymbol{p}_f)^{\mathrm{T}}\mathbf{H}_f(\boldsymbol{p}-\boldsymbol{p}_f) \tag{2.18}$$

[dg] Other, more complicated optimality measures are also possible.

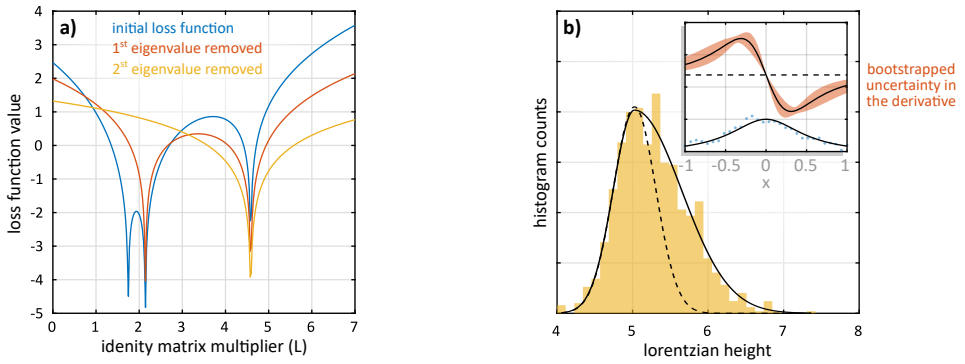


Figure 2.16: a) Loss function value for the eigenvalue optimisation problem, as calculated in Listing 2.29. The successive basins are removes from the loss function by cumulative addition of a discouragement function $\log_{10}\left((L-d_n)^{-2}\right)$. **b)** Histogram of bootstrapped height of the Lorentzian fit, as calculated in Listing 2.31. The centers of the histogram bins are fitted with a Gaussian (dashed line) and an asymmetric piecewise Gaussian (solid black line) lineshapes. The fit functions are given in Listings A.16 and A.18 respectively. The insert shows bootstrapped prediction bounds on the derivative of the same fit. Note that the prediction bounds are not symmetric, nor are they constant in x. Since the model is fixed at the origin, the prediction bounds correctly vanish at $x = 0$.

where $\mathbf{J}_f$ and $\mathbf{H}_f$ are the Jacobian and Hessian matrix evaluated at $\boldsymbol{p}_f$. This is a paraboloid in N-space whose widths in each of the cardinal directions of $\boldsymbol{p}$ – each of the fit parameters, will correspond to the uncertainty about that parameter. If we assume $\boldsymbol{p}_f$ is the optimum, the Jacobian is zero (derivative of l is zero) and the second term of l_q disappears. Furthermore, the width of the paraboloid is independent of the constant term of the l_q, and so the uncertainty of the optimum becomes dependent only on $\mathbf{H}_f$. In general, if the Hessian (matrix of second derivatives) of l at $\boldsymbol{p}$ is positive semi-definite (all eigenvalues ≥ 0), the paraboloid will be convex and have a single global minimum (analogous to the 1D quadratic as shown at the beginning of this section). The asymptotic variance (what the variance would tend to as the number of data points would approach infinity) of $\boldsymbol{p}_f$ can be shown to be related to the inverse of the Hessian. If the standard deviation of the noise is approximated by the standard deviation of the residuals σ_r, the uncertainty of i^{th} parameter δp can be approximated by:[dh]

$$\delta p_i = \sqrt{\sigma_r^2 \mathbf{H}_{ii}^{-1}} \tag{2.19}$$

The uncertainty here is the equivalent of one standard deviation and therefore corresponds to $\approx 68.2\%$ confidence interval (assuming normal distribution of noise). Many solvers evaluate the hessian internally and allow it as one of the outputs. The Hessian can also be evaluated with a simple finite difference scheme after the optimisation terminates. To illustrate lets consider the Lorentzian fit shown in Section 2.2. With the same problem setup, the fit parameter uncertainty can be calculated as follows:

```
[pf,~,~,~,~,H] = fminunc(lossfun,[1 1]);
fit = m(pf(1),pf(2));
dp = sqrt( var(y-fit)*diag(inv(H)) )';
```

Listing 2.30: Fit uncertainty based on the Hessian.

Where `diag` has been used to extract the main diagonal if the inverted Hessian. In this case $\boldsymbol{p}_f = [0.962, 5.090]$ and $\delta\boldsymbol{p} = [0.138, 0.457]$ (with the exact value being, as previously, $\boldsymbol{p} = [1, 5]$). This is a good, fast and simple approach, provided one has reached a convex optimum, or the optimum is sufficiently well behaved. It may happen that the minimum is well-defined in one dimension of $\boldsymbol{p}$, but is a long valley in another (see, for example, the unregularised solution in Figure 2.14c)) which gives an ill-conditioned Hessian, the Hessian may behave poorly under inversion not be invertible at all or not computable numerically due to noise in the data showing up around the minimum of the loss function. This method also not does take into account any boundary conditions and tends to deal badly with minima which are not well approximated by a quadratic.

[dh] The noise is also assumed to consist of independent and identically distributed (IID) samples.

An alternative to using the Hessian for fit uncertainty estimation is the statistical *bootstrap*. First a best fit of the model to the data is obtained and the residuals of the fit are calculated. Then a new set of residuals is calculated by *sampling with replacement* of the original residuals. The new residuals are then added to the *fit*, and refitted. This procedure, always starting at the original data and fit, is repeated a large number of times to build up a distribution of the fit parameters. These distributions are then analysed to get the corresponding uncertainties of the fit parameters. The bootstrap of the residuals for the above fit can be done as follows:

```
fit = m(pf(1),pf(2)); r = y - fit;
options = optimset('Display','none'); % Disable console output
for j = 1:1e3 % Number of iterations chosen
    nr = datasample(r,numel(r),'Replace',true); % Resample residuals
    nr = nr.*sign((rand(size(r))>0.5)-0.5); % Randomize sign
    ny = fit + nr; % Fit + bootstrapped residuals
    lossfun = @(p) log( sqrt(sum( ( ny-m(p(1),p(2)) ).^2 )) );
    ps(j,:) = fminunc(lossfun,pf,options);
    %ts(j,:) = gradient(m(ps(j,1),ps(j,2)))./dx; % Bootstrap gradient
end
dp2 = std(ps);
```

Listing 2.31: Fit uncertainty based on bootstrap resampling of the residulas.

The actual number of bootstrap samples is arbitrary, although at least `100*numel(p)` is recommended. This method, of course, takes significantly more time in comparison to the Hessian uncertainty estimate due to many more function evaluations needed. The refitting, of the data with new residuals is less expensive than the initial fit though, as a good starting point – $\boldsymbol{p}_f$ is already known.

The residual bootstrapping works on surprisingly few assumptions, mainly that the data comes from a noiseless model with additive noise of any kind, and that the number of data points is large enough that the random noise behaves asymptotically. Similarly to MLE, as long as a (broadly) correct model of the noise is used for resampling, bootstrap will work. The resampled residuals should resemble the original residuals not only in magnitude but also in the autocorrelation/spectral properties. For very correlated residuals this can be achieved by (overlapping) block resampling and randomisation of the sign of each of the blocks (wild bootstrap).

Since the bootstrap refits the data with the same optimiser and loss function as the initial fit, it respects any boundary conditions and constraints implemented therein. If, for example, a certain fit parameter is constrained to values larger than 0, every bootstrapped sample will adhere to that constraint too, whereas the Hessian uncertainty estimate may yield uncertainty value larger than the parameter itself. Resampling also respects the correction between the fit parameters. If one of the parameters always goes up when the other goes down, this will be reflected in the sampled distribution and can be observed. Moreover, bootstrap allows for analysis of parameters which are not normally distributed (which is one of the assumptions of the Hessian method). The bootstrapped distribution for every problem and parameter should be analysed individually, but as a general approach one can talk about the *central tendency* as well as the *lower* and *upper spread*. The recommended 'general purpose' measures for these are *median* and *half-normal* standard deviation – obtained from data only above or only below the median. A convenient function to calculate these is:

```
function [m,u,l] = mul(x)
m = median(x); x = x-m; % Edtimate central tendency as the median
u = x(x>=0); u = sqrt(sum(u.^2)./(numel(u)-1)); % Upper spread
l = x(x<0); l = sqrt(sum(l.^2)./(numel(l)-1)); % Loer spread
end
```

Listing 2.32: Central tendency and upper/lower spread estimation.

The distribution of the second fit parameter p_2 (the height) from the Lorentzian fitting above is shown in Figure 2.16b. Its obvious that the distribution is not normal and that the uncertainty is wider on the positive

side. In this case this can be attributed to how the model is formulated – the fit Lorentzian is assumed to always go though the point (0,0). Another advantage of bootstrap is that *any calculable heuristic* of the data can be bootstrapped and its distribution analysed. This is in contrast to the Hessian approach which only gives uncertainties on the fit parameters.[di] This can be seen in the insert of Figure 2.16b where the (numerical) gradient of the fit has been calculated and bootstrapped. The uncertainty bounds of the gradient can be seen to vary strongly with the position along the abscissa. This is difficult to calculate otherwise, but trivial to bootstrap. Note though, that since bootstrapping is based on resampling the residuals, it is only suitable to fitting problems, not every optimisation problem.

[di] The uncertainties of simple functions of the fit parameters can also be calculated by means of error propagation. This, however, assumes uncorrelated uncertainties. Bootstrap allows one to analyse the correlations between parameters/heuristics too.

2.3 Numerical integration

Differential equations are the mathematical description of dynamics, as well static points of dynamical problems. Examples of such problems are chemical reactions, propagation of electromagnetic waves through space and eigenvalue problems in quantum mechanics. It is therefore important to be able to solve such equations. The solution must always involve a set of *boundary conditions* (or initial conditions) which define the solution of the problem as much as the differential equations governing it. Certain differential equation problems can solved analytically, especially for simple cases, but this may be tedious and any alteration to the structure of the equation can often render the analytical solution impossible. Instead, this section will outline methods for solving such equations numerically.

Finite differences

The basis of solving differential equations numerically is the idea of approximating derivatives by finite differences. Lets consider a Taylor expansion a a function $f(x)$ about point $x+h$ where h is a small number, we get:

$$f(x+h) = f(x)+hf'(x)+\mathcal{O}\left(h^2\right) \tag{2.20}$$

where $\mathcal{O}\left(h^2\right)$ are all the higher terms of the expansion, the major one of which is proportional to h^2, and the other ones can be neglected since they comprise of higher powers of h, and h is small. Rearranging the above equation we get:

$$f'(x) = \frac{f(x+h)-f(x)}{h}+\mathcal{O}(h) \tag{2.21}$$

where the order term $\mathcal{O}\left(h^2\right)$ has become $\mathcal{O}(h)$ due to division by h. Equation 2.21 yields an approximation of a first derivative of f, accurate up to $\mathcal{O}(h)$. As $h\to 0$, the order term vanishes and the formula becomes exact. This formula is referred to a *first-order forward* finite difference.

There exists a plethora of finite difference formulae for any number of (partial) derivatives of any degree. All of them are based on repeated applications of Taylor expansions with respect to different space (h) and time (k) steps. The finite difference formulae can also be derived from one another. Taking the average of the formulae for the forward and the analogous *backwards* difference (with the incremental step being taken in the other direction; $-h$ as opposed to $+h$) yields:

$$f'(x)\approx\frac{1}{2}\left(\frac{f(x-h)-f(x)}{h}+\mathcal{O}(h)+\frac{f(x+h)-f(x)}{-h}-\mathcal{O}(h)\right)=\frac{f(x-h)-f(x+h)}{2h}\ \left(+\mathcal{O}\left(h^2\right)\right) \tag{2.22}$$

This is a *center* difference formula. The first order error terms cancel, and a similar analysis with inclusion of more terms of the Taylor expansion reveals that the center difference formula is, in fact, accurate up to a term $\mathcal{O}\left(h^2\right)$ – the error in the approximation of the derivative falls as a second power of h, not h itself. The formula for the second derivative of f in x can be also found in a similar way:

$$f''(x) = f'\left(f'(x)\right) = \frac{f'(x+h)-f'(x)}{h} = \frac{f(x-h)-',2f(x)+f(x+h)}{h^2}\ \left(+\mathcal{O}\left(h^2\right)\right) \tag{2.23}$$

where the 'outside' differential has been approximated by a forward, and the 'inside' one by the backwards finite difference.

Note, that a larger overall step is required for a more accurate formula, as well as to obtain a higher derivative. This is because more and more information about f globally is required to correctly approximate its derivative. This is taken to its logical extreme by the Fourier derivative methods. These are based on a FT relationship:

$$\mathrm{FT}\left\{f'(x)\right\} = i\omega\cdot\mathrm{FT}\left\{f(x)\right\} \tag{2.24}$$

where ω is the appropriate frequency axis of the FT of $f(x)$.[dj] Differentiation/integration can therefore be thought of as multiplication/division by $i\omega$ in the Fourier domain. The FT (and its discrete equivalent) can, therefore, be used to calculate the numerical derivative of an arbitrary function. This, in effect, uses all of the data to calculate the derivative at each point, hence an increased accuracy. The FT derivative method can also be faster and more resilient to numerical roundoff error accumulation than finite difference. This depends on the specific application though. Note that an implicit assumption of a FFT algorithm is that the underlying function is periodic. If this is not the case, error ripples will appear in the derivative close to the edges of the signal.

There exists another family of finite difference methods used in certain cases when the numerical roundoff becomes important. For functions whose extension into the complex plane is well-defined, if $f(x)$ is expanded about a point just off the real axis, the imaginary component becomes proportional to the derivative. Setting $h \to ih$ in Equation 2.20, and rearranging gives:

$$f'(x) = \mathrm{Im}\left\{\frac{f(x+ih)}{h}\right\} + \mathcal{O}\left(h^2\right) \tag{2.25}$$

This type of complex-step formula is more accurate (note the $\mathcal{O}\left(h^2\right)$ as opposed to $\mathcal{O}(h)$ in Equation 2.20), and more resilient to roundoff errors.[66;67]

The three approaches discussed above can be seen in use in the following piece of code. A function $f(x) = \mathrm{sech}(2x)$ was used as an example. Any arbitrary function of vector of values can be differentiated with these methods.

```
f = @(x) sech(2*x); % Function
fp = @(x) -2.*tanh(2*x).*sech(2*x); % Exact derivative (for reference)
N = 21; x = linspace(-1,1,N); h = x(2)-x(1);
% Wavevector for fft derivative
v = linspace(0,1-1/N,N) + (mod(N,2)/N-1)/2; v = 2*pi*fftshift(v)./h;
% Take the derivatives
fd_derivative = gradient(f(x),h); % Finite difference
fft_derivative = real(ifft(1i.*v.*fft(f(x)))); % FFT
cs_derivative = imag(f(x+1i*h))./h; % Complex step
```

Listing 2.33: Numerical derivative methods.

The function `gradient()` is an implementation of finite differences in Matlab. `gradient` uses central difference in the bulk of its input, and forward/backwards differences on the edges where central difference is undefined. Note that all of the methods discussed have used a *constant* step size h on a *uniform* grid. Numerical differentiation on non-uniform grids is, by all means, possible, but significantly more involved, and will not be discussed here.

Timestepping

The methods outlined here allow one to calculate the derivative with respect to any variable (e.g. space, time, magnetic field). The finite difference formulae can also be rearranged such that the value of the function of interest at the point incremented by the step can be calculated, if the derivative is know. This is the basis of *time-stepping* algorithms which are one of the basis of solving differential equations numerically. Equation 2.20 can be used directly to calculate the value of f a step h away, based only on the value of $f(x)$ and $f'(x)$. Only the value of the initial point, and a method of calculating the derivative – the differential equation itself, are needed. These type of problems are referred at as the initial value problems (IVPs).

[dj] This relationship can be proven by differentiating under the integral of the definition of the FT.

The simplest way to attempt to solve IVP is referred to as the *forwards* Euler (FE) method. To illustrate, lets consider a well-studied differential equation, the Van der Pol equation:

$$\ddot{y} - \mu(1-y)\dot{y} + y = 0 \tag{2.26}$$

This is a second order differential equation. For large values of μ this equation results in periodic oscillations of y with sharp corners – a good test of the differential equation solvers. Since this is a second order equation, it has two be expressed as two coupled equations:

$$\begin{aligned} \dot{y}_1 &= y_2 \\ \dot{y}_2 &= \mu(1-y_1)y_2 - y_1 \end{aligned} \tag{2.27}$$

If y_1 and y_2 are considered as different elements of the same vector, Equation 2.27 can be written in a vector form – a standard form for setting up differential equations in Matlab. The Van der Pol equation initial value problem (IVP) be expressed as:

```
mu = 4; % Nonlinearity coefficient
dydt = @(t,y) [y(2);mu*(1-y(1)^2)*y(2)-y(1)];
k = 0.1; % Time step
t = (0:ceil(15./k)).*k; % Time axis
y0 = [1 1]; % Initial condition
```

Listing 2.34: Setup of the Van der Pol equation

The differential equation itself is a vector function evaluated with time and current point as the input (even though, in this case, time is not a variable). The solution to this problem using the forwards Euler method can be found very simply by marching forward in time and calculating the derivative according to `dydt` at each time point:

```
y = zeros(numel(t),2); y(1,:) = y0;
for ti = 2:numel(t)
    dy = k.*dydt(t(ti),y(ti-1,:))'; % Change in y is the current gradient
    y(ti,:) = y(ti-1,:) + dy; % Go forward in time
end
```

Listing 2.35: Forward Euler timestepping

The plot of `y` as a function of `t` can be seen in Figure 2.17, together with the 'exact' solution. The FE method overestimated the step which needs to be taken and therefore it lags behind the correct oscillation (it oscillates too slow). Further, when the time-stepping passes a sharp corner of the Van der Pol oscillator, an instability is induced in the solution. This instability blows up on the second turn of the equation and the solution diverges to infinity. The set of initial conditions chosen here is, in fact, outside of the stability of region of the FE method. The detailed considerations of finite-difference scheme *stability* and *accuracy* are not going to be explored in depth here, but ought to be considered carefully, especially when using self-written differential equation solvers.

Just like there are many finite difference methods, there are multiple time-stepping schemes. For example time stepping based on the backwards difference is referred to as the *backwards* Euler (BE) method.[dk] The equation for the timestep for the BE time-stepping (equivalent of Equation 2.20 for FE) is:

$$f(x+h) = f(x) + hf'(x+h) \tag{2.28}$$

The value of the function a time step away is therefore the value that it is 'now' plus a term depending on derivative a time step away. Since the value of the function there is not known one might say it is an impossible problem to solve and no time-stepping can be done based on the backwards difference. This is

[dk] The FE and BE methods are sometimes referred to as the *explicit* and the *implicit* Euler method. This is done especially in cases where there concept of directionality (forwards/backwards) is ill-defined.

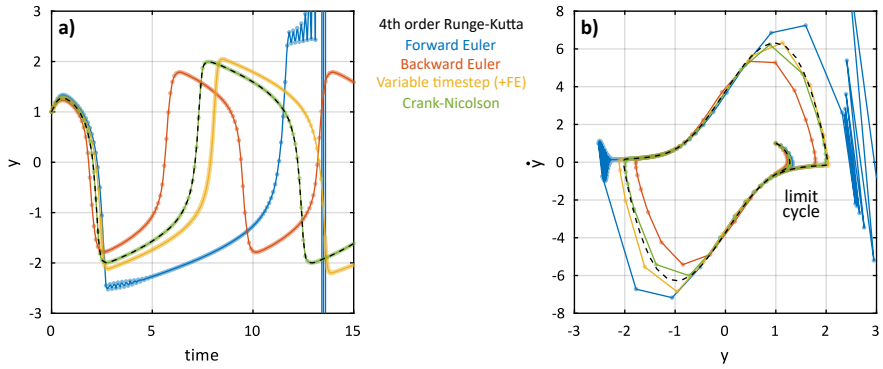


Figure 2.17: Solution to the Van der Pol IVP with different timestepping methods. **a)** The time evolution, and **b)** the limit cycle are shown for the forward Euler (blue), backward Euler (red), variable timestep (yellow) and Crank-Nicolson (green) methods are shown. The reference (black) is the 4$^{\text{th}}$ order Runge-Kutta solution.

however not the case. The value of $f'(x_h)$ can be found for example by fixed-point iteration – guessing $f'(x_h)$, calculating $f(x_h)$ and recalculating $f'(x_h)$ until the answer does not change – or by estimating the value of $f(x_h)$ with a different method, and therefore calculating $f'(x_h)$. This is referred to as a *predictor-corrector* method. The BE iteration on the Van der Pol equation above can be performed as follows:

```
y = zeros(numel(t),2); y(1,:) = y0;
for ti = 2:numel(t)
    % Fixed-point iteration to find change in y
    dy = 0; % Initial guess
    for j = 1:10, dy = k*dydt(t(ti),y(ti-1,:)+dy)'; end
    % Predictor corrector method
    yf = y(ti-1,:)+k.*dydt(t(ti),y(ti-1,:))'; % FE step
    dy = k*dydt(t(ti),yf)';
    % Go forward in time
    y(ti,:) = y(ti-1,:) + dy;
end
```

Listing 2.36: Backward Euler timestepping.

Both the fixed-point iteration (in this case fixed to 10 iterations) and predictor-corrector methods have been shown. The BE method is referred to an an *implicit* method – it depends implicitly on its own result. This is in contract to the FE iteration which is an *explicit* method – each timestep can be explicitly calculated. Implicit methods tend to have much larger stability region but can be more computationally expensive. The plot the BE solution can also be seen in Figure 2.17. The solution does not diverge, but since the gradient is underestimated at each time step, it quickly goes out of phase with the actual solution (it oscillates too fast).

So far all the methods used a constant time increment. A simple scheme like FE could be made much more accurate if the size of the step is dropped whenever the change in y exceeds a certain threshold.[dl] Such methods are referred to as variable-timestep (VT) and are very commonly implemented in differential equation solvers. A simple implementation of VT to Listing 2.35 can be done as follows:

```
y = y0; t = 0; t_end = 15;
while t(end)<t_end
    dt = k; % Start with the given timestep
    while true % Drop the timestep until dy is small enough
        dy = dt.*dydt(t(end),y(end,:))'; % Change in y
        if all(dy<1e-2) || dt<=1e-3, break, end % Stopping condition
        dt = dt/2; % Drop the timestep
    end
    y(end+1,:) = y(end,:)+dy; % Go forward in time
    t(end+1) = t(end) + dt; % Record the time too
```

[dl] Arbitrary accuracy cannot be achieved in this way since, as the size of the step decreases, different time-stepping schemes will eventually reach optimal accuracy and any further decrease in the size of the timestep will in fact *increase* the error. This is not to mention that the iteration with smaller timestep takes much longer to complete.

```
end
% Interpolate onto an evenly spaced time axis
t_even= linspace(0,t_end,1e3); % New time axis
y = interp1(t,y,t_even); t = t_even;
```

Listing 2.37: Variable timestep + FE.

The timestep is halved until the change in `y` is less than 10^{-2}, with the minimum timestep of 10^{-3}. Note that the `for` loop had to be replaced by a `while` loop since the number of iterations is, initially, unknown. The time axis also had to be recorded over the course of iteration. If needed, the result can be interpolated onto an even time axis after the iteration. The advantage of this method over a simple FE can be clearly seen in Figure 2.35. The VT does not diverge and, in fact, it is the closest to the actual solution as of yet.

The combination of forward and backwards finite differences led to a construction of a center difference in Equation 2.22. A similar combination of FE and BE methods can be used for timestepping. Since FE overestimates, and BE over estimated the gradient and therefore the size of the step, evaluating both and taking the average ought to be a good idea. This scheme is referred to as the Crank-Nicolson (CN) method,[dm] and can be easily programmed as follows:

```
y = zeros(numel(t),2); y(1,:) = y0;
for ti = 2:numel(t)
    dy_fe = k.*dydt(t(ti),y(ti-1,:))'; % FE step
    dy_be = k*dydt(t(ti),y(ti-1,:)+dy_fe)'; % BE time step
    dy = (dy_fe+dy_be)/2; % Average timestep
    y(ti,:) = y(ti-1,:)+dy; % Go forward in time
end
```

Listing 2.38: Crank-Nicolson timestepping.

The BE timestep has been evaluated using already calculates FE step and the predictor. The result of CN timestepping can too be seen in Figure 2.22. Iteration can be seen to follow closely the exact result, this without the use of VT.

Finally, the 'exact' solution has been found using a variable timestep 4^{th} order Runge-Kutta (4RK) method this is a highly accurate method based, as the name suggests on a 4^{th} order Taylor expansion of $f(x)$. The variable timestepping is done by comparing the result with a 5^{th} order timestep, and therefore evaluating the error on the 4^{th} order step. This would be a very complicated and error-prone scheme to implement oneself. Fortunately Matlab has a multitude of built-in solvers, 4RK being one of them. The exact formulation of the problem from Listing 2.34 allows the solution to be found by calling:

```
[~,ye] = ode45(dydt,t(end),y0);
```

Listing 2.39: 4^{th} order Runge-Kutta timestepping.

Matlab implements other solvers and like the variable order solver `ode113` which is an excellent all-purpose solver, and the stiff problem solver `ode15s` which ought to be used for problems where the elements of `y` can vary over many orders of magnitude in the same time step (e.g. chemical kinetics problems).

PDEs

The Van der Pol oscillator is an example of a ordinary differential equation (ODE) – the variable y appears as derivatives of one other variable only, in this case time. Many problems of interest are, however, expressed in terms of partial derivatives with respect to many different dimensions. These are referred to as partial differential equation (PDEs) correspondingly. The examples of such equations are the heat equation – governing the diffusion and redistribution of heat – and the wave equation – governing the propagation and interactions of waves. The wave equation, in particular, is at the core of the description of the electromagnetic (EM) radiation, and is therefore of particular interest here.

[dm] Named after 20^{th} century English mathematicians Phyllis Nicolson and John Crank who pioneered this approach.

The classical description of electromagnetism constitutes a set of four coupled partial differential equations relating the shape and time evolution of coupled electric and magnetic fields. These are the Maxwell's equations and, if there are no static electric charges present, they can be written as follows:

$$\nabla \cdot \mathbf{E} = 0 \qquad (2.29a) \qquad \nabla \cdot \mathbf{B} = 0 \qquad (2.29b)$$

$$\nabla \times \mathbf{E} = -\dot{\mathbf{B}} \qquad (2.29c) \qquad \nabla \times \mathbf{B} = c^{-1}\dot{\mathbf{E}} \qquad (2.29d)$$

where $\nabla\cdot$ is the divergence and $\nabla\times$ is the curl operator, $\mathbf{E}$ and $\mathbf{B}$ are the electric and magnetic vector fields respectively, t is the time dimension and c is the speed of light. Divergence of a field is a sum of its first derivatives with respect to the spatial coordinates (it is a scalar function), and curl is a torque which the given field exerts at each point (it is a vector function).[dn] Equations 2.29c and 2.29d describe the dynamics of the electromagnetic field. It is not necessary though to simulate both the electric and magnetic field. Taking the curl of both sides of Equation 2.29c gives:

$$\nabla \times \nabla \times \mathbf{E} = -\nabla \times \dot{\mathbf{B}} \qquad (2.30)$$

The curl of a curl is the difference between the gradient of the divergence and the Laplacian. Further, the partial time derivative operator commutes with the curl. The above equation becomes:

$$\nabla(\nabla \cdot \mathbf{E}) - \nabla^2\mathbf{E} = -\frac{\partial}{\partial t}(\nabla \times \mathbf{B}) \qquad (2.31)$$

Using Equations 2.29a and 2.29d yields:

$$\ddot{\mathbf{E}} = c^2\nabla^2\mathbf{E} \qquad (2.32)$$

This is a partial differential equation for just the electric field component of the EM radiation. It is also a wave equation for a wave travelling with speed c.

The electric field in 2D can be represented as a discrete square N×N grid `E` of points representing the *magnitude* of the electric field E. The second derivative of E in each direction, and at each point can be calculated using a finite difference formula from Equation 2.23. If the grid `E` is vectorised – unwrapped into a long 1D vector `e`, the action of taking its finite difference second derivative can be represented as a multiplication by a matrix. The Laplacian of E can, therefore, also be expressed as a matrix. The setup of the problem looks as follows:

```
N = 100+1; % Number of discretisation points
x = linspace(-1,1,N); h = x(2)-x(1);
[X,Y] = meshgrid(x); % x and y points on a grid
E = zeros(N,N); % Inital electric field (none)
% Construct the laplacian operator
D = spdiags(bsxfun(@times,[1 -2 1],ones(N^2,1)),[-1 0 1],N,N);
D(1,end) = 1; D(1,end) = 1; D = D./(4*h^2); % Periodic boundry
L = kron(D,speye(N)) + kron(speye(N),D); % Sum of first derivatives
lap = @(e) L*e; % Numerical laplacian of vectorised E
```

Listing 2.40: EM radiation setup

The x and y derivatives of the vectorised `E` have been obtained by taking Kronecker products of the 1D derivative with an identity matrix of an appropriate size. Note that the derivative and the Laplacian operators are *sparse* and therefore have been implemented as such. This is crucial, as otherwise the size of `L` in memory would approach 0.5 GB.

The wave equation can be started with some interesting initial state and left to evolve in time. It is of much more use, however, to be able to apply a driving force to it. How much energy actually gets transferred from the driver to the electric field is a function of frequency of the driver, its exact profile (shape) and the speed

[dn] Curl of a vector field describes the nature of the rotation of the field at each point. It can be defined in terms of a line integral through an infinitesimal loop around each point of the field, but analytically it is most commonly described as the 'cross product' of the del operator with the field, hence the symbol $\nabla\times$.

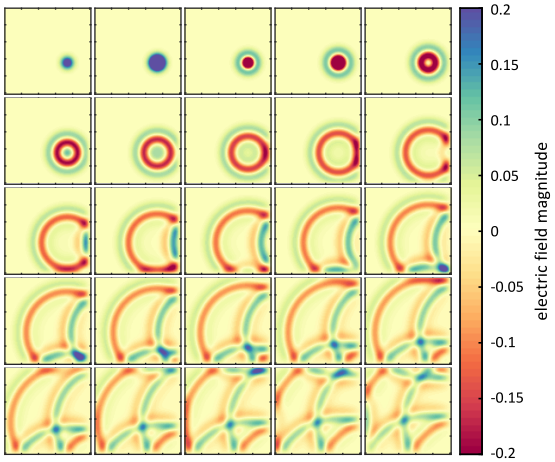


Figure 2.18: Time evolution of the electric field, subject to a brief driving force, as calculated in Listing 2.42. A 4 Hz driver is applied at point [–0.5, –0.3] for a period of single oscillation starting at time $t = 0$.

of light in the medium. The exact characteristics of that have not been modelled here. An example driver function – a single oscillation of a 2D Gaussian located at $x = 0.5$ and $y = -0.3$ – can be formulated as follows:

```
f = 2; % Driver frequency
denv = exp(-((X-0.5).^2+(Y+0.3).^2)*1e3); % Driver envelope
d = @(t) denv(:)*20*sin(2*pi*t*f)*(t<1/f); % Driving function
```

Listing 2.41: Driving force.

The wave equation for the electric field subject to this driving function can be be simulated as follows:

```
v = 1; % Phase velocity
e0 = E(:); e0 = [e0; lap(e0)]; % Starting point
dedt = @(t,e) [e(N^2+1:end) + d(t); 2*v.^2.*lap(e(1:N^2))];
[t,U] = ode113(dedt,linspace(0,2.5,1e2),e0);
% Reshape back to 2d coordinates
for ti = 1:numel(t)
    U_square(:,:,ti) = reshape(U(ti,1:N^2),N,N);
end
```

Listing 2.42: Reflecting gaussian.

The results can be seen in Figure 2.18. The EM wave can be seen to propagate inside of the unit box as expected. It can also be seen to pass *through* the boundary and out the other side of the box. This is due to the *periodic* boundary conditions imposed by the derivative operator `D` in Listing 2.40. These are the easiest boundary conditions to implement,[do] and hence have been used here.[dp] Note that, in addition to vectorising `E`, a method akin to that used in Listing 2.34 has been used to express the second order differential equation as a first order one. The problem solved above is thus, effectively, a system of $2N^2$ coupled differential equations (ca. 20k equations for $N = 101$ used here).

The actual problem of interest here is that of interaction of a beam of light with a thin piece of dielectric of high refractive index (low speed of light). Certain wavelengths of light ought to resonate within the piece of dielectric and create a standing wave pattern, while certain others ought not to, depending on the width of the dielectric and the frequency of the input radiation. This resonance phenomena is used in making of dielectric mirrors and filters, as well as in certain types of spectroscopy as discussed in Section 5.1. An example problem for this can described by the following ODE function:

```
function dedt = wave_odefun(t,e,f,v,X,Y,N,lap)
s = 20; % Sigmoid step gradient
% Sigmoid envelope in X and Y
SY = sigmoid_step(Y,1,0,0.5,s).*(Y>0)+sigmoid_step(Y,0,1,-0.5,s).*(Y<=0);
```

[do] Other boundary conditions could, for example be an *absorbing* or *reflecting* boundary.
[dp] Note that this type of periodic boundary implies the 2D surface of the simulation lies on a torus, not on a ball.

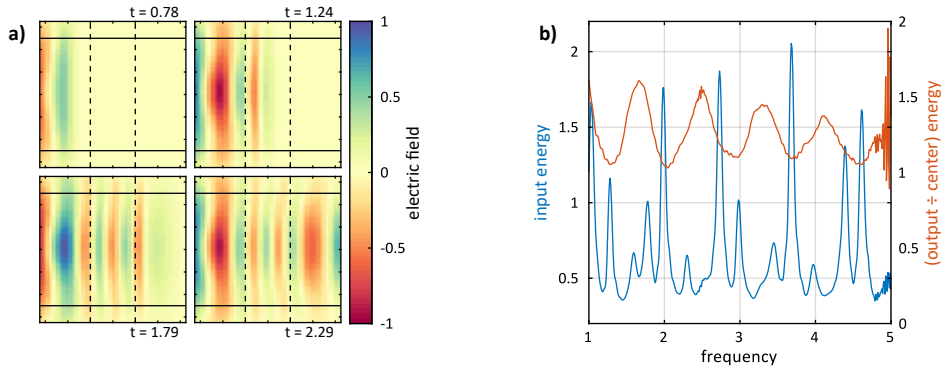


Figure 2.19: Propagation of an electric wave through a dielectric and the resulting resonant structure **a)** The magnitude of the electric field at selected times. The wave can be seen to slow down and increase in frequency upon entering the dielectric layer (index of refraction $n = 2$). Only the ± 0.65 region of a domain is shown. The departure of the wavefront shape from a plane wave occurs due to residual reflections from the absorptive boundaries. **b)** Root-mean-square energy of the wave at the input (left-hand side of the dielectric layer, plotted in blue) and the ratio of output energy to the energy in the dielectric layer (red), as a function of frequency of the input. Note that at high frequencies the grid discretisation cannot support the wave frequencies (Nyquist limit) which leads to invalid finite-difference Laplacian calculation. This can be seen on the right-hand side of the graph.

```
SX = sigmoid_step(X,1,0,0.8,s).*(X>0)+sigmoid_step(X,0,1,-0.8,s).*(X<=0);
n = ones(size(X)); n(X<0.2&X>-0.2) = 2; % Index of refraction landscape
S = SX.*SY; l = (1-S(:)); % Loss constructed based on the X and Y envelope
d = 2*pi*f*cos(2*pi*t*f)*0.5; % Driver magnitude
[~,I] = min(abs(X(1,:)-(-0.75))); % Find x index closest to -0.75
isd = false(N,N); isd(:,I) = true; % Select only the x = -0.75 slice
e1 = e(N^2+1:end) - 50*l.*e(1:N^2); % First derivative + losses
e1(isd(:)) = e1(isd(:)) + d*exp(-Y(:,1).^2*10); % Add the driver
e2 = (2*v./n(:)).^2.*lap(e(1:N^2)); % Seccond derivative - wave equation
dedt = [e1; e2]; % Overall derivative of e
end
```

Listing 2.43: EM radiation ODE function

This problem would require an unfeasibly large domain to be solved with periodic boundary conditions. Instead of redefining `D`, a heavy loss function has been added to the first derivative of `e`, for $|x| > 0.8$ and $|y| > 0.5$. To prevent reflections from the sharp boundary, this has been some with smooth piecewise sigmoid steps. An example solution to this problem has been shown in Figure 2.19a. The wave propagation slows down when it encounters the surface of the dielectric, and hence its frequency increases.

Solving the above problem for a range of the driver frequencies yields a plot in Figure 2.19b. First, the average peak-to-peak magnitude (proportional to energy) of the input radiation is plotted. It can be seen to exhibit a resonance structure, but this is not only due to the resonance within the dielectric layer. The reflection of the surface of the dielectric interferes with the incoming radiation and creates its own standing wave. Also, as aforementioned, the actual energy the driver will impart to the electric field is a function of many parameters inducing frequency. The second plot in Figure 2.19b shows the ratio of the average output energy of the electric field to the energy contained within the dielectric layer. This can be seen to exhibit a clear resonance pattern. A standing wave within the dielectric layer creates a constant oscillating field at its far edge. This effectively drives the electric wave leaving the dielectric. When no standing wave is set up the field at the edge of the dielectric behaves more erratically and the magnitude of the resulting oscillation is lower.

Boundary value problems

The time-dependent Schrödinger equation is also a PDE – connecting the time evolution of the wavefunction describing the system to the result of application of the Hamiltonian operator. Depending on the exact

formulation of the problem at hand, the latter often contains a Laplacian – sum of second derivatives of the wavefunction in space. The initial conditions for the Schrödinger equation are, however very different. Instead of specifying the boundry at one end of the domain (either in time or space), the solution adhering to multiple boundary is required. These type of problems are often referred to as boundary value problems (BVPs), and are typically much harder to solve than IVPs.

An example of such problem is a quantum particle in a potential well. Such particle is described by a wavefunction ψ subject to boundary conditions $x = 0$ at $x = \pm\infty$. The statistic (time-independent) variant of this problem will be solved here, where the solutions are stationary states of the system Hamiltonian – eigenstates. The Hamiltonian for a 1D particle of mass m, in a potential well $V(x)$ can be obtained by adding up the kinetic energy, and the potential energy Hamiltonians. The total Hamiltonian is:

$$\hat{\mathrm{H}} = \hat{\mathrm{H}}_{\mathrm{KE}} + \hat{\mathrm{H}}_{\mathrm{PE}} = \frac{\hat{p}^2}{2m} + \hat{V}(x) \tag{2.33}$$

Akin to the approach described in 1.1, the potential energy operator can be obtained by quantum transcription of the classical expression for potential energy (replacement of $\hat{\boldsymbol{p}}$ – the linear momentum operator by $-i\hbar\hat{\boldsymbol{\nabla}}$). The eigenvalues of the Hamiltonian are the energy levels of the stationary eigenstates (solutions to time-independent Schrödinger equation). Performing the quantum transcription, and applying the Hamiltonian to an eigenstate ψ gives:

$$-\frac{\hbar^2}{2m}\ddot{\psi} + \hat{V}(x)\psi = E\psi \tag{2.34}$$

where the $\ddot{\psi}$ indicates the second derivative of the eigenstate with respect to space coordinate. To solve such an eigenvalue problem is to find the set of functions ψ and the corresponding multiplicative constants E such that the equality holds. These are referred to as eigenstates and eigenvalues. Equation 2.34 can be solved analytically for simple forms of the potential function like $\hat{V}(x) = x^2$, but not in general (for arbitrary $V(x)$). Lets assume, for a moment that E is known, and therefore lets rewrite Equation 2.34 as a simple ordinary differential equation boundary value problem (BVP):

$$\ddot{\psi} = \frac{2m}{\hbar^2}\left(\hat{V}(x) - E\right)\psi \tag{2.35}$$

subject to $\lim_{x\to\pm\infty}\psi(x) = 0$. The boundary condition originated from the fact that the square of the wavefunction is the probability density of the particle in question. The wavefunction ψ must be therefore subject to normalisation condition $\langle\psi|\psi\rangle = 1$. Since the expectation value integral is bound (finite), the wavefunction itself must vanish at infinity.

Such *boundary value problem* will have a solution only when E is an eigenvalue of the original problem as defined by Equation 2.34. One way to tackle such a boundary value problem is to solve it numerically as an *initial value problem*, with the initial conditions being the value at one boundary, and use a normal ODE solver to step the solution to the other boundary. This can be done for an arbitrary E, and the value of E can be adjusted until a solution satisfying both boundary conditions is found. Computationally, the numerical solution to Equation 2.35 cannot be evaluated on an infinite domain. The domain can therefore be limited to $x \in [-L, L]$ and appropriate boundary conditions can be inferred. Further, the use of Matlab solvers requires one to express the differential equation in terms of first derivatives only. This can be done by introducing a second variable $\Phi = \dot{\psi}$. The second derivative of the wavefunction is therefore $\ddot{\psi} = \dot{\Phi}$, and Equation 2.35 can be rewritten as a first order differential *vector* equation:

$$\begin{bmatrix}\dot{\psi}\\ \dot{\Phi}\end{bmatrix} = \begin{bmatrix}0 & 1\\ \frac{2m}{\hbar^2}\left(\hat{V}(x) - E\right) & 0\end{bmatrix}\begin{bmatrix}\psi\\ \Phi\end{bmatrix} \tag{2.36}$$

To demonstrate the solution of this problem lets consider the harmonic oscillator potential, defined as:

$$V(x) = \frac{m}{2}\omega^2 x^2 \tag{2.37}$$

A particle of mass m will oscillate in this potential at angular frequency ω. The analytical solutions to the eigenvalue problem as defined by Equation 2.34 with this V are known; the k^{th} eigenstate is based on k^{th} Hermite polynomial, and the k^{th} eigenvalue is of the form $E_k = \left(\frac{1}{2} + k\right)\omega$. To solve this problem numerically is is convenient to work in atomic units. The setup of the problem is:

```
% Atomic units
hbar = 1; m = 1; % Mass
omega = 1; % Oscillator frequency; Hartree per hbar (per radian)
L = 5; % Domain size
xx = linspace(-L,L,1e3); dx = xx(2)-xx(1);
V = @(x) m/2*omega.^2*x.^2;
```

Listing 2.44: Quantum harmonic oscillator setup.

The energy guess then starts at zero, the problem is set up according to Equation 2.36, and is solved with initial conditions of $\psi = 0$ and $\Phi = \sqrt{\frac{2m}{\hbar^2}(V(-L) - E)}$. If the solution *overshoots* zero the energy guess is too low, if it undershoots the guess is too high. This behaviour is flipped for odd solutions, hence one can climb the ladder of solutions by altering the energy optimisation behaviour parity. This can be programmed as follows:

```
e = 0; de = 0.1; % Energy guess and energy step
tol = 1e-6; % Wavefunction tolerance
N = 5; % Nuber of states to calculate
Psi = zeros(length(xx),Ns); E = zeros(1,N); % Preallocate
for Ni = 1:N
    while true
        dpsidx = @(x,psi) [0 1;2*m/hbar.^2.*(V(x)-e) 0]*psi;
        psi0 = [0 sqrt(2*m/hbar.^2.*(V(-L)-e))]; % psi at x = -L
        [~,psi] = ode45(dpsidx,xx,psi0); % 4-th order Runge-Kutta solver
        psi = real(psi(:,1)); psi = psi./sqrt(sum(psi.^2).*dx); % Norm
        if abs(psi(end))<tol || de<eps, break; end
        if psi(end)*(-1)^(Ni+1) > 0, e = e+de; % Increment energy
        else, de = de/2; e = e-de; % Decrement energy
        end
        % Zero-off exponential runaway
        psi2 = log10(psi(xx>0).^2); xx2 = xx(xx>0); % Right side of psi
        f = @(x) interp1(xx2,psi2,x);
        x0 = fminsearch(f,xx(end)); psi(xx>=x0)=0;
        psi = psi./sqrt(sum(psi.^2).*dx); % Renormalise
    end
    Psi(:,Ni) = psi; E(Ni) = e; % Store the wavefunctions and energies
    de = 0.1; e = e+de; % Reset step size
end
```

Listing 2.45: Shooting method.

The results are shown in Figure 2.20a. The values of energy do indeed match those obtained by the analytical solution, i.e. $0.5, 1.5, 2.5, \ldots$ etc. (note the zero-point vibrational energy of $\hbar\omega/2$). Since the numerical solutions are independent of the exact form of the potential function, this problem can also be solved in an arbitrary potential well. The potential well can be discontinuous and non-smooth, as long as it is finite over the domain of x. A solution for a potential akin to Lennard-Jones chemical bond potential can also be seen in Figure 2.20b.

Note that the initial boundary condition is derived based on assumption that $E \ll V(\pm L)$. This can be seen to break down for higher energy solutions of the second potential function in Figure 2.20 where the gradient of the solution does not go to 0. The domain of x can, however, always be extended arbitrarily (by increasing

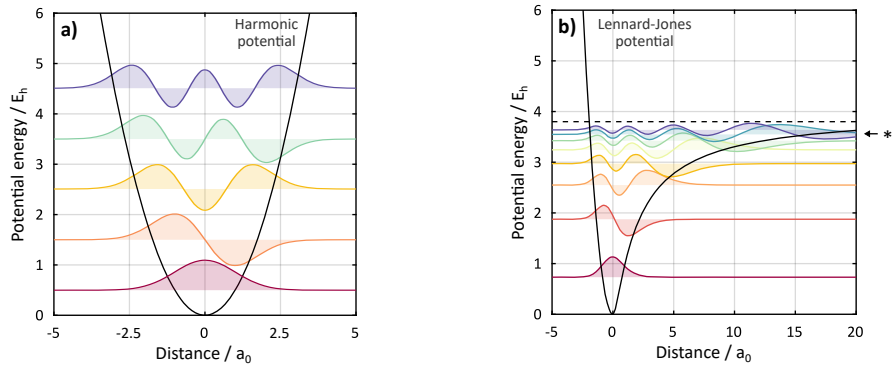


Figure 2.20: Solution to 1D time-independent Schrödinger equation for a particle in an arbitrary potential well. The eigenvalue problem is expresses as a boundary-value problem, and solved by a shooting method. **a)** Harmonic potential well $V(x) = {}^1\!/_2 m\omega^2 x^2$ with $m = 1\,\mathrm{m_e}$ and $\omega = 1\,\mathrm{E_h\,rad^{-1}}$. The solutions are spaced by $\hbar\omega$ with zero-point energy of ${}^1\!/_2\hbar\omega$. **b)** Lennard-Jones-like potential well $V(x) = \begin{cases} x^2 & x \leq 0 \\ 4x^2(1+x)^{-2} & x \geq 0 \end{cases}$. Note the uneven energy spacing of the energy levels as the potential departs from harmonicity. The two solutions of highest energy (indicated with a $*$) have not converged properly since the boundary condition assumption of $V \gg E$ no longer holds.

L) and, as long as $\lim_{x\to\pm\infty} V(x) = \infty$, the problem can be solved. An issue of numerical stability of the algorithms may, however, arise for larger domains. The solution may still diverge for an energy step `de` smaller than machine precision `eps`. This is because the general (analytical) solution is a difference of two exponential terms which cancel out only when the eigenvalue equation holds. Accumulated numerical inaccuracy of the solvers may be enough for one of these terms to not cancel out exactly within the $x \in [-L, +L]$ domain. The solution eigenstates may, therefore, need to be adjusted to fall to 0 at the edges of the domain. This has been done in the Listing 2.45 by finding the first point on the right-hand side which falls off to zero, and setting the wavefunction to the right of it to zero.[dq]

The solutions to the Schrödinger equation of a particle in a harmonic potential are also applicable to diatomic vibrations. The vibration of two atoms can be considered as that of a single *reduced mass* about a mutual center of mass. This vibration occurs on a single motion coordinate and thus obeys the solutions to the 1D Schrödinger equation. Using the Lennard-Jones potential, one can reason about the nature of the bond, aharmonicity and bond-making/breaking dynamics (if x is considered as a generalised reaction coordinate). Furthermore, electron-transfer reactions can be thought to be governed by the overlap of the 1D wavefunctions along the generalised motion coordinate of the species involved (see Section 1.2).

Chaos

Finally, there is an important aspect of differential equations which ought to be demonstrated. The solutions to certain problems are uncertain, not due to the inaccuracy of the method used, but due to nature of the problem itself. A great demonstration of this type of problem is the so-called 'magnetic pendulum'. This device consists of a magnet suspended by a long thread over a board with other magnets attached to it. The magnets are arranged such that the hanging magnet is attracted to all of the magnets below it. If the magnet on the pendulum is displaced from the equilibrium and let go, it will come to rest over one of the magnets below. If this is repeated, with the magnet being let go from, seemingly, the same point the result is, most likely, different.

The details of this system, as well as its description in terms of differential equations, can be found is the work of Christian and Middleton-Spencer,[68] and will not be recounted here in excess detail. Briefly, the pendulum can be modelled as a point in complex plane $\boldsymbol{x}$ subject to a restoring force due to gravity pulling it towards the origin and force proportional to the distance away from each of the fixed magnets. Additionally, a damping term proportional to $\dot{\boldsymbol{x}}$ has to be added in order to dissipate energy and allow the pendulum to

[dq] This procedure is not actually necessary for the specific `L`, `Ns` and `V` given, but has been included here since this code is used in Section 1.2 to calculate higher order solutions which do exhibit exponential runaway.

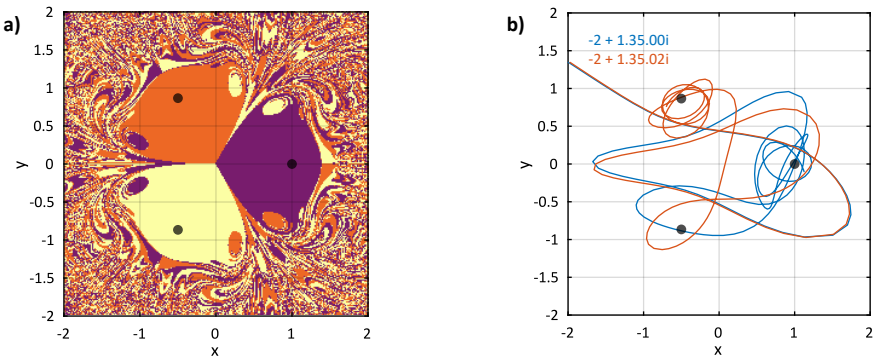


Figure 2.21: a) Basins of attraction for a magnetic pendulum, as calculated in Listing 2.46. **b)** Trajectories for two starting points in the chaotic region, differing by $0.02i$. The paths can be seen to be very sensitive to the initial conditions – any deviation becomes amplified by all the successive interactions.

come to rest. The magnetic pendulum can be modelled by the following differential equation:

$$\ddot{\boldsymbol{x}} + b\dot{\boldsymbol{x}} = \sum_n \left\{ \frac{\mathbf{X}_n - \boldsymbol{x}}{(|\mathbf{X}_n - \boldsymbol{x}|^2 + h^2)^{\frac{5}{2}}} \right\} - \boldsymbol{x} \tag{2.38}$$

where b is the damping constant, and h is related to height of the pendulum above the board of magnets. The following is a simulation of this ODE over a grid of possible starting positions:

```
b = 0.1; h = 1/2; n = 3; % Damping, height and number of magnets
X = exp(1i*(0:(n-1))*2*pi/n); % Fixed magnets
S = @(x) sum((X-x(1))./(abs(X-x(1)).^2+h^2).^(5/2)); % Distance funciton
dxdt = @(t,x) [0 1;-1 -b]*x + [0;1]*S(x(1)); % ODE function
rs = linspace(-2,2,2^8); is = rs; % Grid of starting points
C = zeros(numel(is),numel(rs)); % Matrix of the final resting positions
as = randperm(numel(C)); % Random permutation of indices through C
for ai = 1:numel(as)
    [ri,ii] = ind2sub(size(C),as(ai));
    x0 = [rs(ri)+is(ii).*1i; 0]; % Starting point
    [t,y] = ode23(dxdt,[0 20],x0); % Simulate using a cheap solver
    [~,C(ii,ri)] = min(abs(X-y(end,1))); % Find the resting position
end
```

Listing 2.46: Magnetic pendulum.

The plot of `C`, as well as a few select trajectories of the pendulum are shown in Figure 2.21a and 2.21b respectively.[dr]

Large basis of attraction exist around each of the fixed magnets, since 'letting go' of the pendulum there results in it not having enough energy to escape the potential well of attraction of the respective magnet. The final resting point of the pendulum is, however, very hard to predict if its starting point is moved outside of these basins. Note that this is not a property of the solver or the method uses, *but of the differential equation itself.* The outcome with each set of initial conditions is perfectly deterministic, but it is infinitely sensitive to all of the minutiae of these initial conditions. This behaviour is referred to as *chaos.*[69] Whether a system under question is chaotic needs to be carefully considered when attempting to solve a problem based on differential equations. Examples of chaotic systems are motion in a potential field (e.g. orbits of planets or charges orbiting a nucleus), fluid flow and behaviours of certain electrical circuits.[ds]

[dr] Note the similarity of Figure 2.21a to Figure 2.12b.
[ds] Two more famous examples of a chaotic systems are the *double pendulum* and the *Lorentz attractor.*

2.4 Uncertainty and its propagation

Due to nature of an experiment, there will always be an uncertainty associated with any measurement made. For the measurements of be valid it is *crucial* for one to evaluate this uncertainty, as otherwise the measurements might as well be considered infinitely uncertain, and therefore objectively of no value. A typical way one will attempt to quantify this uncertainty is to obtain multiple measurements of the same quantity, *or* assert that multiple 'neighbouring' measurement (either in time or in space) are in fact coming from the same underlying statistical distribution or a process, and hence can be analysed as such. Further, one often assumes (or knows) the probability distribution of the quantity in question to be *unimodal* - consist of a single maximum (derivative of the probability density function is crossing zero only once). Having obtained multiple measurements of a quantity, the underlying (unimodal) probability distribution can be easily modelled with two parameters: 1) its *central tendency* and 2) its *spread*. The most common quantifiers for these are the mean and the standard deviation,[dt] due to their well-defined values for many probability models, as well as their relationship with the normal distribution. Many of such measurements, if sufficiently uncorrelated, will approach the normal distribution by virtue of a *central limit theorem*.[du]. Since the mean and standard deviation are, in fact, the only degrees of freedom of a normal distribution, they are sufficient to fully describe the probabilistic nature of the uncertainty of the measurement in this simple case.

Most practical measurements can be analysed as if they were following a normal distribution, but a more careful analysis is sometimes needed. Other measures of central tendency and spread can be more appropriate. For a more accurate description of a skewed distribution, one can employ three parameters by describing it in terms of the central tendency, upper spread and lower spread. For such distributions, the median, as well as upper and lower standard deviation equivalence can be used (see Listing 2.32). This can be thought of as modelling the distribution as piecewise-normal - where the lower and upper halves are allowed to have different standard deviations (see Listing A.18). Note that this description is purely empirical. The piecewise Gaussian distribution has not been observed to arise naturally in any physical process.

Some thought ought to also be put towards the correlation between the neighbouring measurements. A measurement can be well described by a normal distribution in a purely probabilistic sense, while having either no, or significant correlation between the neighbouring samples. The three cases shown in Figure 2.22a have identical probability distributions, while having a strong positive, none, and strong negative correlation between the neighbouring samples.[dv]

Lets consider two experimentally measured quantities a and b with the (symmetrical) uncertainties σ_a and σ_b respectively. A function $f(a,b)$, not a and b themselves may, in fact, be a quantity of interest though. The uncertainties σ_a and σ_b will therefore need to be propagated though the function f to obtain the uncertainty in its value σ_f. The function f can be approximated, to the first order term by its Taylor expansion about the point $f(a,b)$:

$$f \approx f(a,b) + \left(\frac{\partial f}{\partial a}\right) a + \left(\frac{\partial f}{\partial b}\right) b \tag{2.39}$$

Any error in the measurement of a/b will therefore get scaled by the partial derivative of f with respect to a/b. The uncertainty in f is therefore proportional to the combined uncertainties σ_a and σ_b, scaled by the

[dt] Mean and standard deviation are referred to sometimes as the first and the second moment of a distribution respectively.

[du] The central limit theorem states that a sum of N independent random variables with *any* well-behaved probability distributions (no poles) approaches a normal distribution as $N \to \infty$. This can be seen in Figure 2.22b.

[dv] A correlated noise is referred to as 'coloured' noise. White noise has no correlation between the samples. Positive correlation corresponds to brown/pink noise and negative correlation to blue/purple noise. The colour of noise can be assessed by considering its power spectral density.

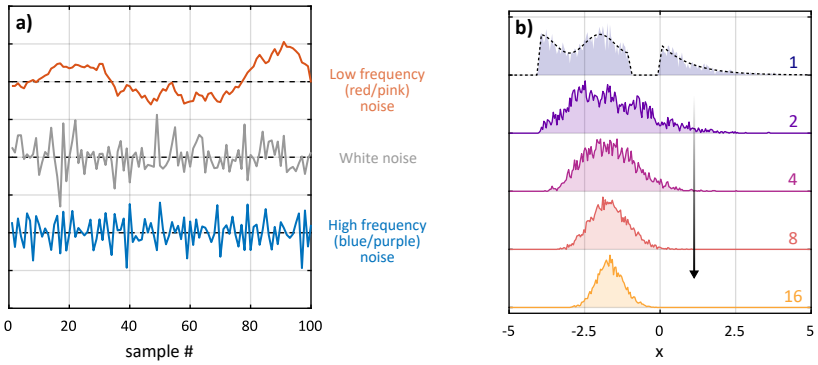


Figure 2.22: a) Examples of uncorrelated noise (shown in grey), as well as noise with strong positive and negative correlation between the neighbouring sample values (shown in red and blue respectively). Generated with `randnd` (see Listing A.5). **b)** Demonstration of the central limit theorem. A noise sampled from an artificial, non-white distribution (top) can be seen to approach a normal distribution when every 2, 4, 8 and 16 samples are averaged together. Note that the distribution is centered on the mean value of all the samples.

respective partial derivatives.

$$\sigma_f^2 \approx \left(\frac{\partial f}{\partial a}\right)^2 \sigma_a^2 + \left(\frac{\partial f}{\partial b}\right)^2 \sigma_b^2 + 2\left(\frac{\partial f}{\partial a}\frac{\partial f}{\partial b}\right)\sigma_{ab} \tag{2.40}$$

The term σ_{ab} is a cross correlation between a and b. For uncorrelated variables $\sigma_{ab} = 0$ and the last term can be discarded. The above equation becomes quite simple for simple functions. In case of addition/subtraction the uncertainties add in quadrature:

$$f(a,b) = a \pm b \qquad \rightarrow \qquad \sigma_f = \sqrt{\sigma_a^2 + \sigma_b^2} \tag{2.41}$$

In case of multiplication/division the *fractional* uncertainties add in quadrature.

$$f(a,b) = a \times b^{\pm 1} \qquad \rightarrow \qquad \sigma_f = |f|\sqrt{\left(\frac{\sigma_a}{a}\right)^2 + \left(\frac{\sigma_b}{b}\right)^2} \tag{2.42}$$

The assumption of discarding of the last term in Equation 2.40 can be easily demonstrated by setting $f(a,b) = a + b$ and $b = -a$. The Equation 2.41 predicts an uncertainty of $\sqrt{2}\sigma_a$ while the result is exactly zero since any error in a will get exactly cancelled out by the corresponding error in b. This is because b is very strongly correlated to a.

An example of this which is of specific interest in the magnetoreception research is the calculation of spectroscopically detected %MFE defined as:

$$\text{MFE} = \frac{I_{\text{on}} - I_{\text{off}}}{I_{\text{off}}} \times 100 \tag{2.43}$$

where I_{on} and I_{off} are the intensities of light with, and without the magnetic field applied respectively. One could be tempted to propagate the respective uncertainties, σ_{on} and σ_{off}, though the subtraction in the numerator, and then though the division. This will, however, heavily overestimate the uncertainty in %MFE since, even though I_{on} and I_{off} are assumed to be uncorrelated, the numerator of Equation 2.43 is correlated with the denominator (they are both a function of I_{off}). This problem can be fixed either by using Equation 2.40 directly, or by rephrasing Equation 2.43 as:

$$\text{MFE} = \left(\frac{I_{\text{on}}}{I_{\text{off}}} - 1\right) \times 100 \tag{2.44}$$

The numerator and the denominator are no longer correlated, and the fractional uncertainty in %MFE can be found. Some care must still be take with the order of operations though. The subtraction of 1 must, for

example be done *after* calculation of the uncertainty in the MFE. The following piece of code illustrates a correct order of operations by implementing Equation 2.42:

```
on_center = ⟨1 × n vector⟩; on_spread = ⟨1 × n vector⟩;
off_center = ⟨1 × n vector⟩; off_spread = ⟨1 × n vector⟩;
mfe_center = on_center./off_center;
% Fractional underainties add in quadrature
mfe_fractional_spread = sqrt(abs(on_spread./on_center).^2 + ...
                        abs(off_spread./off_center).^2);
mfe_spread = mfe_fractional_spread.*abs(mfe_center);
mfe_center = mfe_center-1; % No uncertainty in the '- 1' part
% Convert to %MFE
mfe_center = mfe_center.*100; mfe_spread = mfe_spread.*100;
```

Listing 2.47: Calculation of uncertainty of %MFE

Note that this is a very simple case where all of the light intensity measured is the reference for the %MFE. Equation 2.43/2.44 is applicable, for example, to the case of MFE measurement on a single fluorescing species, with no other fluorophores or sources of light present.

Finally, the demonstration of the central limit theorem in Figure 2.22b required the generation of random noise according to an arbitrary probability density function `pdf`. This task is non-trivial and, sometimes, impossible for a continuous distribution. A good approximation, however, can be made by interpolating a normalised cumulative distribution function with random uniform noise. This can be dome as follows:

```
% Make the probability density function
x = linspace(-5,5,1e2); % PDF discritisation axis
pdf = (x>-4 & x<-1).*(1+0.3.*cos(pi*x)) + (x>0).*exp(-x);
pdf = pdf./sum(pdf); % Normalise pdf
cdf = cumtrapz(pdf+eps); cdf(end) = 1; % Cumulative distribution function
% Generate random points according to pdf
N = 1e5; % Number of points to generate
y = interp1(cdf,x,rand(1,N),'linear'); % Interpolate the cdf
% Calculate the histogram
bins = linspace(-5,5,round(sqrt(N))+1); % Histogram bin edges
yh = histcounts(y,bins); % Count density of occurances in each bim
xh = bins(1:end-1) + diff(bins)/2; % Bin centers
```

Listing 2.48: Sample noise from an arbitrary PDF

The plot of `yh` against `xh` is a histogram of N random numbers `y` coming from `pdf`. Note that `eps` had to be added to the `pdf` when calculating the cumulative distribution function (CDF), in order to ensure the values of `cdf` were unique for interpolation. This code has been used in Figure 2.22 for calculation of the arbitrarily distributed noise.

Chapter 3

Development of overtone modulated MARY spectroscopy

In this chapter, significant developments will be reported in both the instrumental and data analysis aspects of modulated magnetically-altered reaction yield (MARY) spectroscopy. This chapter forms the main part an upcoming publication resulting from this work.[12] The principles behind the ModMARY spectroscopy will be discussed, and the details of the developed method for simulation of the modulated data, as well as its analysis will be reported.

Data on the Pyrene/1,3-Dicyanobenzene system has been acquired in collaboration with Olivia Foster Vander Elst (MChem student 2015/16). The initial concept and proof-of-principle data analysis methodology were developed by Dr. Jonathan Storey. Rigorous mathematical description, understanding and data analysis were developed by the author. The ModMARY experiment was constructed, and its control software was designed and programmed by the author.

Note that, since this chapter is based on a working manuscript, the style of it may be different to the other chapters. For that reason, and due to multiple components of the magnetic field being used, the notation used for the magnetic field is sightly different. The total magnitude of the magnetic field is denoted with b, while its DC and AC components are denoted with b_o and b_m.

3.1 Introduction to ModMARY spectroscopy

Modulated MARY (ModMARY) spectroscopy is a variant of fluorescence spectroscopy used for detection of magnetic field effects where the applied magnetic-field has both a static and a modulated component. It is most commonly employed for measurements of fluorescence MFEs on exciplex[70] and solid-state systems.[71] Due to the field-modulation employed, the signal detected is related to the derivative of the MARY curve. One of the utilities of ModMARY is therefore its ability to determine the width of the MARY curve with high accuracy and precision. This has been widely employed in studies of self-exchange in magnetosensitive systems based in both homogeneous[72] and micellar solutions,[73] and to investigate RP-solvent interactions.[74] Since ModMARY can easily determine the presence of a low-field effect it has also been used in exploratory studies of new polyarene-based compass chemical systems,[71] and in experimental investigations of the effects of the hyperfine coupling on the LFE.[75]

There are other, related, techniques which make use of the field-modulation and derivative detection. In recent years, ModMARY technique has been combined with laser scanning to yield both fluorescence[76] and absorption-based ModMARY imaging[77] (referred to in the literature as the magneto-fluorescence imaging (MFI) and magnetic intensity modulation (MIM) imaging respectively). The magnetic field modulation has further applications in fluorescence imaging with use of magnetically-modulated optical nanoprobes[78] – magnetic nanoparticles with spatially anisotropic fluorescence. The magnetic field modulation allows for rapid rotation of such nanoparticles and therefore their localisation with high S/N.[dw] The concept of magnetic-field modulation and derivative detection is also well-known in CW-EPR spectroscopy where it is used to enhance the S/N ratio of the observed microwave absorption/emission signals. The more general technique of stimulus-modulation can be seen in even wider range of fields of spectroscopy, notably in optical spectroscopy – frequency (wavelength) modulation using tuneable light sources. One of the

[dw] The interaction on an external modulated magnetic field with (magnetic) nanoparticles can also allow one to affect the microscopic environment around them.[79] This has many interesting applications in itself; from study of interactions in colloidal systems and self-assembly, to cell biology to localised drug delivery.

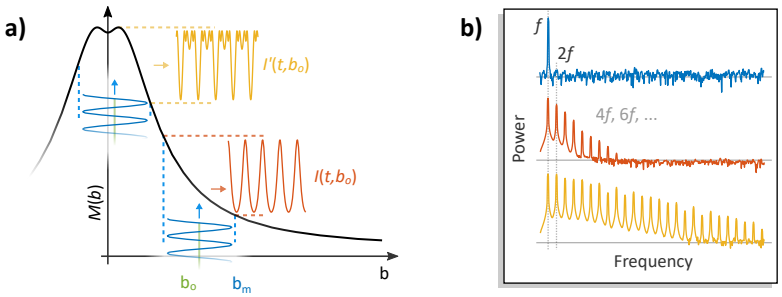


Figure 3.1: a) Representation of the MARY curve acting as a transfer function M from magnetic field (abscissa) to fluorescence intensity (ordinate). The magnitude of the modulated fluorescence signal I shown is approximately proportional to the derivative of M in the modulation region. However, I can undergo harmonic distortion due to the nonlinearity of M. **b)** Power spectral density spectra of the input field (blue) and I (red and yellow) the overtones of the fundamental frequency f can be seen. The more non-linear the transfer function (yellow) the more harmonic distortion. These spectra have been calculated with Listing A.22

most sensitive techniques in cavity-based optical spectroscopy (see Chapter 5) is one such technique – Noise-immune cavity-enhanced optical heterodyne molecular spectroscopy (NICE-OHMS).[80]

In ModMARY spectroscopy a fluorescing sample of interest is placed in an oscillating magnetic field (frequency f_m, amplitude b_m) and is continuously excited with an appropriate wavelength light in order to establish an oscillatory steady state fluorescence (see Figure 3.1). The magnetosensitive components of the fluorescence will then oscillate at the frequency (and overtones) of f_m. The amplitude of this oscillation is measured as a function of a static offset field (b_o). This could be done simply by notch filtering of a Fourier transform of the digitised signal. However, much narrower bandwidths, and thus better noise rejection, are usually achieved with lock-in detection.[81–84] The lock-in amplifiers (LIAs) are, therefore, the most common way of demodulating the fluorescence signal in ModMARY experiments.[70;73;74;85] The f_m component of the total applied magnetic field (b) is usually small and thus the signal detected is proportional to the derivative of the measured MARY curve (see Figure 3.1). However, in order to achieve the high signal to noise ratio required to measure small field effects, a much higher modulation amplitude is often necessary, which can in turn distort the signal.

3.2 Theory of the field modulation spectroscopy

In order to understand the effect of modulation on the signal, we need a model of the detector. The lock-in amplifier (LIA) measures the amplitude of an oscillation component of the input voltage I, which is both frequency and phase coherent with the reference oscillation R. In this case R is the control voltage for the modulation coils power supply. This is achieved by direct multiplication of I and R in the time domain (mixing). This corresponds to convolution in the frequency domain and is also referred to as demodulation. The zero frequency (DC) component of the mixer output is the root-mean-square amplitude of I. It can be recovered with the use of a low-pass (LP) filter to give the output voltage S.[86] A characteristic setting of a LIA is its time constant τ, effectively its averaging time. The quantity $1/\tau$ is the width of the corresponding LP filter applied to the demodulated signal. The action of a simple resistor-capacitor (RC) LP filter can be represented by an integral:[87]

$$y(t) = \frac{1}{\tau}\int_{-\infty}^{0} e^{t/\tau}\, x(t)\, \mathrm{d}t \tag{3.1}$$

Where $y(t)$ is the filtered version of input signal $x(t)$ with LP filter of width $1/\tau$. In order to account for a possible phase difference between I and R, most LIAs implement a quadrature detection scheme whereby the input signal is split into two identical paths. One path mixes I with R, and the other with R delayed by quarter of an oscillation of f_m. The in-phase signal can be reconstructed from the resulting measurement (see Section 3.4). R can therefore be represented as a complex exponential where the real and imaginary

components correspond to the quadrature channels of the LIA:

$$R(t) = \cos(h\omega t) - i\sin(h\omega t) = \exp\left(i\frac{h\eta}{\tau}t\right) \tag{3.2}$$

It will prove convenient to work with the 'wavenumber' η, a proxy for a number of radians of oscillation of R in a time constant τ defined as $\tau\omega = \eta$. We have also introduced an integer harmonic number h which, for the time being, should be considered to be equal to 1 and will be useful later. The output of a LIA can be modelled by the following integral:

$$S_h(b_o) = \int_{-\infty}^{0} D_h(t)\, I(b_o, t)\, \mathrm{d}t \tag{3.3}$$

where D_h is defined as:

$$D_h(t) = \frac{\sqrt{2}}{\tau}\exp\left(\frac{1+ih\eta}{\tau}t\right) \tag{3.4}$$

Both of the quadrature components of the reference waveform, the normalisation constant, and the exponential detection window of the LP filter have been combined in a single exponential term – the detection kernel D_h. The factor of $\sqrt{2}$ accounts for the fact that the output of the LIA is proportional to the root-mean-square amplitude of the input signal.[84]

The case of $\eta \gg 1$ and $h > 0$ corresponds to the period of R being much longer than the LP filter averaging time τ. Then D_h by itself integrates to 0:

$$\int_{-\infty}^{0} D_h\, \mathrm{d}t = \begin{cases} \sqrt{2} & h = 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

$h = 0$ can be thought of as the special case of direct signal detection with no modulation (DC). D is represented pictorially in Figure 3.3a.

Single tone input

It will prove useful to understand how the LIA output, defined in Equation 3.3, responds to a single tone, 1 V root-mean-square (rms), input:

$$I(t) = \sqrt{2}\cos(\omega_I t + \theta) = \sqrt{2}\cos(\eta_I t/\tau + \theta) \tag{3.6}$$

where ω_I and η_I are the (angular) frequency, and wavenumber of the input tone. For $\eta \gg 1$, Equation 3.3 integrates to:

$$S = e^{-i\theta}\,\frac{1 - i(\eta - \eta_I)}{1 + (\eta - \eta_I)^2} \tag{3.7}$$

which is a Lorentzian lineshape with height 1, centered at η_I. The phase (θ) of the input rotates between the real and imaginary components of the Lorentzian. The real part corresponds to the the absorptive and the imaginary to the dispersive components of the Lorentzian.[88] The above equation can be simplified if we take the square modulus to obtain the power, as opposed to the magnitude, of the signal. The phase dependence of the quadrature channels vanishes, and Equation 3.7 simplifies to:

$$|S|^2 = \frac{1}{1 + (\eta - \eta_I)^2} \tag{3.8}$$

which is the response of the LIA to a unit root-mean-square (rms) amplitude, high frequency ($\eta \gg 1$) tone with wavenumber η_I as a function of the wavenumber of the reference waveform η. This is a notch with

the magnitude of the rms of the input centered at η_I. The width of $|S|^2$ at half-magnitude (–3 dB) is one wavenumber. This corresponds to the width of a corresponding RC filter with width $1/\tau$.

Transfer function

Having described the action of the LIA, we shall now consider the ModMARY experiment. The total magnetic field 'seen' by the sample is:

$$b(t) = b_o + b_m \cos(\omega t + \theta) \tag{3.9}$$

The oscillating input of the LIA in the ModMARY experiment arises due to the MARY curve acting akin to a transfer function M from field to fluorescence intensity (Figure 3.1). In the future data analysis, M will modelled by a (scaled and shifted) Lorentzian, or double Lorentzian lineshape (see Equation 3.39). The derivation presented in this section will only make the most basic assumptions about M – that it is a smooth, continuous function, well-defined over the entire domain of interest. Furthermore, the shape of M will be assumed to be time invariant. This assumption could be relaxed in more detailed analysis.

If M is the exact shape of the MARY curve, then the signal seen by the LIA is:

$$I(b) = I_0 + M(b) = I_0 + M(b_o + b_m \cos(\omega t + \theta)) \tag{3.10}$$

where I_0 is the baseline fluorescence component which has not been magnetically altered. The MARY curve M can be approximated locally by using a Taylor expansion of $M(b)$ about point b_o. This can be done for sufficiently small b_m, such that $b \approx b_o$ (see Equation 3.9). Substituting this Taylor expansion of M, up to the 1st order term, into Equation 3.10, gives:

$$\begin{aligned} I(b_o, t) &= I_0 + M(b_o) + M'(b_o)\left(b - b_o\right) + \mathcal{O}\left(M^{n\prime}(b_o)\right)_{n>1} \\ &\approx \left(I_0 + M(b_o)\right) + M'(b_o)\, b_m \cos(\omega t + \theta) \end{aligned} \tag{3.11}$$

where $'$ indicates a derivative with respect to magnitude of the magnetic field applied b, and $\mathcal{O}$ is the order function signifying higher order terms. This signal consists of the DC component $I_0 + M(b_o)$ and the AC component which, in this case, is a single tone at ω. This is an approximation of the input into the LIA. The corresponding output can be found by applying the LP filter equation. Substituting Equation 3.11 into Equation 3.3, and setting $h = 1$ gives:

$$\begin{aligned} S(b_o) &= \left(I_0 + M(b_o)\right) \int_{-\infty}^{0} D_1 \, \mathrm{d}t + M'(b_o)\, b_m \int_{-\infty}^{0} D_1 \cos(\omega t + \theta) \, \mathrm{d}t \\ &= e^{-i\theta} \frac{\sqrt{2}}{2} b_m M'(b_o) \end{aligned} \tag{3.12}$$

The DC component of I has been filtered out, and the detected signal is therefore the magnitude of the input signal – the derivative of the MARY curve M' scaled by the modulation amplitude $\sqrt{2}b_m/2$ and phased between the detection channels by angle θ.

The higher the b_m, the higher the signal, but also the more we depart from the regime where we can truncate higher order terms of the expansion in Equation 3.10. This approximation fails even for small b_m at, and close to, turning points where M' is by definition small, but M'' is large. This can be seen by plotting the accuracy of the local Taylor approximation of a Lorentzian lineshape with truncations at progressively higher terms. The log of local error of the expansion up to the n^{th} term (LLE_n) has been plotted in Figure 3.2. It can be seen that when LLE_1 is highest, LLE_2 is lowest. The successive even terms of the Taylor expansion contribute most where the previous odd ones contribute least, and vice versa. We can mitigate the problem

of inaccurate representation of I by leaving one more term in the expansion, giving:

$$\begin{aligned} I(b_o,t) &= I_0 + M(b_o) + M'(b_o)\big(b - b_o\big) + \frac{1}{2}M''(b_o)\big(b-b_o\big)^2 + \mathcal{O}_{n>2}\big(M^n(b_o)\big) \\ &\approx \left(I_0 + M(b_o) + \frac{M''(b_o)}{4}b_m^2\right) + M'(b_o)\,b_m\cos(\omega t+\theta) + \frac{M''(b_o)}{4}b_m^2\cos\big(2(\omega t+\theta)\big) \end{aligned} \tag{3.13}$$

There are now two tones, one at ω with phase θ, and one at 2ω with phase 2θ. The main component of an n^{th} harmonic of ω is the n^{th} derivative of M. The magnitude of the n^{th} harmonic also scales as $\mathcal{O}(b_m^n)$, although this does not necessarily translate into signal magnitude. This is because the magnitude of higher derivatives of physically useful M's tend to vanish as n increases, since such M's tend to be bounded and limited to a certain region (e.g. electron paramagnetic resonance spectra, spectral extinction coefficients, emission profiles). Note also, that M'' contributes to the DC component but will, as previously, be discarded by LP filtering.

Substituting Equation 3.13 into Equation 3.3 gives:

$$\begin{aligned} S_h(b_o) = M'(b_o)\,b_m &\int_{-\infty}^{0} D_h\cos(\omega t+\theta)\,\mathrm{d}t \\ + \frac{M''(b_o)}{4}b_m^2 &\int_{-\infty}^{0} D_h\cos(2\omega t+2\theta)\,\mathrm{d}t \end{aligned} \tag{3.14}$$

We can now use the previously introduced harmonic number h to tune into either of the two tones, since setting $h=2$ and $h=1$ will set the first and second integrals to 0 respectively. The first and second harmonic signals are given by:

$$S_1(b_o) = e^{-i\theta}\frac{\sqrt{2}}{2}b_m M'(b_o) \tag{3.15a}$$

$$S_2(b_o) = e^{-i2\theta}\frac{\sqrt{2}}{8}b_m^2 M''(b_o) \tag{3.15b}$$

The second harmonic signal (or first overtone) is proportional to the second derivative. It can therefore be used to reconstruct the MARY curve in regions where M' is small.

In principle, the greater the value of b_m, the greater the non-linearity of M in the region swept by the modulation ($b_o - b_m$ to $b_o + b_m$). This means that higher order expansions are needed to approximate M correctly and therefore higher overtone signals are detectable from the experiment.

As all the harmonics scale with powers of b_m, they become larger and easier to detect as b_m increases. However, it becomes much more problematic to interpret the data for high b_m. Just as the 2$^{\text{nd}}$ order expansion had an effect on the 0th order term (DC), the 3$^{\text{rd}}$ order expansion will have an effect on the 1$^{\text{st}}$ order term S_1. In general, each n^{th} order term contributes to the harmonics $n, n-2, n-4, \ldots$ etc.[89] This will be seen as broadening of the corresponding signals as b_m sweeps over larger and larger regions (see Figure 3.5).

Periodic split

To proceed with the derivation, we need to split the integral from Equation 3.3 into a sum of integrals over the units of a full period of ω, the fundamental reference frequency (see Figure 3.3a). Furthermore, we change the variable of integration from t to $\mathfrak{t} = \omega t/2\pi$, effectively time in units of one full period of ω. The split integral can be written as:

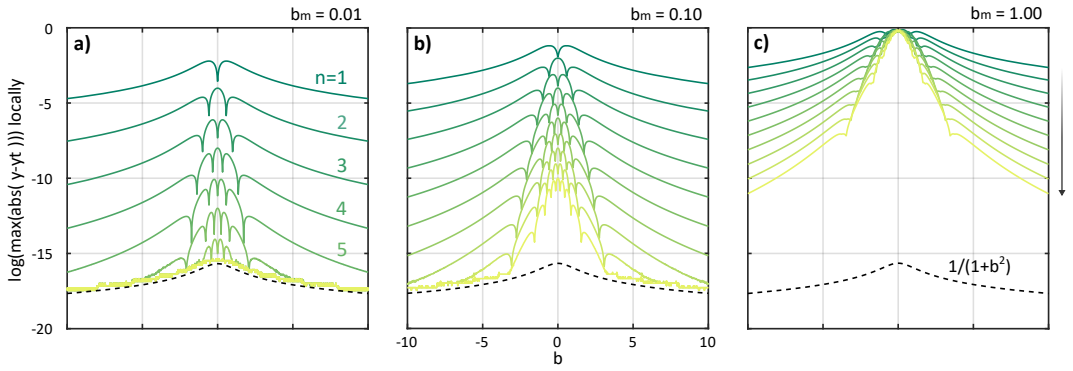


Figure 3.2: Log local error (LLE) in successive Taylor expansions of a Lorentzian lineshape $L(b)=\left(1-b^2\right)^{-1}$ (dashed black line) modelling a simple MARY curve. LLE_n is defined as $\log(\max(|L-L_n|))$ where L_n is the Taylor expansion of L *up to and including* the n^{th} derivative term in the region $b\pm0.1$. LLE_n can be treated as inversely proportional to the information about L, carried by the n^{th} term (and, by extension, the n^{th} harmonic) of the expansion. The region where $n=1$ has the largest error is best approximated by $n=2$. This shows how the second harmonic detection helps with reconstruction of the high curvature (i.e. high 2nd derivative) regions of M. The used for the LLE calculation can be found in Listing A.23.

$$\begin{aligned} S_h &= \int_{-\infty}^{0} D_h I \,\mathrm{d}t = \sum_{j=0}^{-\infty} \int_{\frac{(j-1)\pi}{\omega}}^{\frac{j\pi}{\omega}} D_h I \,\mathrm{d}t \\ &= \frac{2\sqrt{2}\pi}{\eta} \sum_{j=0}^{-\infty} \int_{j-1}^{j} \exp\left(\frac{2\pi}{\eta}\mathrm{t}\right) \exp\left(2\pi i h \mathrm{t}\right) I \,\mathrm{dt} \end{aligned} \tag{3.16}$$

where the explicit dependence of S_h on b_o has been dropped to simplify notation.

For each individual period, the exponential term $\exp\left(\frac{2\pi}{\eta}\mathrm{t}\right)$ can be expressed as:

$$\exp\left(\frac{2\pi}{\eta}\mathrm{t}\right) \quad \overset{j^{\text{th}}\text{ period}}{\rightarrow} \quad \exp\left(\frac{2\pi}{\eta}\left(j-{}^1\!/_2\right)\right) \times \exp\left(\frac{2\pi}{\eta}(\mathrm{t}+{}^1\!/_2)\right) \tag{3.17}$$

The first term is the height of $\exp\left(\frac{2\pi}{\eta}\mathrm{t}\right)$ half way through j^{th} period:

$$J(j,\eta) = \exp\left(\frac{2\pi}{\eta}\left(j-{}^1\!/_2\right)\right) \tag{3.18}$$

and the second is the exponential envelope:

$$E(\mathrm{t},\eta) = \exp\left(\frac{2\pi}{\eta}(\mathrm{t}+{}^1\!/_2)\right) \overset{\eta\gg1}{=} 1 \tag{3.19}$$

J is independent of t, and so can be taken out of the integral as a constant. The entire integrand is now independent of j and so the entire integral, in turn, can be taken out of the summation. Inserting the definitions from Equations 3.18 and 3.19 into Equation 3.16, and rearranging as described, yields:

$$S_h = \frac{2\sqrt{2}\pi}{\eta} \sum_{j=0}^{-\infty} \left\{J(j,\eta)\right\} \int_{-1}^{0} E(\mathrm{t},\eta) \exp\left(2\pi i h \mathrm{t}\right) I \,\mathrm{dt} \tag{3.20}$$

The limits of the integral have been changed accordingly, as the period-wise shift back in time (j) is fully accounted for by J. This can be done as the integrand is periodic in t with period of 1, by definition. The

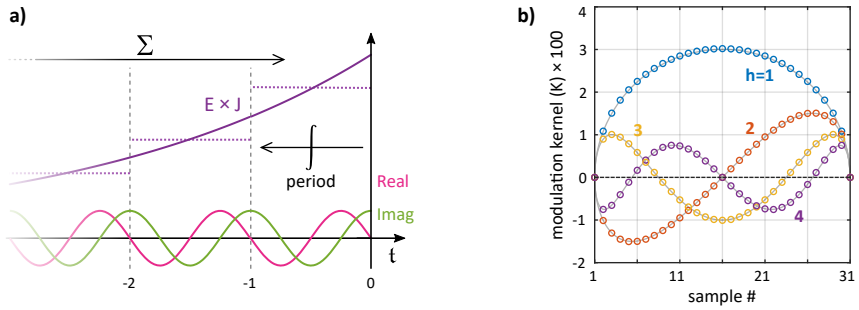


Figure 3.3: a) Pictorial representation of the detection kernel D and the split of the integral from Equation 3.16 into an infinite sum over full periods of the reference. Real (green) and imaginary (pink) quadrature components of the reference waveform are shown. The exponential integrator window is shown as the envelope function $E \times J$ (solid purple line) and the constant level (dashed purple line) **b)** First four modulation kernels K_h sampled with 31 points and scaled to conserve signal magnitude under convolution. K_2 is akin to the differentiation kernel for finite difference numerical differentiation (see Section 2.3). Correspondingly, K_3 is the second derivative. Note that these kernels operate on the derivative of M.

sum over J is bound and evaluates to a hyperbolic cosecant:

$$\sum_{j=0}^{-\infty} J(j,\eta) = \sum_{j=0}^{-\infty} \exp\left(\frac{2\pi}{\eta}\left(j - {}^1\!/_2\right)\right) = \frac{1}{2}\operatorname{csch}\left(\frac{\pi}{\eta}\right) \tag{3.21}$$

Equation 3.20 therefore becomes:

$$S_h = k \int_{-1}^{0} E \exp\left(2\pi i h \mathrm{t}\right) I \, \mathrm{dt} \tag{3.22}$$

where k is the scaling scaling constant defined as:

$$k = \frac{\sqrt{2}\pi}{\eta}\operatorname{csch}\left(\frac{\pi}{\eta}\right) \overset{\eta \gg 1}{=} \sqrt{2} \tag{3.23}$$

Change of domain

We have simplified an infinite integral to one over a single period of the fundamental frequency. We know how the applied magnetic field changes in time. We can use this knowledge to change the variable of integration from time to magnetic field – natural domain of the signal measured. To do this we to start by defining a modulation parameter $\mathfrak{m}$ as:

$$\mathfrak{m} = \cos(2\pi \mathrm{t} + \theta) \tag{3.24}$$

where $\mathfrak{m}$ is the oscillating part of the magnetic field scaled to unit amplitude, and $\omega t \rightarrow 2\pi \mathrm{t}$. The corresponding equations to change the variable of integration are:

$$\mathrm{t} = \frac{1}{2\pi}\cos^{-1}(\mathfrak{m}) - \frac{\theta}{2\pi} \tag{3.25a}$$

$$\mathrm{dt} = \mp \frac{1}{2\pi\sqrt{1-\mathfrak{m}^2}}\,\mathrm{d}\mathfrak{m} \tag{3.25b}$$

where the sign of the derivative changes between the left and right half-periods of ω. Changing the variable from t to $\mathfrak{m}$ in Equation 3.22 results in:

$$S_h(b_o) = \frac{k}{2\pi} e^{-ih\theta} \int_{\text{period}} \mp E \frac{\exp\left(ih\cos^{-1}(\mathfrak{m})\right)}{\sqrt{1-\mathfrak{m}^2}} M \, \mathrm{d}\mathfrak{m} \tag{3.26}$$

where the integration range has not been explicitly specified yet as it needs further careful attention. $I(t)$ has been replaced by $M(b) = M(b_o + b_m\mathfrak{m})$, the shape of the MARY curve in the range $[b_o - b_m, b_o + b_m]$. Since

only the integrand is complex, the integration can be carried out for the real and the imaginary components of Equation 3.26.

The real part of the integral from Equation 3.26 is:

$$\mathbb{I}_{\mathrm{Re}} = \int_{\text{period}} \mp E \frac{\cos\left(h\cos^{-1}(\mathfrak{m})\right)}{\sqrt{1-\mathfrak{m}^2}} M \, \mathrm{d}\mathfrak{m} \tag{3.27}$$

The limits of the integral are problematic since the inverse function of $\cos(x)$ is multivariate. The integral above must therefore be evaluated piecewise. Doing this for arbitrary η and h for both real and imaginary parts would give $2(2h+1)$ integrals. Proceeding with this would allow the derivation of a general solution, even far from the limit of $\eta \gg 1$. This analysis is complicated and unnecessary for this work. From now on we shall assume $\eta \gg 1$. In this limit we can use the previously established approximations $E \approx 1$ over the entire period ω, and $k \approx \sqrt{2}$. Applying these limits to Equation 3.27 yields:

$$\mathbb{I}_{\mathrm{Re}} = \int_{1}^{-1} \frac{-\mathcal{T}_h(\mathfrak{m})}{\sqrt{1-\mathfrak{m}^2}} M \, \mathrm{d}\mathfrak{m} + \int_{-1}^{1} \frac{\mathcal{T}_h(\mathfrak{m})}{\sqrt{1-\mathfrak{m}^2}} M \, \mathrm{d}\mathfrak{m} \tag{3.28}$$

where $\mathcal{T}_h(\mathfrak{m})$ has been recognised as a Chebyshev polynomial of the first kind, of order h, over $\mathfrak{m}$:

$$\mathcal{T}_h(\mathfrak{m}) = \cos\left(h\cos^{-1}(\mathfrak{m})\right) \tag{3.29}$$

For even h, the limits of the integral do not matter as the integrand is symmetric about $\mathfrak{m} = 0$, and hence the integrand is zero. For odd h the limits of the right integral can be swapped by changing the sign and so in each case the two integrals are equal. The real part of the integral from Equation 3.26 can therefore be written as:

$$\mathbb{I}_{\mathrm{Re}} = 2\int_{-1}^{1} \frac{\mathcal{T}_h(\mathfrak{m})}{\sqrt{1-\mathfrak{m}^2}} M \, \mathrm{d}\mathfrak{m} \tag{3.30}$$

A similar argument can be applied to the imaginary part of the integral. Here the corresponding term:

$$\pm \mathcal{U}_{h-1}(\mathfrak{m})\sqrt{1-\mathfrak{m}^2} = \sin\left(h\cos^{-1}(\mathfrak{m})\right) \tag{3.31}$$

is a Chebyshev polynomial of the second kind of order $h-1$, over $\mathfrak{m}$. The sign depends on the domain of the $\cos^{-1}(\mathfrak{m})$ term. It is positive for $\cos^{-1}(\mathfrak{m}) \in [-\pi, 0]$ and negative for $\cos^{-1}(\mathfrak{m}) \in [-2\pi, -\pi]$. This, conveniently, corresponds to the way we split the integral, and the additional sign change means the two halves cancel each other out. The imaginary part of the integral from Equation 3.26 is:

$$\mathbb{I}_{\mathrm{Im}} = \int_{1}^{-1} -\mathcal{U}_{h-1}(\mathfrak{m}) M \, \mathrm{d}\mathfrak{m} - \int_{-1}^{1} \mathcal{U}_{h-1}(\mathfrak{m}) M \, \mathrm{d}\mathfrak{m} = 0 \tag{3.32}$$

Combining both real (Equation 3.30) and imaginary (Equation 3.32) parts of the integral, Equation 3.26 becomes:

$$S_h(b_o) = \frac{\sqrt{2}}{\pi} e^{-ih\theta} \int_{-1}^{1} \frac{\mathcal{T}_h(\mathfrak{m})}{\sqrt{1-\mathfrak{m}^2}} M(b_o + b_m\mathfrak{m}) \, \mathrm{d}\mathfrak{m} \tag{3.33}$$

This integral in the field domain can be interpreted as a convolution of scaled M with a Chebyshev kernel. M is required to be scaled 'horizontally' in the field direction such that $b_m = 1$. This can always be done without loss of generality.

This expression for the LIA signal approaches a useful form. The only remaining problem now is that the $\mathcal{T}_h(\mathfrak{m})(1-\mathfrak{m}^2)^{-1/2}$ term has an infinite amplitude as $|m| \to 1$. This can be resolved by simplifying the integral from Equation 3.33 using integration by parts. Integrating the Chebyshev term and differentiating M (with respect to b) gives:

$$S_h(b_o) = \frac{\sqrt{2}}{\pi h} e^{-ih\theta} \left\{ M \int_{-1}^{1} \mathcal{U}_{h-1}(\mathfrak{m}) \sqrt{1-\mathfrak{m}^2}\, \mathrm{d}\mathfrak{m} + \int_{-1}^{1} \mathcal{U}_{h-1}(\mathfrak{m}) \sqrt{1-\mathfrak{m}^2} M' \, \mathrm{d}\mathfrak{m} \right\} \tag{3.34}$$

As the first integral is 0 for $h \geq 1$, the above equation can be simplified to:

$$S_h(b_o) = \int_{-1}^{1} K_h(\mathfrak{m}) M'(b_o + b_m \mathfrak{m})\, \mathrm{d}\mathfrak{m} \tag{3.35}$$

or, more compactly:

$$S_h(b_o) = K_h \otimes M' \tag{3.36}$$

where $\otimes$ represents the operation of convolution and K_h is the convolution kernel:

$$K_h = \begin{cases} \exp(-ih\theta)\, \frac{\sqrt{2}}{\pi h}\, \mathcal{U}_{h-1}(\mathfrak{m}) \sqrt{1-\mathfrak{m}^2} & \mathfrak{m} \in [-1, 1] \\ 0 & \text{otherwise} \end{cases} \tag{3.37}$$

It can be seen that the phase of the reference only rotates the signal between the real and imaginary planes. When considering a well phased signal ($\theta = 0$), the exponential term is equal to 1 and K_h and S_h are purely real.

M' is the field derivative of the transfer function which describes the magnetic field sensitivity of the sample of interest, and K_h is an *analytical* modulation kernel (Equation 3.37) which encapsulates both the modulation broadening and any further derivative action (for higher signal overtones). Representative 31-point convolution kernels for h from 1 to 3 have been plotted in Figure 3.3b. The amplitude of the modulation is used to scale the field axis of M' such that $b_m = 1$. Equivalently M' could be left unchanged and K_h scaled to go from $-b_m$ to $+b_m$. Note that, as long as $\eta \gg 1$, K_h does *not* depend on modulation frequency f_m nor on the LIA time constant τ.

This formulation of the ModMARY signal (and indeed, more generally, *any derivative spectroscopy signal*) is immensely useful. Not only does this give us an exact analytical shape of the obtained derivative signal (including the distortion due to modulation), but even when that analytical result is hard/impossible to calculate, numerical convolution is fast, even for very large number of data points. This allows fitting of *any* M to the derivative signal as long as the modulation amplitude is known. Moreover, this formulation allows us to interpret overtone signals easily. As discussed in Section 3.2, higher overtone signals often contain valuable information about regions of interest of M which, in the case of this work, is the LFE region of the MARY curve.

The utility of this method will be demonstrated in the following section with an experimental study and fitting of a well known magnetosensitive model system (Py/DCB) at different modulation amplitudes.

3.3 Technical details of the experimental apparatus

A schematic of the experiment can be seen in Figure 3.4. The sample is excited by a 365 nm LED (Thorlabs M365L2) which is collimated into a ~0.5 cm wide beam and directed onto the 1 cm quartz cuvette (Hellma Analytics, 130-10-40, 10 × 10 mm) through a reflective prism. The sample was flowed through the sample cell at 10 µLmin^{-1} and allowed to reach a steady state before the measurement. The effective radiant power

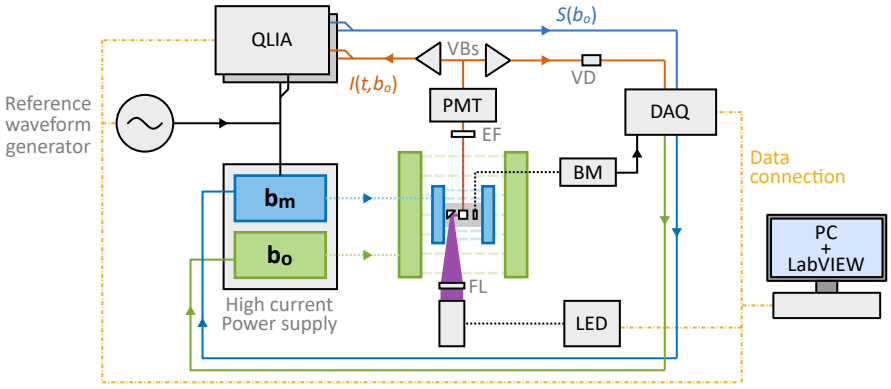


Figure 3.4: Schematic of the ModMARY experimental apparatus. The sample is excited at right angle to the detection by a LED passing though a focusing lens (FL) and a reflection prism. The samples' fluorescence passes through an excitation filter (EF) and is measured by a photomultiplier tube (PMT). The voltage reading of the PMT passes through a voltage buffer (VB) and is then the signal input I to the quadrature LIA setup (QLIA). The sample is contained within 2 parallel sets of magnetic field coils which produce the offset b_o and the modulated b_m magnetic field components. The QLIA and the b_m power supply share a common frequency reference from a waveform generator. The magnetic flux density next to the sample is measured with a Hall probe device (BM). Its output voltage, as well as the QLIA output and voltage divided (VD) PMT output are measured by a data acquisition device (DAQ). The DAQ also controls the modulation and offset field magnitudes. QLIA, reference waveform generator, LED, and DAQ are controlled through LabVIEW software.

at the sample has been estimated as ∼100 mW. The fluorescence is collected with a lightguide and directed to the photocathode of a photomultiplier tube (PMT, Hamamatsu) through a 455 nm LP filter (Comar optics, 455 GY 25). The PMT has been set in a μ-metal box and placed far from the magnetic field coils to shield it from the influence of the magnetic fields. Two sets of magnetic field coils are placed around the sample, one for the offset and one for the modulated components of the applied magnetic field. The fields applied by the coils are parallel to one another and each set has an independent power supply (Physical and Theoretical Chemistry Laboratory (PTCL), University of Oxford, Electronics workshop).

A Hall probe gaussmeter (PTCL electronics) is fixed in the sample cell holder to monitor the field applied over the course of the experiment. The probe has been calibrated against a commercial Gaussmeter (Lakeshore 425 with HST-4 probe), with the probe placed in the sample cell holder, which also accounts for any field distortion due to the cell housing.

The PMT outputs up to 15 V DC at maximum light exposure. The output was kept at ∼10 V by adjusting the gain before starting the experiment. The gain was then kept constant. The signal from the PMT is split into separate paths by two unity-gain voltage buffers (PTCL electronics). One path is AC coupled and demodulated by two analogue LIAs (Stanford Research Systems 510) corresponding to two quadrature channels. Both LIAs have been calibrated in the same way and, except for the $\pi/2$ phase shift are identical to one another. The other path goes through a calibrated voltage divider to bring the output of the PMT into the 10 V range.

The (scaled) PMT DC voltage, the analogue output of the LIAs and the gaussmeter signal are digitised by a data acquisition card (DAQ, National Instruments, PCIe-6323) at 2 kHz sample rate in 50 ms consecutive snapshots (data acquisition rate of 20 Hz). The PMT voltage was recorded through a voltage divider to ensure the DAQ card input was not overloaded. Each snapshot of the gaussmeter signal was processed additionally by converting the measured voltage to calibrated magnitude of the magnetic field with a 4^{th} order polynomial, and measuring the detected f_m and b_m with the FFT based single tone measurement search algorithm.[90] The DAQ card was also used to output an analogue ±10 V signal controlling the offset field power supply. The negative control voltage corresponding to the opposite current flow and hence magnetic field being applied in the opposite direction.

The experiment was controlled by a custom written program (National Instruments, LabVIEW, 2015b). The magnetic field was constantly adjusted using a fast Ziegler-Nicolas PID controller[91] to ensure any effects of magnet hysteresis were negligible. Moreover, the field was swept in an up-down-up fashion and between ± maximum field to ensure symmetrical signal with no artefacts. At each field point the experiment paused for at least 3 times the time constant of the LIAs before acquiring the 50 ms data sample. Each data point consists of the mean and standard deviation of each of the sample of the LIAs outputs. The PMT voltage, the gaussmeter reading, measured f_m and measured b_m were also recorded.

The offset magnetic field applied to the sample was in the range of ±40 G, with an accuracy of better than 0.1 G. The average field settling time following change of field set point in a sweep was 400 ms. Both the 1st and the 2nd harmonic (h_1 and h_2 respectively) signals have been measured for each dataset shown. The modulation amplitude was always kept constant throughout the course of the experiment. The achievable range of amplitudes was dependent on f_m due to the coils' impedance increasing at higher frequency. For all of the following experiments the modulation frequency was kept at 193 Hz, except for 20 G modulation depth where is was set to 71 Hz. Note that, by convention, ModMARY spectroscopy refers to the 'modulation depth' which is *twice* the modulation amplitude and corresponds directly to the region swept by the oscillating component of the magnetic field .

The reference frequency for the LIAs and the power supply of the modulation coils was provided by an external waveform generator (Rigol DG1022). The sample cell holder block was temperature controlled by a recirculating chiller (240 W, Grant Instruments, LT ecocool 100). The steady state temperature at the return of the chiller was just above the room temperature (23 °C). The equipment has been checked extensively to ensure that the magnetic field does not produce artefacts in the data (see supplementary information).

3.4 Methods of data analysis

The data acquired from each of the quadrature channels of the LIA S_x and S_y, has been phased by minimising the magnitude of the imaginary component Im with respect to the complex angle or rotation θ. The result is a phased signal S_p which preserves the sign of the original data:

$$\bar{\theta} \leftarrow \min_{\theta} \sum \mathrm{Im}\left[\left(S_x + iS_y\right) \times \exp(i\theta)\right]^2 \tag{3.38a}$$

$$S_p = \left(S_x + iS_y\right) \times \exp(i\bar{\theta}) \tag{3.38b}$$

In order to reconstruct the underlying MARY curve (fluorescence magnetosensitivity transfer function M) we can fit the h_1 and h_2 data to Equation 3.36. We can do this by creating a parametrised model for M, calculating the appropriate K from the experimental parameters, and performing a least-squares fit.

Consider a typical shape of the MARY curve (e.g. Figure 3.1a). We can use a sum of two Lorentzian lineshapes with opposite signs to model its shape, and therefore model M. This model M_L is:

$$M_L(x \,|\, \boldsymbol{p}) = 4x^2 \left(\frac{p_1 + p_2}{p_3^2 + 4x^2} - \frac{p_2}{p_4^2 + 4x^2} \right) \tag{3.39}$$

where $\boldsymbol{p}$ is the vector of fit parameters, the two amplitudes are given by p_1 and p_2, and the two widths of the high and low field components of M_L by are given by p_3 and p_4 respectively.

An analytical gradient for M_L is easy to calculate, but in the interest of generality, and to benchmark performance for an arbitrary M, a numerical gradient of densely sampled M_L was used instead. M_L' was convolved with the appropriately sampled and scaled K_h. The additional scaling was necessary to account for the discrete sampling of K_h (see Listing A.24). Special care was taken to sample M over a wide enough region for the convolution with K_h to be valid. This densely sampled result was then interpolated onto

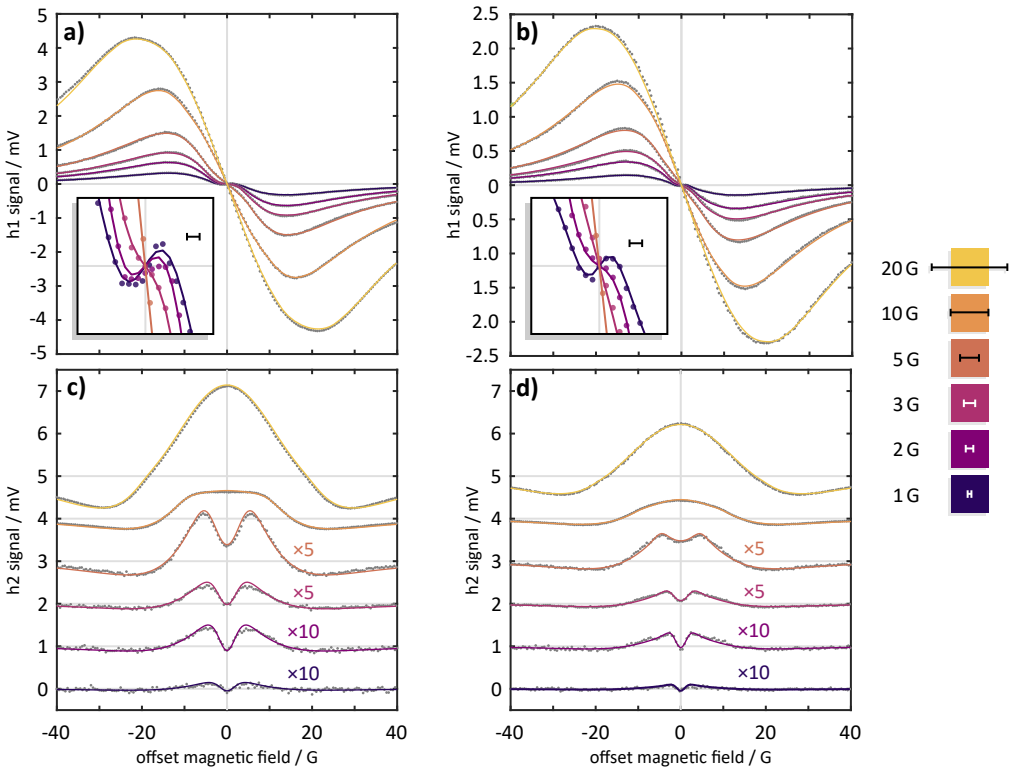


Figure 3.5: Simultaneously fitted 1st (top row) and 2nd (bottom row) harmonic data for PyH10 (left column) and PyD10 (right column). The scale bars shown on top of the legend on the right hand side correspond to the scale of the abscissa of all of the plots. The inserts in a) and b) show the detail of the LFE region of the 1st harmonic data. The scale bars in the inserts are 2 G wide. Modulation depths of 10 and 20 G have been omitted from the inserts as they would appear as a featureless vertical line. Note also the change of ordinate scale of b) with respect to a). The 2nd harmonic data in c) and d) has been shifted by progressive integer multiples of 1 mV to allow for easier comparison. Spectra at low modulation depths have been scaled up 10× for 1 and 2 G, and 5× for 3 and 5 G.

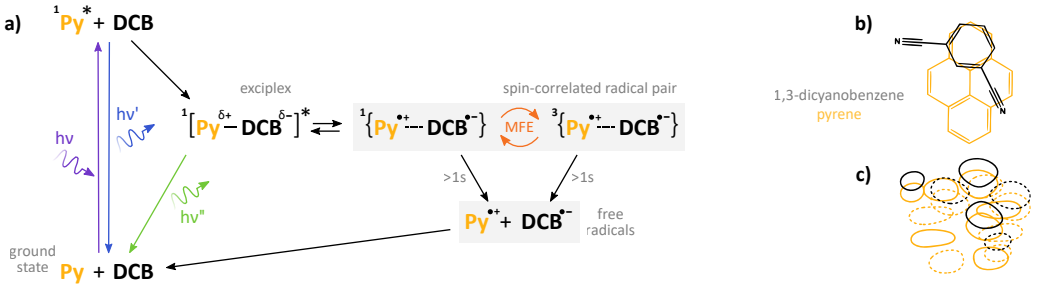


Figure 3.6: a) Photoscheme of the pyrene/1,3-dicyanobenzene system. The ground-state pyrene (Py) is excited a 365 nm. Photoexcited pyrene forms a fluorescent excited complex (exciplex) with 1,3-dicyanobenzene (DCB). Py-DCB exciplex fluorescence is red-shifted with respect to the ground-state Py fluorescence (see Figure B.2) and so it can be easily distinguished from that of S_1 pyrene. The Exciplex can undergo a full charge transfer to form a spin-correlated radical pair which can undergo coherent spin evolution, as discussed in Section 1.4. Compare this figure with Figure 1.12a. **b) & c)** The structure of the Py-DCB exciplex, as calculated by Ferguson.[71] c) shows the π-overlap between the species. Dashed and solid lines correspond to different signs of the wavefunctions.

the data axis and the mean square error between the fit and the data was evaluated and optimised as a function of $\boldsymbol{p}$. For fitting multiple datasets simultaneously, in order to account for varied signal magnitude between different modulation amplitudes and harmonics, the residuals for each dataset were scaled by its root-mean-square magnitude. The fitted vector of model (M_L) parameters $\bar{\boldsymbol{p}}$ was obtained by:

$$\bar{\boldsymbol{p}} \leftarrow \min_{\boldsymbol{p}} \left[\sum \left(S_p - K_h \otimes M'_L(\boldsymbol{p}) \right)^2 \div \sum S_p^2 \right] \tag{3.40}$$

The optimisation was performed with a Nelder-Mead search algorithm[58]. It is worth noting that an improper implementation of the convolution can lead to artefacts appearing in the spectrum due to zero-padding of the signal beyond its domain. In *the worst case scenario*, these artefacts can appear like features of interest, for example the LFE feature. In order to prevent this, M was always evaluated over the data range extended by $\pm b_m$ and trimmed to size for the purpose of fitting.

A method of bootstrap resampling of the residuals[92–94] and refitting was employed. This allows the evaluations of 'goodness of fit', as well as sampling parameters of interest of the fitted MARY curves. Briefly, the residuals of the best-fit were sampled in blocks, with replacement. The autocorrelation function of the residuals was used to estimate the block size in order to preserve the non-white nature of the new resampled residuals. These were then added onto the best-fit line(s) and refitted. This procedure was repeated 1000 times and the distribution of the fit values was analysed. The median, lower and upper standard deviation equivalence were extracted giving the fit parameters $\bar{\boldsymbol{p}}$ and their spread $\delta\bar{\boldsymbol{p}}$. The bootstrapped parameters of interest were the (x,y) position of the MARY curve amplitude at half saturation $B_{1/2}$, as well as the position of the maximum of the LFE feature.

The fitting was found to take roughly 25 ms for ~2 k points dataset on a typical desktop PC (Intel(R) Core(TM) i5-6600 CPU @ 3.3 GHz, 8 GB RAM). The subsequent 1 k points bootstrap took correspondingly ~30 s. The majority of this time was spent on the convolution, which averaged ~150 µs per call with $\sim 2 \times 10^6$ calls. This analysis was performed in MATLAB 2017a.[95]

3.5 Pyrene/1,3-dicyanobenzene system

The investigated sample is a donor/acceptor, mixture of pyrene (Py) and 1,3-dicyanobenzene (DCB) respectively.[96] This system has been extensively studied[71;73;97;98] and shows a large magnetic field effect, as well as a pronounced low field effect[72;85] making it a perfect model system for this analysis. Briefly, photoexcited S_1 pyrene forms an exciplex with DCB. The fluorescence of the exciplex is distinctly red shifted compared to that of the ground state.[96;99] The exciplex can also dissociate into a magnetosensitive radical pair which,

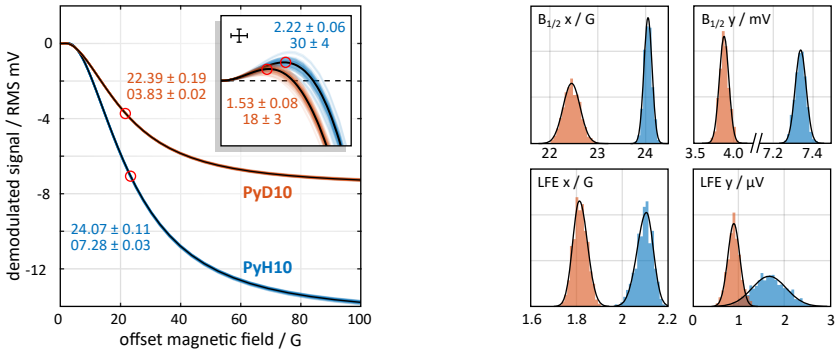


Figure 3.7: Reconstruction of the MARY curves of PyH10 (blue) and PyD10 (red), based on the fit of all of the datasets from Figure 3.5. The points of interest, $B_{1/2}$ and the maximum LFE, are marked, and the bootstrapped values with prediction bounds are given. The insert shows details of the LFE region. The scale cross in the insert is 0.5 G wide and 20 µV high.

due to the high viscosity of the solvent, has a high probability of spin-selective recombination. This means that a large percentage of the exciplex fluorescence is magnetosensitive, leading to large, easily observable MFEs. The photoscheme for this system can be seen in Figure 3.6.

Py (98 % Sigma Aldrich 185515) and DCB (98 % Sigma Aldrich 145858) were made up to concentrations of 400 µM and 40 mM respectively in pre-prepared 9:1 mixture of cyclohexanol (99 % Sigma Aldrich 105899) and acetonitrile (99.8 % Sigma Aldrich 271004). The concentration of the quencher (DCB) was kept in excess to suppress pyrene-pyrene excimer formation.[100] The solvent mixture was chosen due to its high viscosity, which promotes the formation of magnetosensitive exciplex species. Its high dielectric constant also stabilises charged species, including the exciplex and the radical pair.[70]

We have measured both natural abundance PyH10 and fully perdeuterated PyD10 pyrene samples. Deuteration decreases the molecular hyperfine coupling constant of the pyrenyl radical cation in the radical pair, and therefore should strongly affect the size and shape of the MARY curve.[75]

3.6 Results

We measured both the first (h_1) and second (h_2) harmonic data over a range of modulation depths for both PyH10/DCB and PyD10/DCB samples. For each of the two samples, phased signals for all of the harmonics and modulation depths were fitted simultaneously as described in Section 3.4. The results of that fit can be seen in Figure 3.5.

It was found that the simulation using M_L as a model for the MARY curve can reproduce both the shape and the magnitude of the signal very well, especially since the choice of modelling the LFE using a differential Lorentzian feature is somewhat arbitrary as a simple LFE model. Special care had to be taken for the signal magnitude to stay constant over the course of the experiment. Any change in the signal magnitude for long experiments has been attributed to 1) sample photodegradation and 2) heating of the modulation coils at very high modulation depths leading to increased impedance.

The sensitive detection of the LFE is one of the main goals of this work. We will therefore discuss a selection of the apparent features of the data that relate specifically to it. The LFE gives rise to a clearly visible feature around 2 G for the harmonic data (see insert of Figure 3.5a,b). As discussed, however, the information content of the h_1 (~1st derivative) signal in this region drops quickly with increasing modulation depth, and the LFE feature can only be seen clearly for $2 \times b_m$ in the range of 1–3 G. Conversely, the h_2 data shows a discernible LFE feature (dip/inflection in the middle of the signal) all the way up to $2 \times b_m = 10$ G for both datasets. h_2 data also contains information about the shoulders of the MFE at higher fields, but the signal

	PyH_{10}	PyD_{10}
$B_{1/2}$	24.07 ± 0.11 G	22.39 ± 0.19 G
	7.28 ± 0.03 mV	3.83 ± 0.02 mV
LFE	2.22 ± 0.06 G	1.53 ± 0.08 G
	30 ± 4 µV	18 ± 3 µV

Figure 3.8: Points of interest of the fits shown in Figure 3.7.

quickly vanishes at higher fields (compare e.g. signal magnitude for $b_o = 10$ G for low modulation depths). This means that h_2 alone is not sufficient for the reconstruction of M.

Good estimates of the MARY curve could not be obtained from the $2 \times b_m = 20$ G data. Higher overtones become easier to detect at higher modulation depths since the magnitude of the signal scales *approximately* with powers of $2 \times b_m$ (see Equation 3.13). Hence, a good fit at $2 \times b_m = 20$ G could, succeed if higher overtone data was measured and incorporated into the fit.[89] Interestingly, the LFE in the h_1 data for PyH_{10} gives rise to a small deflection in the gradient of the signal close to peak signal at –20 G (Figure 3.5). This is predicted exactly by the fit and indeed makes perfect sense as that is where the oscillating field begins to 'see' the LFE. This apparently visible LFE feature did not, however, improve the quality of the fit of that data alone as the information about the LFE has been distributed too thinly across the signal.

The analysis of the fitted data suggests that measurements of a magneto-sensitive chemical system at higher modulation depths – where $2 \times b_m$ is up to 5 times the expected LFE width, ought to succeed in reconstructing the underlying MARY curve M with the correct LFE feature by simultaneous fitting of the h_1 and h_2 signals (and higher overtones if signal magnitude is attainable), as the information about the higher order terms in Taylor expansion of the model of M (Figure 3.2) is retained by the higher order harmonics. Notably, this is a model-based technique and a correct model for M is assumed. As a simple model of a M for the purposes of MFE research, authors suggest the M_L model described above (although better, fast and more insightful models of a MARY curve are being actively developed[101]). This technique could however be used to model most stimulus-modulated derivative signals (those which conform with the assumptions made in the derivation). The correctness of M model must, in that case, be carefully considered.

A subset of reconstructed MARY curves M from this analysis, as well as the bootstrapped fits serving as prediction bounds can be seen in Figure 3.7.[94] It is clear that the PyD_{10} sample gives rise to a lower magnitude field effect with a narrower LFE feature (see insert of Figure 3.7). The bootstrapped (x,y) coordinates of $B_{1/2}$ and the maximum LFE point are tabulated in Table 3.8. The reported uncertainty is the standard deviation of the bootstrapped heuristics which were found to be approximately normally distributed. This was less of the case for the LFE maximum, at it sits close to the fixed point of M_L at (0,0). Compared to the natural isotopic abundance, the LFE feature of the perdeuterated pyrene shrinks and the magnitude of the overall field effect decreases. A strong effect of deuteration on the LFE is expected, since the hyperfine coupling constant of the deuterium atoms is approximately 25 % that of the protons.[71] The change in hyperfine MFE magnitude is also pronounced due to the resulting change in singlet probability. This can be contrasted to the work of Richert *et al.*[102] They showed no pronounced change in the magnitude of the MFE upon deuteration of pyrene in the pyrene/dimethylalanine system, which can be attributed to degenerate electron exchange.[72] The fact that we observe such change in our measurements suggests the electron self-exchange does not play a major role in Py/DCB, and the majority of the reaction proceeds through geminate exciplexes.

3.7 Conclusions

With minimal assumptions, we introduced a model of an arbitrary stimulus-modulated derivative signal and its overtones. We have shown its utility with specific application to modelling of Modulated MARY spectroscopy data of magnetic field effects. We were able to resolve a sub-Gauss shift in the low-field effect

feature upon deuteration, which is of great interest for the field of RP-based MFEs [6;103] and magnetoreception research. [3;14]

Multiple harmonics of the same sample response can be detected simultaneously with multiple quadrature detectors or a single multichannel one, especially if digital lock-in techniques are employed. We therefore propose that this type of harmonic reconstruction technique is widely applicable, and can be especially useful in Modulated MARY spectroscopy, where accurate reconstruction of the LFE feature is of utmost importance.

Chapter 4

Measurements of MFEs by confocal microscopy

This chapter will outline the work done towards the detection of MFEs by means of confocal laser-scanning microscopy. The technique will be briefly introduced and its potential advantages for the detection of MFEs will be discussed. Then, selected measurements of a magnetosensitive model system (FMN/HEWL) will be discussed, both in solution and crystal environments. Author's contributions to this work were the development of data analysis procedures, and a degree of scientific insight. The focus will, hence, be placed on these aspects (rather than, for example, technique development or crystal growth techniques). The samples were prepared and measured by Victoire Déjean (current DPhil candidate) and Jamie Gravell (MChem student 2018/19). The work described in this chapter has been a part of a publication by Déjean *et al.* (submitted for review), and the text of this chapter is partially based on it.[13]

The use of microscopy to detect MFEs has multiple potential advantages:

- Spatial *selectivity* allows for measurements of very small amounts of potentially precious and scarce bulk samples. This is of great value for measurements of e.g. cryptochrome solutions.
- High spatial *resolution* allows for imaging of microassemblies like crystals or cells. Both of these are interesting targets for MFE research as the spatial evolution of the MFEs can shed light on previously unexplored aspects of magnetosensitivity mechanisms.
- Low probe volume, combined with high light collection efficiency allows for use of lower excitation powers and hence decreased photodegradation.

The confocal laser-scanning microscopy (CLSM) further allows one to selectively image a focal plane, allowing a precise selection of single microscopic species of interest.[dx] Preliminary *widefield*,[2] as well as some *confocal* microscopy measurements of MFEs have been attempted before,[104] but never with the degree of success and scrutiny presented here.

4.1 Principle of CLSM

The principle of CLSM can be easily understood by recognising that the action of a lens is to map back and forth between directions of rays entering it, and positions of the focal plane. The core of a CLSM setup can be seen in Figure 4.1. The excitation beam enters the infinity-space of the imaging setup, couples into the objective by means of a dichroic mirror and gets mapped onto a single point on the focal plane.[dy] which then fluoresces in all directions.[dz] The proportion of the fluorescence captured by the objective enters the infinity space, passes *through* the dichroic mirror and gets focused though a pinhole into a monochromatic detector. Any fluorescence coming from above/below the focal plane is convergent/divergent in the infinity plane, does not get imaged onto the pinhole, and is therefore heavily attenuated at the detector. The excitation beam gets scanned though all N×N directions in the infinity-space, corresponding to the N×N positions in the image – individual pixels. This can be usually achieved by two scan mirrors mounted on galvanometers, inserted into the infinity space between the objective and the dichroic mirror. This has the added benefit of

[dx] The measurement of a single focal plane allows, too, for volumetric measurements of the fluorescence intensity (and therefore MFEs) by scanning the z-height of the focal plane.

[dy] Most modern objectives are infinity-focused. This means that all light coming form a point in the focal plane of the objective exits the objective as parallel rays (focused at infinity). The image can be constructed by refocusing the rays on the detector.

[dz] Delayed fluorescence, or phosphorescence can create artefacts in CLSM images.

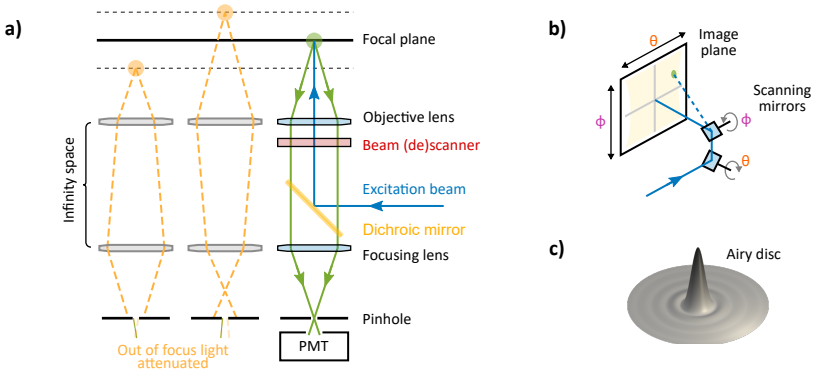


Figure 4.1: a) The principle of confocal microscopy. The rightmost column shows the fluorescence of the sample coming from the focal plane, and hence entering the infinite space as parallel rays. The two columns on the left show what happens with fluorescence coming from above/bellow the focal plane. Note that, for simplicity of the diagram, the objective lens is shown as a piece of 'thin optics'. **b)** Outline of the 2-mirror excitation beam scanning. The objective lens is not shown. **c)** Rendering of the Airy disk – diffraction limited intensity profile of the excitation beam.

descanning the fluorescence coming back from the sample.[ea] Each raster scan of the focal plane consists a single monochromatic image/frame.

Note that the wavelength multiplexing of the excitation and emission used in CLSM precludes the detection of emission close to the excitation wavelength. This is usually not a problem since the fluorescence of most chemical species is heavily red-shifted (bathochromic shift) due to fast vibrational relaxation of the excited state species, and therefore fluorescence occurs from a lower vibrational level than the one on which the initial exciton was created. Note too, that in practice the excitation beam spot size (and hence the region of the sample emitting the fluorescence) is finite and diffraction-limited. The exact spot shape is an *Airy disk* with the minimum-to-minimum diameter (approximately twice the full-width at half-maximum (FWHM)) of approx. 400 nm.[eb] An example Airy disk has been plotted in Figure 4.1c. The diameter of the Airy disk is is a measure of resolution in confocal microscopy, and it is therefore referred to as an Airy unit (AU).[ec]

The experiments performed here used a commercial CLSM system: Zeiss confocal microscope LSM 5 Exciter. The sample was continuously excited at 458 nm through a 63× oil-immersion objective. The fluorescence intensity was recorded by a photomultiplier tube (PMT). A 475 nm long-pass dichroic mirror was used to couple the excitation light into the objective. The typical framerates used for the experiments in this chapter were on the order of 1 frame per second. The detailed imaging parameters used can be found in the publication by Déjean *et al.*[13].

The magnetic field experiments used a pair of home-built ferrite-core solenoid coils mounted on the sample stage. A magnetic field of strength of ca. 16 mT was alternately switched on and off in synchrony with the framerate of the microscope using a waveform generator (Rigol DG1022). The field rise time was found to be on the order of 100 ms. No attempt has been made to synchronise the internal clocks of the microscope scan-head and the waveform generator. Instead, the period of the field switching waveform was simply matched with the integer multiple of the framerate. No dephasing of the field and the frames acquired was observed for the framerates and experiment lengths used.

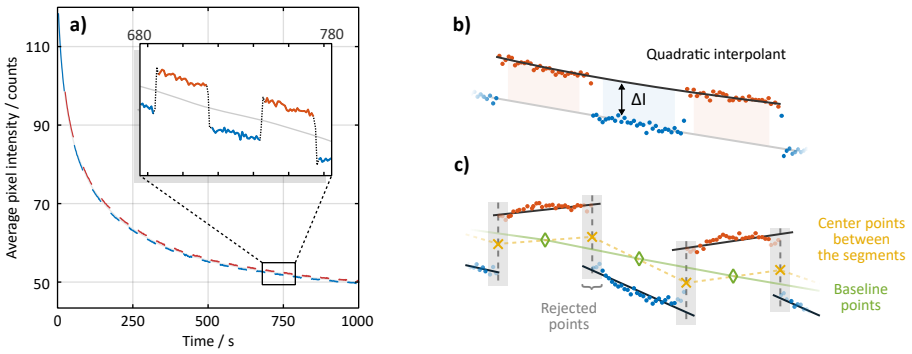


Figure 4.2: a) Representative average pixel intensity (`pixsum`) of a solution of 10 µM FMN and 1 mM HEWL solution. The intensity has been sliced into field-up (red) and field-down (blue) segments. A selected triplet of segments has been shown in detail in the insert. The centerline of the steps is also shown. **b)** Outline of the step interpolation methodology. The outer segments are fitted to a quadratic. The difference between the interpolant and the middle step ΔI is calculated for each point. **c)** Outline of the methodology for establishing the fluorescence baseline. The center points between the segments (yellow) and the baseline points (green) are indicated.

4.2 Image data analysis

Each frame is a raster scan of the excitation laser over the entire field of view (FOV) of the frame.[ed] The data is acquired as a stack of 8-bit frames (`uint8`). The basic analysis involves averaging together pixels in the region of interest (ROI) of each frame to obtain a single value. The time evolution of this average pixel intensity $\langle I \rangle$ is then analysed over the course of the experiment. The ROI is a spatial binary mask,[ee] constant across all the frames. The following is a simplified excerpt from the function `pixsum()` which is responsible for averaging together pixels in a specified ROI:[ef]

```
tstack = ⟨height × width × time matrix⟩;
roi = ⟨height × width boolean matrix⟩;
% Central tendency of the pixel values in the ROI
center_fun = @(x) mean(x);
% Standard error of the mean assuming every other pixel
% contributes a reasonable degree of freedom
spread_fun = @(x) std(x)/(sqrt(numel(x)/2));
for fi = 1:size(tstack,3)
    frame = double(tstack(:,:,ti)); % Current frame
    frame(~roi) = []; % Remove non-crystal
    frame = frame(:); % Reshape into 1D vector
    frame = frame(~isnan(frame)); % Remove NaNs
    clips = frame>=255 | frame<=0; frame = frame(~clips); % Remove clips
    pixsum_center(fi) = center_fun(frame);
    pixsum_spread(fi) = spread_fun(frame);
end
```

Listing 4.1: Average pixel intensity of an ROI of each frame in a stack.

[ea] The beam scanning speed is restricted by the impedance of the galvanometer coils, as well as the galvanometer having to overcome the momentum change of the mirror changing direction every scan line. The framerates achievable (~60 frames^{-1}) are, however, more than sufficient for high resolution spatiotemporal measurements.

[eb] Airy disk is sometimes referred to as the diffraction-limited point-spread function (PSF). PSF of a microscope is a function which represents all the inaccuracies and the physical nature of the imaging setup. The image from a (single acquisition) microscope is, effectively, the perfect image convolved with the PSF.

[ec] The abbreviation 'AU' should not to be confused with 'a.u.' – arbitrary units.

[ed] The raster scan is done by varying the two angles on the scanning mirror galvos (see Figure 4.1). The distance along the frame is a non-linear function of the angles. Further, the mirrors are often scanned in resonance with the galvanometer coil frequencies to achieve faster scan rates. This scans the angle if the mirror in a sine wave, increasing the non-linearity. Careful mapping of the exact galvanometer position and the corresponding pixel position is crucial to obtaining accurate images. This non-linearity can be mitigated by altering the shape of the scan mirrors.

[ee] In principle the ROI mask does not have to be binary.

[ef] The name `pixsum` is chosen for historic reasons, as well as for compatibility with previous versions of the code. It is indeed the *average* (not sum) of the pixels in the ROI that is calculated.

For each frame, the pixels from the region of interest are reshaped into a 1D vector. Any NaNs ('Not a Number' value defined in floating-point arithmetic[eg] which can be used as a placeholder for invalid data) as well as pixel values equal exactly to 0 or 255 are removed. The latter is done not to underestimate $\langle I \rangle$. The regions where the intensity is clipped are, therefore, completely removed from the frame an not considered for the average. In the full code of `pixsum()`, warnings are issued whenever any NaNs or clips are found. The *central tendency* and *spread* of the pixel values in the region is then calculated. The default center function is the arithmetic mean. The spread function which was found to express the typical noise of the data well is the standard error of the mean (`std(x)/sqrt(numel(x))`) adjusted such that, on average, every *other* pixel contributes a degree of freedom. This is only an estimate, but it is safe to make as it only *increases* the estimated spread (and therefore the uncertainty). An example plot of $\langle I \rangle$ against time can be seen in Figure 4.2.

Note the resolution of a single pixel is 8-bit – there are $2^8 = 256$ distinct values a single pixel can take. The resolution of the average value of many pixels is, however, considerably higher. Averaging together n pixels increases the resolution by $\log_2(n)$ such that average of two 8-bit pixels has resolution of 9-bit, 4 pixels – 10-bit etc. Further, assuming the light incident on the averaged pixels is uniformly distributed, the *dynamic range* (ratio of the dimmest to the brightest possible signal detectable above the noise floor) of the average pixel value increases too.[eh]

The data is then split into up/down (u/d) segments in synchrony with the field steps. Since the field switching period is a integer multiple of the framerate, this can be easily done by specifying the length of a single segment, and an additional offset which brings the slicing in phase with the steps. Note the similarity of this procedure to Fourier transform, and especially to the modulation/demodulation scheme discussed in section 2.1. The slicing period is the reference, and the steps in the data are the signal frequency. Finally, a number of data points are omitted at the edge of each segment, to allow for minor uncertainties in timing as the laser scans through the whole frame. This is usually only 1 point for segment length of ca. 25 points.

Spatiotemporally-resolved MFE To obtain a time-resolved MFE, each triplet of u-d-u and d-u-d segments is considered separately. The outer segments are fitted with a quadratic, which is then used to interpolate the values of the central segment. An example of this can be seen in Figure 4.2. The %MFE is calculated as the difference between the central data segment and the interpolant ΔI, divided by the field-off segment I_{off}. This can be done for all of the points in all of the segments obtaining a time-resolved %MFE.

Different interpolating functions were considered and tested, but a quadratic was found to yield the best results. Further, a fast, carefully optimised, variant of the Matlab's `polyfit` and `polyval` (as well as a weighted fit extension) were developed in order to facilitate the polynomial fitting of the MFE steps. Briefly, `polyfit_fast()` relies on constructing and inverting a Vandermode matrix – a matrix whose n^{th} column is the $(n-1)^{\text{st}}$ power of the `x` vector. Lots of unnecessary work can be avoided if the Vandermode matrix is cached between the fits (provided the `x` vector remains unchanged). Furthermore, the same Vandermode matrix can be used for `polyval_fast()`. The listings of these can be seen in A.11, A.12 and A.13. Note that the initial change in the fluorescence intensity is often too rapid, on the timescale of the field switching, for this procedure to work well. Some initial data points are, therefore, often discarded. Note, again the similarity to modulation/demodulation. The rapid initial change (usually a decay) in the fluorescence intensity can be considered a high frequency noise which, therefore, interferes with the carrier rejection by LP filtering (section 2.1).

[eg] NaNs, alongside of ±Infs are two special kinds of numbers defined in the Institute of Electrical and Electronics Engineers (IEEE) 754 standard for floating-point arithmetic used by all the modern computers. They correspond to undefined operations and over/under flow respectively. Other standards for computer arithmetic exist (e.g. *posit* arithmetic).

[eh] This assumes pixel noise of the neighbouring pixels is uncorrelated. Analysis involving correlation can become much more complicated very quickly as it depends on the exact position of the noise floor and the nature of the noise itself. Detector shot noise is, for example, always a positive, additive noise – there are no negative counts.

If a N×N image is split into multiple M×M bins (for example 128×128 image split into 32^2, 4×4 square bins), the above procedure can be applied for each of such bins to obtain *both* spatially and temporally resolved %MFE. All of the above discussion can be bought together in the following example:

```
tstack = ⟨512 × 512 × time matrix⟩ % Stack of frames
time = ⟨1 × time vector⟩; % Time axis
offset = 3; length = 30; dead_points = 1; % Splitting parameters
f = size(tstack,3); % Number of frames
%% Bin 4x4 pixels
binstack = zeros(128,128,f);
for fi = 1:f
    binstack(:,:,fi) = imresize(double(tstack(:,:,fi)),1/4);
end
%% Prepare u/d splitting and split time
n_segments = floor((f-offset)/(2*length))*2; % Number of u/d segments
trim_region = (1+offset):n_segments/2*(2*length)+offset;
time = reshape(time(trim_region),length,n_segments);
time = time((1+dead_points):(end-dead_points),:);
%% Calculate time-resolved MFE for each bin
MFE = zeros(128,128,numel(time(:,2:end-1)));
for ni = 1:128
    for mi = 1:128
        slice = squeeze(binstack(ni,mi,:)); % Select slice
        % Split into segments
        pixsum = reshape(slice(trim_region),length,n_segments);
        % Trim the edges of each segment
        pixsum = pixsum((1+dead_points):(end-dead_points),:);
        % Calculate mfe
        mfe = [];
        for j = 1:(n_segments-2) % For each triplet of segments
            y = pixsum(:,[j j+2]); x = time(:,[j j+2]);
            P = polyfit_fast(x(:),y(:),2); % Fit a quadratic to outer
                segments
            m = pixsum(:,j+1)./polyval_fast(P,time(:,j+1))' -- 1;
            mfe = [mfe; abs(m)];
        end
        MFE(ni,mi,:) = mfe;
    end
end
MFE_time = time(:,2:(end-1)); MFE_time = MFE_time(:); % MFE time axis
```

Listing 4.2: Example code for obtaining spatiotemporally-resolved MFE.

The spatial binning is done with the use of Matlab's `imresize()` (from the image processing toolbox) for simplicity. The actual analysis code performs the binning explicitly in a similar manner to the way `pixsum` is calculated in 4.1. The splitting of the `pixsum` is done by trimming it to a double-integer multiple of the segment length `l` (to ensure equal number of up and down segments), and reshaping it into a `l×n_segments` matrix. `d` corresponds to the number of edge points to trim, and `o` to the initial offset of the data. %MFE calculation has been done as described above, with use of `polyfit_fast()` and `polyval_fast()`. The speed of the polynomial fitting employed is important due to the volume of data required to be processed. `MFE` is also a stack of frames, but of reduced size (due to binning) and length (due to slicing). Note that this is only a demonstration code – it does indeed work, but does not account for multiple possible pitfalls (like, for example, different size of the image) and should therefore not be used verbatim.

MFE step dynamics The procedure outlined above is excellent for determination of the overall spatiotemporal behaviour of the MFE is in question. It does not, however, allow one to explore the shape of the individual MFE steps. A different procedure altogether is, therefore, required to explore the dynamics of the %MFE within a single such step. To do this, the baseline of the fluorescence has to be determined. This is done in two steps: 1) the center points between all the pairs of segments are found, 2) the centerpoints

between these are found and used as the baseline fluorescence. This procedure may appear overly complicated, but the need for it becomes apparent when one considers the example shown in Figure 4.2c. The center points between the segments (shown in yellow) do not reflect the true shape of the baseline as they are affected by the shape of the individual steps. These influences get cancelled out when one considers the center points between these (akin to a second difference, shown in green). The *centerline function* (`cfit`) is a spline interpolant though these.[ei]

The following is an example code establishing the baseline of the average pixel intensity:

```
tstack = ⟨height × width × time matrix⟩;
time = ⟨1 × time vector⟩; % Time axis
[h,w,f] = size(tstack);
pixsum = squeeze(sum(sum(double(tstack),1),2))./(h*w); % Entire frame
o = 10; l = 20; d = 1; % Splitting parameters
n_segments = floor((f-o)/(2*l))*2; % Number of segments
trim_region = (1+o):n_segments/2*(2*l)+o;
% Split and trim edges of time and pixsum
pixsum = reshape(pixsum(trim_region),l,n_segments);
time = reshape(time(trim_region),l,n_segments);
pixsum = pixsum((1+d):(end-d),:); time = time((1+d):(end-d),:);
% Find center points
for j = 1:n_segments-1
    P1 = polyfit(time(:,j),pixsum(:,j),1); % Fit j'th segment
    P2 = polyfit(time(:,j+1),pixsum(:,j+1),1); % Fit j+1'st segment
    x = (time(end,j) + time(1,j+1))/2; % Intermediate time point
    y = (polyval(P1,x) + polyval(P2,x))/2; % Average of poly. fits
    cp(j) = x + 1i*y;
end
cp = diff(cp)/2 + cp(1:end-1); % Find baseline points
cfit = @(x) interp1(real(cp),imag(cp),x,'spline'); % Interpolating fun.
```

Listing 4.3: Determination of the fluorescence baseline.

Since the splitting of the pixsum into segments is an inherent part of establishing the baseline, the splitting procedure has been shown here explicitly, in a manner very similar to that shown in Listing 4.2. Each pair of segments is fitted with a polynomial which in turn is evaluated at the middle time point between them. The points are stored as numbers on complex plane for the ease of evaluation of the second difference.

The magnetic-field induced change in intensity can be found by subtracting the centerline (`cfit`) from the average pixel intensity (`pixsum`). The %MFE relative to the centerline can then be found by dividing the result by the centerline intensity. Note that the same considerations as those discussed in section 2.4 have to be taken into consideration.

Slicing and shape correction As part of data analysis, it sometimes becomes useful to take a 1D slice though a 2D frame. This can be done by interpolating the data onto N data points between the line endpoints. N can be chosen arbitrarily, but in general it should be chosen such that the spacing between the interpolated points is as close as possible to the spacing between the centers of the individual pixels. The slicing of an entire tstack can be done as follows:

```
tstack = ⟨height × width × time matrix⟩;
[h,w,f] = size(tstack);
p =⟨1 × 2 complex vector⟩; % Points defining the slice line
l = linspace(p(1),p(2),ceil(norm(p(2)-p(1)))); % Line from p1 to p2
for fi = 1:f
    frame = double(tstack(:,:,fi));
    slices(:,fi) = interp2(1:w,1:h,frame,real(l)',imag(l)','linear',NaN);
```

[ei] Note that a somewhat different, yet broadly equivalent, procedure was used in the publication by Déjean *et al.*.[13] There, no second difference between the points was taken. The centerline function was a Savitsky-Golay *smoothing* interpolant through the center points between the segments.[65]

```
8  end
```

Listing 4.4: Obtaining slices through each frame of the stack of frames.

The points defining the slice have been expressed on complex plane due to the ease of interpolation.

The slices though a crystal can sometimes be 'distorted' due to the initial distribution of the fluorophore in the crystal matrix. The features one is interested in are often the *difference* of the n^{th} and the 1^{st} frame. The shape of the (smoothed) first frame can, therefore be subtracted from all of the data. However, the visual data analysis can then become quite difficult, as one has to always mentally account for looking at differential, rather than absolute fluorescence intensity data. A different approach, which has been adopted here, is to model the first frame as a smooth function, 'correct' its shape to the model and then apply that correction to all the other frames. This procedure has been applied to the slices shown in Figure 4.4. The data has been modelled as a smooth bump consisting of two sigmoid functions. The simplified code for that procedure is:

```
slices = ⟨slice length × time matrix⟩;
x = 1:size(slices,1); y = slices(:,1)'; % First slice
model = @(p) (x<mean(x)).*sigmoid_step(x,p(1),p(2),p(3),p(5)) + ...
            (x>=mean(x)).*sigmoid_step(x,p(2),p(1),p(4),p(5));
f = @(p) mean((y-model(p)).^2);
p0 = [min(y) max(y) 0.1*x(end) 0.9*x(end) 1];
pf = fminsearch(f,p0); % Fit the first slice with the model
cor_slices = bsxfun(@times,slices,(model(pf)./sgolayfilt(y,1,3))');
```

Listing 4.5: Correcting the shape of slices according to a model.

A similar procedure has been also applied to the entire frame of the data shown in Figure 4.9. There the first frame was modelled by a constant. The code for that correction is:

```
tstack = ⟨height × width × time matrix⟩;
frame1 = double(tstack(:,:,1)); % First frame
ref = mean(frame1(:))./imgaussfilt(frame1,1); % Reference
tstack_corr = bsxfun(@times,double(tstack),ref);
```

Listing 4.6: Correcting the shape of an entire tstack such that the first frame is flat.

Note that, since all of the frames are multiplied by exactly the same correction factor, this procedure does not affect the slicing and the resulting %MFE.

4.3 Spatiotemporal evolution of MFEs in crystals

Following the analysis procedure described above, this section will present and discuss the data obtained on hen-egg white lysozyme (HEWL) crystals co-crystallised with FMN. The samples have been prepared and measured by Jamie Gravell. The data analysis has been developed by the author. The scientific insight and theories were developed jointly.

Briefly, the crystals have been grown by a batch method – by mixing the precipitant (1.7 M NaCl) and the FMN/HEWL solution (60 mgL^{-1} HEWL (3.5 mM), 1 mM FMN in 100 µM pH 4.5 sodium acetate buffer) directly, in a 1:1 ratio in a sealed container. The concentrations and volumes were chosen such that the resulting solution exists in a metastable region where the crystallisation of the protein occurs spontaneously. This was left to incubate for 1-2 days to allow the crystal growth. The details of the imaging parameters for all the datasets were as follows. The pinhole was set to 10 AU which corresponded to about 5 µm optical slice thickness, while the crystal itself is estimated to be ca. 100 µm thick. The excitation laser was a 450 nm, 0.9 µW with a dwell time on each pixel of 10.24 µs. 475 nm LP filter was used to separate the excitation and emission light.

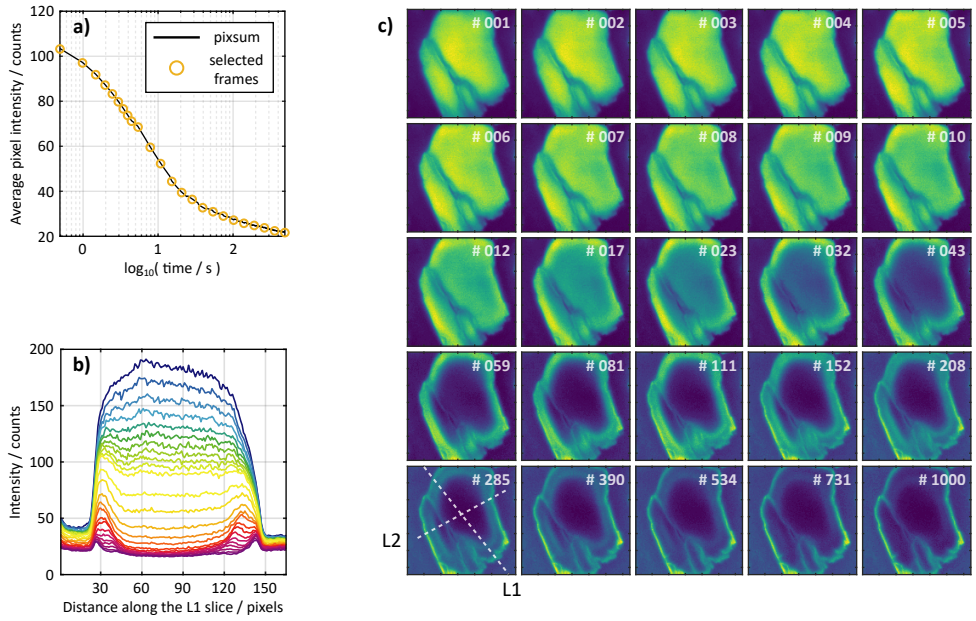


Figure 4.3: Fluorescence images of HEWL crystal co-crystallised with FMN. **a)** Average pixel intensity over the course of the experiment. Selected frames (approximately linearly spaced on the log-timescale) have been indicated and shown in full in c). **b)** Slices through the line indicated by L1, of the selected frames. Blue/green lines corresponds to early, and red/purple to late times. **c)** Selected frames of the dataset. The exact frame number has been given in the top right corner. The frame size is 202×202 µm – 512×512 pixels.

A 202×202 µm image of one such crystal can be seen in Figure 4.3. The average fluorescence intensity over the course of the experiment is shown in Figure 4.3a (note the logarithmic time axis). A selection of frames logarithmically-spaced in time have also been shown in Figure 4.3c. Fluorescence intensity decays substantially over the course of the experiment (∼8 min).

The dynamics underlying the exact shape of this decay are manyfold: photodegradation, reoxidation and, to a substantial degree, diffusion in both solution and in the crystal environment. An additional consideration could be the diffusion of the oxygen in and out of the FOV, since it contributes to both reoxidation and de-excitation pathways.[ej] More in-depth modelling of the interplay of these phenomena has not been the focus of this work. Instead, the decay of the fluorescence has been fitted by an empirical model consisting of two decaying exponential functions and a constant offset. The faster of the two exponential components has been attributed to the initial adjustment of the photocycle *within* the FOV to the onset of the illumination. The establishment of the steady-state of the diffusion of photoproduct out of, and fresh fluorophore into the FOV has been attributed to the slower component. The constant offset represents the overall steady-state fluorescence of the sample. Note that each pixel is, in fact, excited only periodically – every time it is visited by the raster scan.[ek] These are not conditions of constant illumination for process which are fast on the timescale of the raster scan (e.g. fluorescence), but ought to appear akin to constant for any process which are slower – diffusion and photodegradation.

There are a few aspects of the chemical system under consideration, important for understanding the fluorescence images shown in Figure 4.3c.

- The fluorescence quantum yield of single molecules (like FMN) is a function of the viscosity of their immediate environment.[105;106] This happens since the internal roto-vibrational motion of the

[ej] Ground state oxygen is a molecular triplet. It can react with a molecular triplet state of FMN in a spin-conserved reaction de-exciting it to the ground-state FMN and creating an excited singlet oxygen. This process can be thought akin to triplet-triplet annihilation in solid state exciton chemistry.

[ek] The exact illumination conditions are also a function of the Airy disk size. A raster scan resolution much higher than limited by the size of the excitation spot is usually possible. Each location of the excitation beam – each pixel, will therefore co-excite its neighbours.

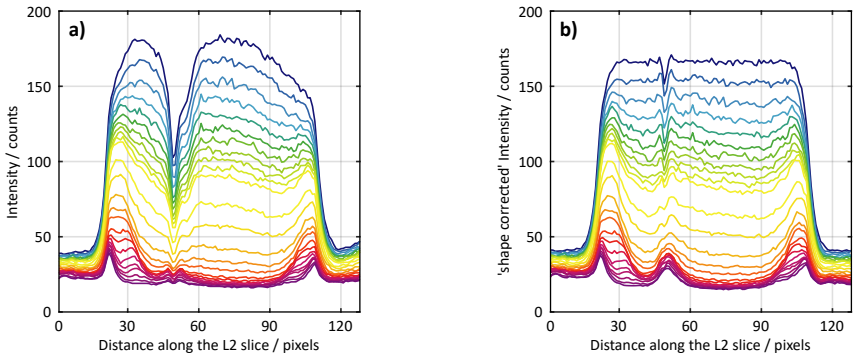


Figure 4.4: a) Slices along the line indicated by L2 in Figure 4.3c. The dip in the data corresponds to the brake in the crystal. **b)** the same slices, with their shape 'corrected' as described in Listing 4.5. The model for the shape of the first frame is a combination of two sigmoid steps. The brake in the crystal is almost invisible, and the underlying evolution of the wave-like feature can be clearly seen.

molecule can lead to a non-radiative decay of the exciton. A viscous environment restricts the internal degrees of freedom of the fluorophore therefore reducing the number of non-radiative exciton decay pathways.[el]

- HEWL crystal are known to be very porous and contain solvent channel which can be utilised for the fluorophore transport. Approximately 40% of the volume of the crystal has been reported to be taken up by the solvent/dopants.[107;108] This is a restricted environment which, nevertheless, allows for long-distance diffusion of the fluorophore though the crystal matrix. The (isotropic) diffusion coefficient of flavin in solution and in the lysozyme crystal are estimated as $5.75 \times 10^{-10}\,m^2s^{-1}$ and $2.4 \times 10^{-13}\,m^2s^{-1}$ – three orders of magnitude difference.[13;109;110]
- When considering FMN and HEWL in solution (with HEWL molecules suspended in solution, rather than crystallised in a matrix) the electrostatic interaction of the net-negative FMN molecules with the net-positive surface of HEWL plays an important role in the development of the MFE. The electrostatic interaction increases the encounter and recombination rate of the radicals therefore increasing the yield of the radical pair, as well as its downstream photoproducts. This interaction may also play a role within the crystal environment, although it is not clear whether this would be a dominant interaction.

There are a few features of the spatiotemporal evolution of the fluorescence shown in Figure 4.3c which are worth particularly close attention. The crystal can be seen to bleach preferentially in the center, leaving the edges bright. This can be explained by rapid replenishment of fluorophore from the solution, which then accumulates in the edge of the crystal. Furthermore, the fluorophores located at the edge of the crystal will be in a more restricted environment than those in the center.[em] This will further increase the fluorescence quantum yield and could contribute to this bright-edge effect. This effect has been seen in all other measurements, as well as many published confocal measurements of fluorescence in crystals (both HEWL and otherwise).[111]

A less apparent, but arguably much more exciting, feature of the fluorescence intensity is the faint intensity band which starts close to the edge of the crystal, but then moves inwards as a function of time. To illustrate it better, a series of slices though data in Figure 4.3c has been taken along the line indicated with L1, and shown in Figure 4.3b. To demonstrate that this feature emerges on all edges of the crystal, a slice through a line indicated by L2 has been shown in Figure 4.4. The origin of this band of intensity is presently unknown. Since it travels from the edges to the center of the crystal, it is likely to be related to diffusion processes inside of the crystal. The rate of travel of the wavefront of this feature broadly agrees with the linear diffusion

[el] Since the fluorescence lifetime is related to the fluorescence quantum yield, this is the basis of fluorescence lifetime imaging microscopy (FLIM).

[em] Consider, for example, a particle moving randomly through a rectangular grid, in small steps only towards its 4 nearest neighbours. If this happens with equal probability for each of the neighbours, the particle will spend, on average, more time around the edges of the grid since the edge cell has only 3 neighbours. The edge has, effectively (and literally) lower dimensionality than the bulk.

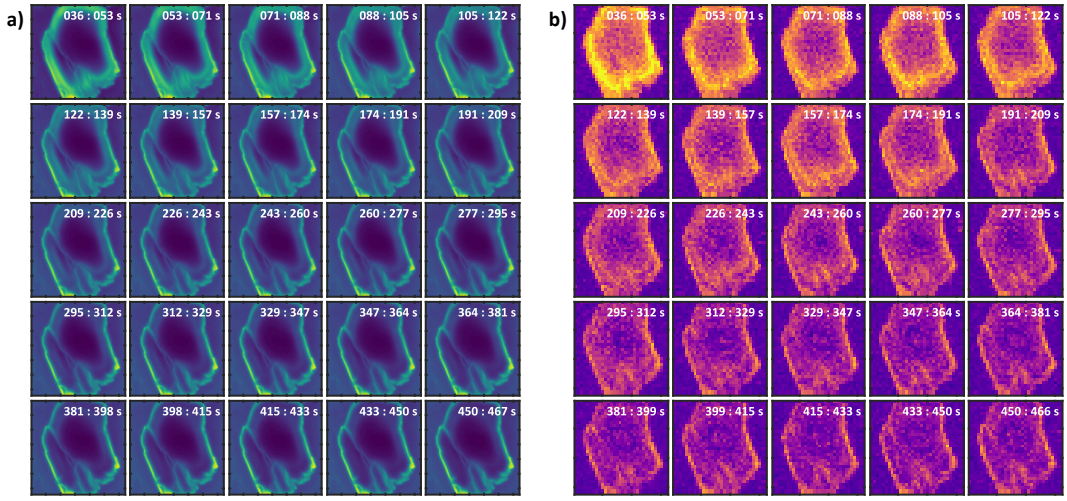


Figure 4.5: a) 15 s time-bin average of the intensity data shown in Figure 4.3c. Exact edges of the time-bins are given in each image. **b)** Spatially-resolved %MFE of the data in a). Spatial binning of 4×4 pixels and time-binning akin to a) was used.

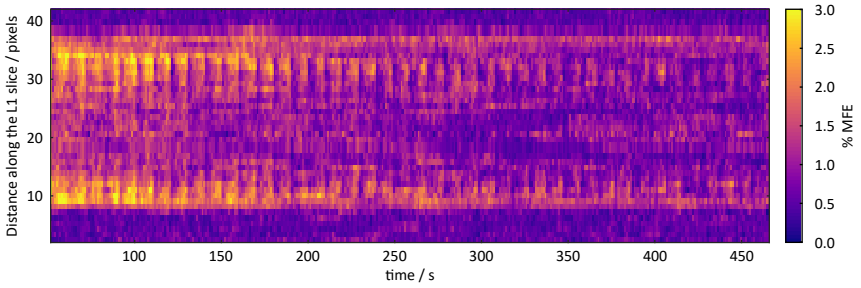


Figure 4.6: Time evolution of the L1 slice (as indicated in Figure 4.3c) of the %MFE data shown in Figure 4.5b. No time binning was used.

rate of FMN in HEWL crystals. An important observation is, however, that diffusion by itself is a purely dispersive phenomena – *diffusion does not create waves.* This wave must be therefore related to the interplay of the diffusion with some other effect.

It is possible that this interaction is the strong association of the fluorophore with the crystal. As FMN is diffusing into the crystal, it would travel much slower than in solution and therefore, the flux into the crystal was strong enough, it could build up. This, coupled with the continuous photodegradation could result in a wave-like feature. A different explanation could be centered around the photodegradation of FMN. The photodegradation product of FMN is lumiflavin (LF) which is not excited at 458 nm and therefore does not contribute to the fluorescence signal.[112] Since the diffusion in solution is much faster than within the crystal, there would be a net concentration gradient of LF *out* of the crystal, at the same time as FMN diffuses *in.* These two opposing fluxes would not normally be considered to interact strongly (if at all), but since they both happen in a very restricted environment, they would most likely interact in this case. This, again, could lead to a creation of such wave-like feature.[en] Note that, since this feature evolves on the timescale much slower than the framerate of the experiment, it has, most likely, little to do with the periodic nature of the illumination in CLSM.

Figure 4.5b shows a spatially resolved MFE, obtained in a manner described in the previous section (see Listing 4.2). The data has been shown as an average over 15 s time-bins to improve the signal to noise ratio

[en] The argument presented here is based on the large-scale fluid mechanics behaviour of two liquids with different densities and opposing flows (e.g. that of molten wax and mineral oil in lava lamps). A related phenomena is the Rayleigh–Taylor instability which mixes the opposing flows. Note, though, that macroscopic flow dynamics *does not necessarily translate well* into the molecular flow regime inside of the HEWL crystal, and so a direct comparison ought not to be made.

of the MFE images. The absolute fluorescence images have also been averaged over the same time bin and shown in Figure 4.5a, for comparison. Interestingly, the same wave-like band discussed above also appears in the spatially resolved MFE. Bright edges of the crystal can also be observed on the MFE images. This may not be as readily apparent as in the fluorescence intensity data, due to lower signal-to-noise and lower resolution of the MFE data. The MFE data has therefore been sliced through the line L1 akin to the way shown in Listing 4.4, and consecutive slices have been shown in Figure 4.6. The appearance of the bright edges, and the wave-like feature in the MFE moving towards the center of the crystal can now be seam much better. The correspondence between the fluorescence intensity and the %MFE images means that the magnitude of the MFE increases in the regions of increased fluorescence intensity and therefore, most likely increased local FMN concentration. This suggests that the presence of the fluorophore is the limiting factor in the development of the MFE in this system. This is a trend which is opposite to that observed by Evans in measurements of fluorescence of bulk solutions of much higher concentrations.[113] It is also possible that the observed features can be attributed, at least partially, to the local viscosity, rather than the FMN concentration gradients though the crystal. As discussed, local viscosity affects the fluorescence intensity, but it can also have an effect also the magnitude of the MFE. This happens either by changing the fluorescence quantum yield, and therefore the proportion of the photocycle which is allowed to progress though the magnetosensitive step, or by affecting the geminate recombination rates of the RP partners.

A careful analysis and further understanding of this newly-observed phenomena is therefore warranted, if not for understanding it for its own sake, then because it appears to have a strong connection with the spatiotemporal evolution of the MFE. Before proceeding, however, there is another feature of these data which is has not been discussed yet – the dynamics of the MFE step itself. This is the origin of the vertical stripes in Figure 4.6. Each step consists of a *prompt* MFE step, and then further, slow, increase in the MFE magnitude over the course of the field step. This effect has been referred to as the *enhanced* MFE (EMFE) by Kattnig *et al.*,[2] although, rather confusingly, it has also been observed to diminish the *asymptotic* (long-timescale) MFE following a large prompt MFE. The prompt component of the MFE is attributed by Kattnig *et al.* to the attainment of a new steady state at a rate limited by the bulk re-encounter of the free radicals. The the slow component – its rate and the asymptotic value has been 'shown to arise from asymmetrical rates of termination reaction of the two radical pair partners'.[13] In the case of Kattnig's work these termination reactions are the redox reactions of free radicals in solution. In this work these rates are most likely related to the fact that one of the RP partners (FMN) is diffusing freely though a matrix consisting of the other RP partner (HEWL). The EMFE is therefore predicted to occur only within the crystal. This can be seen in Figure 4.7 where the average, baseline-subtracted pixel intensity is plotted for two regions of interest: inside (ROI1) and outside (ROI2) of the crystal. Note that the baseline is used as the reference for the MFE. The %MFE in Figure 4.7, and also later in Figure 4.12 are therefore reported *with respect to the centerline*, not with respect to the zero field.

The solution around the crystal (Figure 4.7b) shows a small MFE step, as expected, but no EMFE effects. The center of the crystal (Figure 4.7l), on the other hand, shown very pronounced EMFE steps, which can be seen to drop in magnitude over the course of the experiment. The average MFE step for both of these traces have been plotted in Figure 4.8. To assess the timescale of the EMFE, the EMFE step has been fitted with a piecewise exponential step wave (see Listing A.21). The EMFE time constant was found to be 7.19 s.

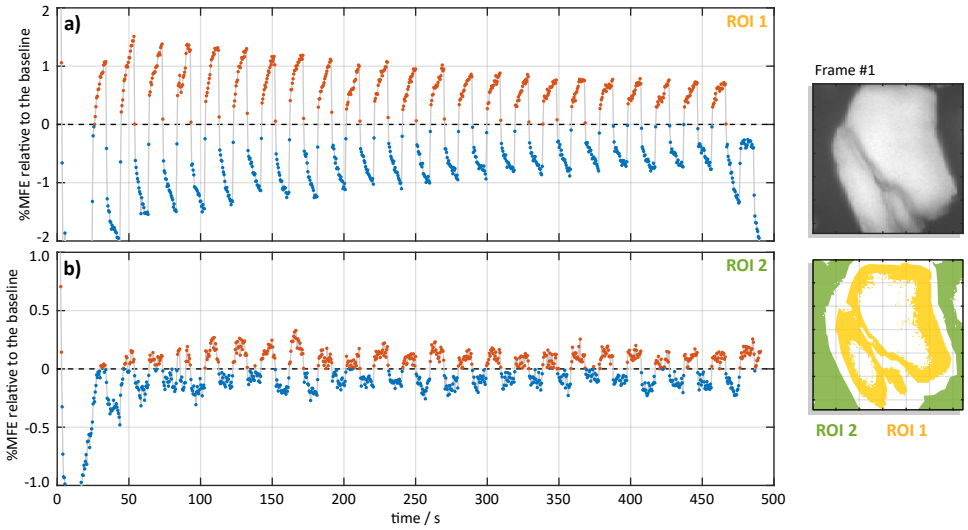


Figure 4.7: Time evolution of the pixel intensity change, relative to the centerline, in two ROIs of the crystal shown in Figure 4.3c. **a)** ROI just inside of the crystal. The fluorescence can be seen to exhibit clear EMFE steps. **b)** ROI is the solution surrounding the crystal. The MFE steps are much smaller, with no visible EMFE. Note that some of the points at the edges of the data are distorted due to invalid baseline fitting.

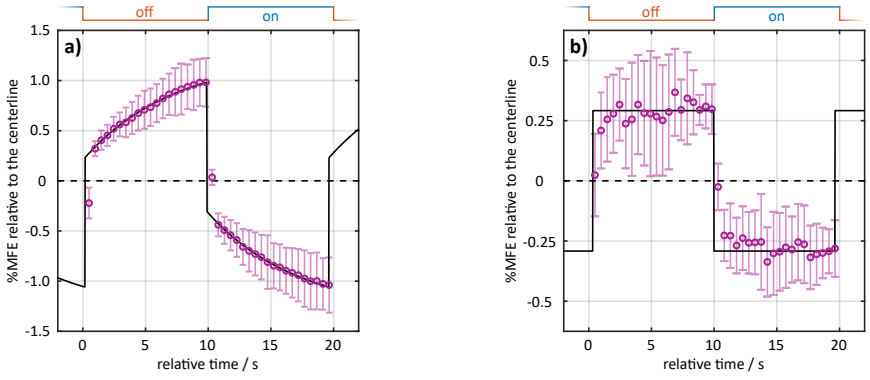


Figure 4.8: Average step of the data shown in Figure 4.7. The points at the edges of the data were discarded. **a)** Average step in ROI1 in Figure 4.7. **b)** Average step in ROI2 in Figure 4.7. The error bars are the standard deviation of the average of the steps. Steps in ROI1 are by no means of constant size, hence the corresponding error bars appear larger than expected.

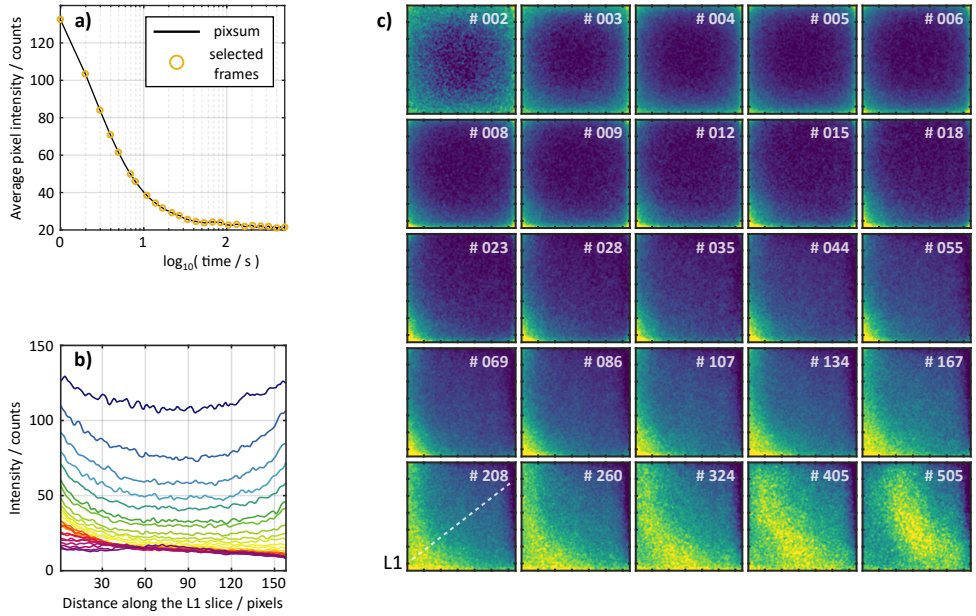


Figure 4.9: HEWL crystal co-crystallised with FMN. The FOV is placed just inside of the crystal, with the crystal edge just outside of the frames' bottom-right corner. This data has been shape-'corrected' as shown in Listing 4.6. **a)** Average pixel intensity over the course of the experiment. Selected frames (approximately linearly spaced on the log-timescale) have been indicated and shown in full in c). **b)** Slices through the line indicated by L1, of the selected frames. Blue/green lines corresponds to early, and red/purple to late times. **c)** Selected frames of the dataset. The exact frame number has been given in the top right corner. The first frame is used for shape correction, and hence not shown here. The frame size is 51×51 µm – 512×512 pixels.

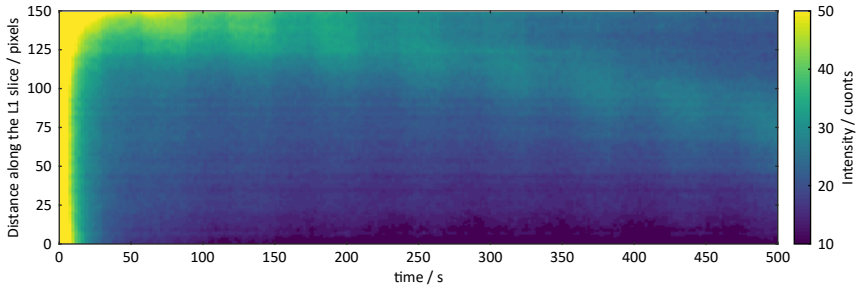


Figure 4.10: Slices through the fluorescence intensity, along the line indicated by L1 in Figure 4.9. The MFE steps, as well as the travel of the wave can be clearly seen.

4.4 MFE dynamics imaging

In order to explore more details of the fluorescence wave travelling though the crystal, and its relationship to MFEs, a different dataset was acquired and analysed. The data for this crystal is shown in Figure 4.9 in a manner akin to that from Figure 4.3. Here, the FOV is much smaller than in the data shown in Figure 4.3. The FOV is located just inside of the body of the crystal, such that the wave-like feature can be seen to move across the frame over the course of the experiment. Further, the field switching period has been set to 30 s such that the full EMFE evolution can be observed. Note that the data shown is shape-corrected as can be seen in Listing 4.6. This is done to better showcase the wave travelling through the crystal. All of the following analysis has also been performed with unaltered data to the same result.

A slice through the intensity and the spatially resolved MFE has been taken parallel to the direction of travel of the wave, and can be seen in Figures 4.10 and 4.11 respectively. The position slice has been indicated in Figure 4.9c with L1. The correspondence of the wave between the intensity and MFE slices can be clearly seen. Further, because of the increased resolution and zoom of this dataset, one can also observe the

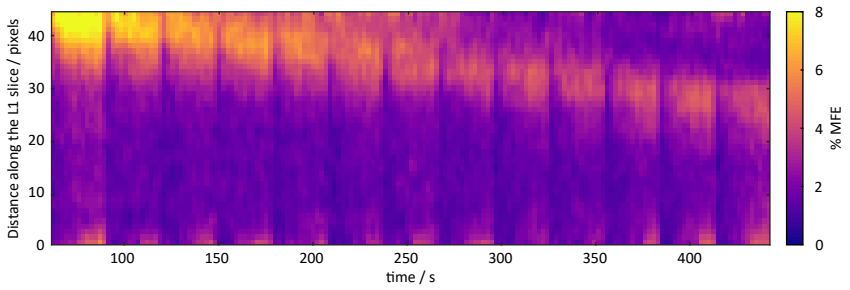


Figure 4.11: Slices through spatially-resolved %MFE of the data from Figure 4.9. The slices have been done along the line indicated by L1 in Figure 4.9. The MFE EMFE of the steps, as well as the correspondence of the %MFE and fluorescence intensity can be seen in comparison to Figure 4.10.

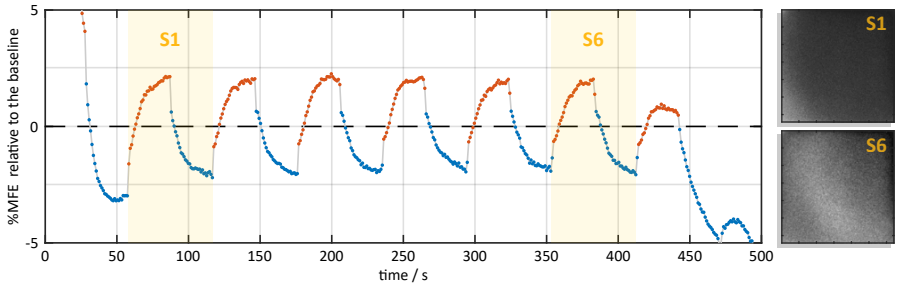


Figure 4.12: Time evolution of the average pixel intensity of the entire frame, relative to the centerline, of the data shown in Figure 4.9c. The average intensity over the first and the 6th step (S1 and S6 respectively) is shown. Similarly to the data shown in Figure 4.7, some of the points at the edges of the data are distorted due to invalid baseline fitting.

difference between the EMFE in the *path* and in the *wake* of the wave (see, for example, the ~400 s region of Figure 4.11).

The spatiotemporal dynamics of EMFE in this data can be explored by splitting the data into spatial bins, averaging and then analysing each individual MFE. The EMFE parameters will constitute a stack of images, one time point per segment, describing the evolution of the EMFE. The pixsum of an entire frame can be seen in Figure 4.12. There are 6 usable u/d segments and therefore the final EMFE image stack will have only 6 frames. This analysis can be performed relatively easily in a manner similar to that shown in Listing 4.2. The 512×512 pixels frame has been divided into 8×8 bins and each segment of each spatial bin has been fitted with a piecewise exponential step wave, like described before. Each fit has to be performed on a significantly lower signal-to-noise ratio single step coming from an average of 64 pixels in each spatial bin.[eo] Three parameters of the EMFE are plotted in Figure 4.13 – the asymptotic MFE, the prompt MFE and the fitted EMFE time constant. Additionally the ratio of the prompt to asymptotic MFE is also plotted. Note that, in the case of this work, 'prompt' referrers to as seen in the data. It is possible the actual fast step of the MFE, like the one observed by Kattnig *et al.*,[2] is faster than the one which be seen with the time resolution available here.

The asymptotic MFE can be thought of as simply the magnitude of the MFE. It does, indeed, appear to follow the same time evolution as the raw fluorescence intensity and the corresponding spatially-resolved MFE. The prompt, however, MFE appears constant both in space and time. This suggests the process behind the prompt MFE – initial establishment of the MFE governed by the re-encounter rate of the free radicals – is constant in the entire FOV. This is in agreement with the fact that the entire FOV is located in a single environment – the crystal. The ratiometric images of prompt-to-asymptotic MFE (3rd row in Figure 4.13) reveal that prompt MFE dominates in the wake of the wave, while being only a small contribution to the overall MFE in the regions which the wave has not passed through yet.

[eo] To aid the fitting, the period and the horizontal shift of the exponential step wave were fixed. A `patternsearch` solver was used for the fitting.

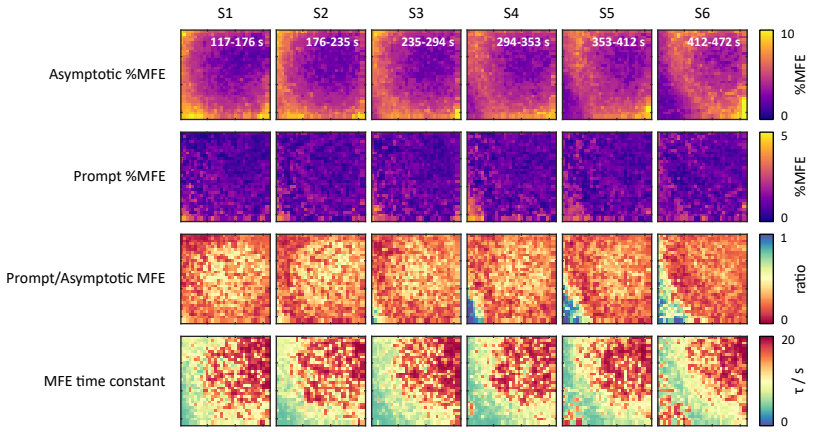


Figure 4.13: Spatially-resolved EMFE for each of the steps shown in Figure 4.12. The wave can be seen in both the asymptotic MFE, and the τ-images. The prompt MFE appears constant (ca. 1 %) across the entire frame.

A possible explanation for this observation could be a change in the internal structure of, maybe not necessarily the HEWL crystal itself, but the organisation of the solvent, salt and the FMN molecules in the crystal pores. This could be interpreted in terms of the theory of the origin of the wave where it is caused by the opposing fluxes of FMN and LF towards and away from the center of the crystal respectively. Initially, the structure of the two flows is chaotic and the overall motion of the FMN and LF is relatively slow in the entire crystal. Close to the edge the two flows begin to organise – certain solvent channels (or certain paths though the solvent channels) become used only for FMN and certain only for LF transport. This organisation moves towards the center of the crystal. The interface between the organised and unorganised solvent channels requires reorganisation of the existing flow structure and therefore imposes additional steric restrictions on the fluorophores there, therefore increasing the viscosity and therefore fluorescence intensity and %MFE as discussed. In the wake of this interface, the solvent channels are left organised and the flow is fast, and the effective viscosity is therefore low. The fluorescence intensity and the %MFE are, therefore diminished. In the region where the diffusion is relatively fast (i.e the wake), the prompt-only MFE dominates since it is the only component of the MFE dynamics which has time to develop.

Further evidence for this theory could be extracted from the time constant images (4^{th} row in Figure 4.13). Unfortunately, the fitting of the time constant is possible only in the region of the wave where the MFE and the associated EMFE are very large. The fitted values in the path and the wake are completely uncertain (here they have been capped at 20 s). The time constant information for larger regions can, however, be extracted by considering only the last segment (S6 in Figure 4.12), and taking the path, wave and the wake as three separate ROIs. The fitted EMFE steps for these three ROIs are shown in Figure 4.14. A much better signal-to-noise ratio as a result of averaging over a larger region allows a clear fitting of the EMFE. The time constants for the EMFE steps are given in the figure. The overall trend appears to be $\tau_{\text{wake}} > \tau_{\text{wave}} > \tau_{\text{path}}$. This seemingly agrees with the theory of opposing flows described above, although further simulations are needed to understand it better.[ep] Note that the trend in the fluorescence intensity is $I_{\text{wake}} < I_{\text{path}} < I_{\text{wave}}$.

4.5 Conclusions and outlook

The FMN/HEWL crystals measured in this work exhibit never-before seen fluorescence dynamics which couples with the observed MFEs. The observed MFEs exhibit complicated, as-of-yet poorly understood dynamics and much larger magnitude than those observed in solution. It has been shown how these features of the data can be extracted, presented and analysed. Further, a tentative explanation in the form of coupled

[ep] Preliminary measurements suggest that this feature is not observed when fluorescent probes other than FMN, which are either photostable or weakly interacting are used. This suggests the importance of photodegradation and/or strong probe-crystal interactions in the development of the wave-like feature observed.

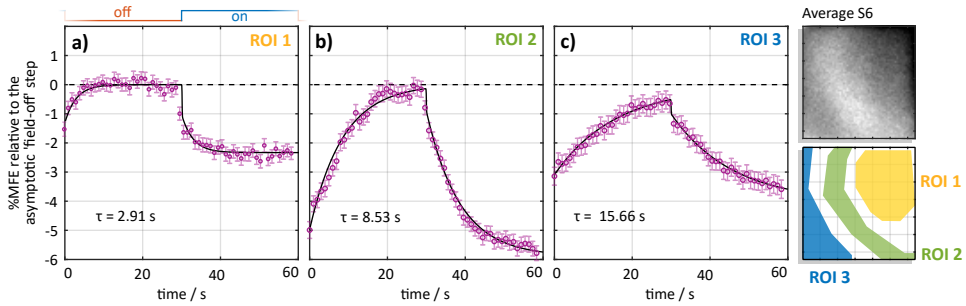


Figure 4.14: Fitted EMFE for three ROIs of the last step (S6). The fitted time constants are given on the plots. The steps have been shifted down by the asymptotic value of the %MFE to correspond to the negative MFE. The fit function is given in Listing A.21. The ROIs are selected in the **a)** wake **b)** wave and **c)** path regions, and are shown in detail next to the plots.

reaction-diffusion with frustrated flow has been proposed. A more detailed understanding of the dynamics of the diffusion and interaction of FMN (and its photoproducts) with the crystal is of utmost importance for further research on this system. The following are suggested as a future investigation direction:

- The processes, as well as their coupling are clearly complicated and therefore difficult to understand by intuition alone. The only way to understand such a system is, therefore, to simulate it. A typical way to simulate reaction-diffusion processes is with the bulk concentration approximation. The interactions involved here are, however, on a molecular scale and hence a stochastic discrete-molecule simulation may be more appropriate.[27]

- The details of the interaction of the fluorophore with the crystal have to also be understood much better. Fluorescence lifetime imaging (FLIM) could be an appropriate technique such an investigation, since it can provide the information about the fluorescence quenching rates and therefore the local environment of the flavin molecules.

- In order to investigate the observed 'wave' phenomena, measurements of the interactions of other dyes with the HEWL crystals ought to be investigated. An interesting target could also be crystalline Glucose Oxidase (GOx, see Chapter 5) since it binds flavin as a cofactor. This would allow the investigation of the spatiotemporal fluorescence of flavin in absence of chemical diffusion.[eq]

- Preliminary measurements suggest the photodegradation of flavin proceeds through a different pathway when photoexcited at the red shoulder (488 nm) of the $S_0 \rightarrow S_1$ absorption band as opposed to at the absorption maximum (~450 nm). The photodegradation of flavin is significantly reduced with the red-shoulder excitation.[er] Since the phenomena described in this chapter are clearly coupled to the dye photodegradation, their investigation with the red-shoulder excitation would be of great interest.

- Finally, the specific focal plane slice taken through the crystal could affect the measurement. These measurements have been performed on a confocal microscope which exhibits a large degree photoexcitation out of the focal plane.[es] A comparative measurement on a total-internal reflection (TIRF) microscope, or a 2-photon microscope would allow for an interesting comparison – to determine whether the out-of-plane excitation plays a role in the phenomena observed.

While FMN/HEWL crystals provide a new take on a known magnetosensitive system, the investigation of MFEs on porous crystalline systems is a new exciting direction for the MFE research in general. Such

[eq] The only observed phenomena would therefore be photodegradation, excitation diffusion and photorecovery.

[er] Note that both of these wavelengths excite *the same* absorption band. The extinction coefficient of flavin is smaller at 488 nm, but the power of the laser has been adjusted accordingly to detect a comparable signal. This suggests that the photodegradation proceeds through a secondary absorption of the photon by the excited state: $FMN^* \rightarrow FMN^{**}$. A lower photodegradation yield can then be caused by the absorption of light by FMN^* being lower at 488 nm (even accounting for increased power).

[es] In a single-photon microscope all the excitation and photodegradation also occurs out-of-plane of the excitation, it is just not focused onto the detector in the same way as the in-plane.

systems could allow to study the details of the interactions of the fluorophores with their environment and therefore progress our understanding of the ET reactions leading to the MFEs. They could also potentially be used as biological model systems of MFE anisotropy, or as organic ET devices.

Chapter 5

Magnetic field effect measurements with cavity-enhanced absorption spectroscopy

This chapter will outline the principles of the cavity-enhanced absorption spectroscopy (CEAS) technique, used in the study of proteins in the following Chapter 6. The origin of sensitivity, newly developed calibration method and limitations of the technique will be discussed. This will be done in the context of measurements of absorption of liquid samples.

Very brief history of cavity-enhanced spectroscopy

The earliest example of an optical cavity is the Fabry-Pérot interferometer – two mirror optical whose output is focused to create a pattern of concentric circles corresponding to resonant pathlengths of light in the cavity. Such devices were mostly used akin to a very narrow bandpass filter to allow measurements of high resolution atomic spectral lines. The first documented recordings of inter-cavity absorption (or the atomic beam of barium) were made by D. A. Jackson in 1961.[114] In Jacskon's work, however, the (scanning) Fabry-Perot cavity was used only for the wavelength selection. The first demonstration of the use of the use of a cavity for the *sensitivity enhancement* was the 1988 measurement of the differential cavity ringdown time by O'Keefe *et al.*[115] CW cavity-enhanced measurement technique was developed 10 years later.[116;117] Although initially prohibited by the brightness of the light sources available,[118] the broadband incoherent light employed for cavity-enhanced spectroscopy followed soon after.[119;120] A review by Romanini *et al.* is recommended for a much more detailed history and discussion of cavity-enhanced spectroscopy.[121]

5.1 Principles of cavity spectroscopy

Cavity ring-down spectroscopy

Before the first cavity-enhanced absorption measurements by O'Keefe *et al.*, the main interest in the field of cavity spectroscopy methods was the measurement of the 'photon lifetime'.[121] This was first achieved by measuring the phase-shift of the light leaving the cavity and then by means of cavity ring-down. Both found its utility in measurements of the reflectivity of very high quality mirrors.[122] The measurement of the reflectivity of the mirrors by cavity ring-down would proceed as follows. A pair of mirrors M1 and M2, were placed facing one another, and a pulse of light was injected through M1. If the faces of the mirrors were adequately parallel, therefore forming an *optical cavity*, the pulse would bounce back and forth between the mirrors. The intensity of the pulse would decrease, losing energy to scattering and absorption of the medium between the mirrors, as well as leak out of the optical cavity though each mirror on each reflection. Assuming the latter is the only contribution, the rate of loss of light in the cavity can be related to the product of the reflectivities of the two mirrors. By convention the light is injected into the cavity though M1 and the cavity output is monitored through M2.

As the pulse travels through the cavity, the light which leaks out of M2 consists of pulses decaying in intensity by a factor of R_1R_2. If the intensity of the 0^{th} pulse is 1, the n^{th} pulse has the intensity $(R_1R_2)^n$, where R_1 and R_2 are the reflectivities of M1 and M2 respectively. These pulses are spaced in time by $^{2l}/_{c}$ where l is the length of the cavity, and c is the speed of light (the roundtrip of a cavity means the pule has travelled a distance of $2l$). This is an exponential decay, and its time constant τ is the ring-down time τ of such an

optical cavity:

$$\tau = -\frac{2l}{c\ln(R_1R_2)} \tag{5.1}$$

The ring-down measurement is, therefore, related to the product R_1R_2, as well as l – an easily measurable quantity.[et] If three different mirrors are measured pairwise, their respective reflectivities can be inferred.[eu]

The fractional error in the reflectivity measurement for two identical mirrors can be found by setting $R_1 = R_2 = R$, and differentiating with respect to τ. The fractional error in R as a function of fractional error in τ is then:

$$\frac{\delta R}{R} = -\ln(R)\frac{\delta\tau}{\tau} \tag{5.2}$$

Somewhat counter-intuitively, the higher the reflectivity of the mirrors measured, the higher the sensitivity of the measurement ($-\ln(x)$ is monotonically decreasing from $+\infty$ to 0 for $x \in [0,1]$). Equation 5.2 reflects the fact that the larger the R, the longer the ring-down time, and hence the more accurate its measurement.[ev] This sensitivity is not, however, as unbound as Equation 5.2 would suggest. As $R \to 1$, the fractional error in the measurement does, indeed go to zero, but so does the amount of light coming out of the cavity. This is not accounted for in the analysis above, but has to be taken into account in general.

The incredible sensitivity of the ring-down measurements can be harnessed for the measurements of the interaction of light with samples of interest. If a sample of interest is placed inside of a ring-down cavity, the fractional light loss upon transmission though it, T, will compound itself on per-pass light losses due to light leaking though M1 and M2. If the reflectivity of the mirrors is known or calibrated for by measuring an empty cavity, T can be inferred from the ring-down time of the cavity.[ew]

Note that, even though T is the transmittance of the sample – it is indeed defined as the ratio between the output and input light intensities – it *cannot* be equated simply with the samples absorbance losses. Fractional losses from multiple sources combine by multiplication. Hence, T can be expressed as a product of scattering T_S, and absorption T_A contributions $T = T_A \times T_S$. In a typical single-pass absorption measurement, for example, T_S is close to 1 and so for a sample with absorbance $A = 0.3$ OD (3 dB) and hence $T_A = 10^{-0.3} \approx 0.5$, overall T is dominated by the T_A term and hence $T \approx T_A$ can be assumed. In cavity spectroscopy, however, the scattering losses T_S are a large proportion of the overall sample transmittance since the overall sample's transmittance is close to unity. The scattering losses originate from the scattering of light of the surfaces of the sample cell as well as the sample itself,[ex] and therefore cannot be easily calibrated for.

For that reason, many cavity measurements employ differential absorbance measurement. The samples' ground-state absorbance is then not the quantity of interest. Instead, the *change* in samples' absorbance is

[et] l can either be measured directly or, inferred from the roundtrip time of the cavity. The roundtrip time can, however, be measured only if the length of the laser pulse is shorter than a roundtrip time, and the detector used is fast enough to resolve individual pulses exiting the cavity.

[eu] Following Equation 5.1, the product of the reflectivities for the combination of the mirrors i, j and k is $R_iR_j = \exp\left(-\frac{l}{c\tau_{ij}}\right) = r_{ij}$ with cyclic symmetry $i \to j \to k \to i$. The (nonlinear) equations for three pairwise reflectivities can then be solved to give $R_i^2 = r_{ij}r_{ki}r_{jk}^{-1}$.

[ev] This is the case for three main reasons. Firstly, with constant error in the ring-down measurement $\delta\tau$, the fractional error $\delta\tau/\tau$ drops with increasing τ. Secondly, the contribution of the detector noise to the measurement is spread over multiple points in longer ring-down trace, and hence can be fitted more accurately. Thirdly, longer ring-down traces take much longer to decay down to the detection limit of the detector and hence there are more data points available for fitting of τ.

[ew] The discussion in this chapter is confined to sensing using free-space optical cavities. This is, by no means, the extent of the field of cavity-based sensing. The notable examples include linear and fiber-loop cavity ringdown spectroscopy, [123] as well as the whispering-gallery mode microresonators.

[ex] There are many reasons a sample may scatter light. i) If the refractive index of the sample is significantly different from that of the sample cell, the interface between the cell walls and the sample may reflect the wavefront of incident light in a way which will make it exit the cavity. ii) The sample may scatter light due to large protein aggregates present therein, or other sources of turbidity like sediment, microbial growth, silica from the filtration unit or calcium carbonate sediment due to hard water. iii) If the sample is flown though the cavity, micro-bubbles of air can enter the sample through a permeable silicone tubing, or otherwise through not fully-sealed joints in the tubing, due to negative pressure inside of the tubing induced by flow. The faster the flow, the larger the pressure drop and the higher the possibility of such micro-bubbles. iv) If the sample is a mixture of solvents of different viscosities (e.g. water and glycerol), or if the sample cell has been washed with solvent of one viscosity before introducing a sample of another one, the mixing currents of different solvents can scatter light. v) Flowing the sample through the cavity can induce mechanical flex in the sample cell holder, or the sample cell itself. This can alter the alignment and therefore introduce scattering losses to the cavity. vi) Finally any micro-scatterers in the air (dust) will contribute to the scattering of light in the cavity.

measured. The assumption of negligible scattering is then non necessary, so long as the scattering stays constant over the course of the differential measurement. The most typical absorption differential measured in this way is the change in sample's absorbance upon photoexcitation.

The measurement of a sample's transient absorbance by measuring the ring-down of a cavity is referred to as Cavity Ring-Down Spectroscopy (CRDS). Such a measurement is usually done in a pump-probe fashion, with the pump pulse exciting the sample directly, followed by a probe pulse which is allowed to enter the cavity though M1 and ring down while being measured through M2. The pump-probe delay is allowed to vary through a range from negative to positive values, and the ring-down time measurements are accumulated over the course of multiple pump-probe events.[ey] The negative values of the pump-probe delay correspond to the background measurement, and the positive ones to measurements of the photoinduced transient. A big advantage of CRDS is that it is insensitive to the fluctuations of the probe-beam intensity, since only the timescale of the ring-down event is measured, not its absolute magnitude.

Optical resonator

So far, the wave-like nature of light has been ignored, and the light was treated as a ray. The absorption and scattering was expressed as simple fractional losses, reflections off surfaces as a simple Newtonian-like collision, and refraction using Snell's law.[ez] This model of light as rays is the basis of geometrical optics, and will will turn out to be mostly sufficient for understanding of cavity spectroscopy in the condensed phase.

One of the fundamental features of cavity spectroscopy cannot, however, be explained by geometrical optics. Optical cavities are sometimes referred to as *optical resonators*. This is because only certain frequencies of incident radiation can resonate and create a standing wave within the cavity. When the length of the path light travels inside of the cavity corresponds to a integer number of its wavelengths, it creates a constructive interference pattern with itself. The intensity of light inside of the cavity builds up and, eventually, exits the cavity out of both M1 and M2. Other frequencies of light create a destructive interference pattern, which causes the cavity to radiate significantly less energy out of M2. The overall energy of the radiation field is, however, conserved. This is because the destructive interference pattern created inside of the cavity causes the incoming radiation field to reflect off M1 (in addition to the reflection caused by M1 itself) before entering the cavity.[fa]

An optical cavity will, therefore, be able to support only certain frequencies of light. The spacing between the supported frequencies is referred to as the free spectral range (FSR). In the simplest case of purely-transverse travel, the light will interfere constructively only when twice the cavity length $2l$ (the roundtrip back and forth though the cavity) is an integer multiple of the wavelength of the light $2l = n\lambda$. The FSR f_{FSR} is, therefore:

$$f_{\mathrm{FSR}} = \frac{c}{2l} \tag{5.3}$$

[ey] The measurement like that is usually done with a single colour probe beam. If spectral resolution is desired, the experiment can be repeated over multiple wavelength points by using a tuneable laser (e.g. a dye laser with tuneable resonator or an optical parametric generator). An alternative is to use a pulsed white light source as the probe, and record the ring-down of the entire spectrum at the same time. Such simultaneous time and wavelength resolution could be achieved with, e.g. spinning-mirror streak camera coupled to a spectrograph.

[ez] Snell's law is derived from the postulate that the light ray always takes the shortest path between two points. Geometrically this turns out to be equivalent to $\frac{\sin\theta_1}{\sin\theta_2} = \frac{v_1}{v_2}$ where θ and v are the angles the incoming and the refracted beam make with the normal of the surface, and their ray velocities (phase velocities of the wavepackets). The refractive index of a material is the ratio of speed of light in vacuum, to the speed of light in that material $n_i = \frac{c}{v_i}$. Hence, the common form of the Snell's law: $n_1 \sin\theta_1 = n_2 \sin\theta_2$.

[fa] A similar principle of multiple constructive and destructive interference patterns is, in fact also the basis of how the broadband dielectric supermirrors comprising the cavity are created. Such mirrors have multiple layers of thin precision coatings of varied refractive index. The reflective/transmitive effective of all the micro-cavities created between the layers results in an overall pattern of reflectivity which is then optimised (see section 2.2) as a function of the thickness' of all the dielectric coatings.

The intensity of light transmitted through the cavity I_T as a function of its frequency f is described by an Airy distribution. For two identical mirrors with reflectivity R, the Airy distribution is:

$$I_T(f) = \frac{(1-R)^2}{(1-R)^2 + 4R\sin^2\left(\pi \frac{f}{f_{\text{FSR}}}\right)} \tag{5.4}$$

This is a comb of Lorentzian lineshapes, periodic in f/f_{FSR}.

The geometry of a two-mirror optical cavity is described in terms of the *cavity axis* – the axis between the centers of M1 and M2. If the cavity is rotationally symmetric about its axis, its geometry can be described in terms of the transverse (axial) and longitudinal (equatorial) direction. In general light can create a standing wave with any integer number (including 0) of nodes with planar or radial symmetry with respect to the equatorial dimension of the cavity.[fb] The different patterns of resonance inside the cavity are referred to as modes, and can be enumerated as M_{ij} where i is the number of radial, and j the number of equatorial nodes. FSR as given by Equation 5.3 is defined for the transverse-only mode of the cavity (M_{00}). If higher order modes of the cavity are excited, the frequency of the resonance will change slightly.

The concept of modes can be demonstrated by calculating the Airy distribution for the M_{00} case of a simple 1D cavity, as follows:

```
l = rand; R = 0.7; % Cavity length and reflectivity
fc = 5/l; % 5 oscillations per cavity length
fsr = 1/(2*l); % Free spectral range
N = max(20,round(log(1e-3)/log(R))); % Number of reflections
X = bsxfun(@plus,linspace(0,l,1e3),2*l*(0:(N-1))'); % x for all
    reflections
fs = linspace((fc-1.5*fsr),(fc+1.5*fsr),1e3); % Freqencies
Y = sin(bsxfun(@times,X,2*pi*permute(fs,[3,1,2]))); % All the reflections
Y = bsxfun(@times,Y,R.^(0:(N-1))'); % Loss of R on each reflection
% Peak-to-peak amplitude of the sum(Y) -- numerical Airy distribution
airy_calc = (1-R)^2*squeeze(2*mean(sum(Y).^2,2));
frel = (fs-fc)/fsr; % Frequency relative to the central frequency
airy = ((1-R)^2)./((1-R)^2+4*R*sin(pi*frel).^2); % Analytical Airy distr.
```

Listing 5.1: Numerical calculation of the Airy distribution

A wave representing the electric components of the light is set to travel from M1 towards M2 though a cavity of length l. Its frequency is chosen such that, arbitrarily, it makes 5 oscillations as it travels through the cavity. The magnitude of the electric field is calculated for N reflections though the cavity, where N is large enough that $R^N \leq 10^{-3}$. No phase shift is introduced on the reflections. The numerical Airy distribution is calculated as the peak-to-peak magnitude of the interference of all the reflections, scaled by the reflectivity of the input and the output mirror (such that peak-to-peak magnitude of the input light is 1).[fc] Additionally, a relative frequency vector `frel` (in units of f_{FSR}), and the analytical Airy distribution as given by Equation 5.4 are calculated.

Figure 5.1b shows the plot of `airy_calc` as a function of `frel` for different values of R. As R increases, the sharpening of the transmission peaks of the cavity can be clearly seen. The higher the mirror reflectivity, the higher the N, and the more reflections one has to sum over to obtain the overall intensity pattern. The first 10 reflections of the electric field inside of the cavity, for a point just off-resonance (indicated in the figure), have been plotted in Figure 5.1a. The first few reflections keep broadly in-phase, and if the reflectivity of

[fb] Looking down the cavity axis, the planar and radial nodes would appear as lines and circles in the intensity pattern of light inside of the cavity. The mathematical description is closely related to that used for atomic orbitals; n^{th} shell orbitals having n planar (conical) or radial nodes.

[fc] This piece of code makes a heavy use of `bsxfun` function to avoid multiple `for`-loops. As a quick reference, `bsxfun` is a **b**inary **s**ingleton e**x**pansion **fun**ction. It '*applies element-wise operation to two arrays with implicit expansion enabled*'.[124] For example `C = bsxfun(@plus,A,B)` where A is a $N \times 1$ *column* vector, and B is, similarly, a $1 \times N$ *row* vector. C is then a $N \times N$ matrix, where $C_{ij} = A_i + B_j$. In newer versions of Matlab this behaviour is implicit for all typical binary operators (addition, multiplication, exponentiation, min, max, mod, atan2 and various boolean operations).

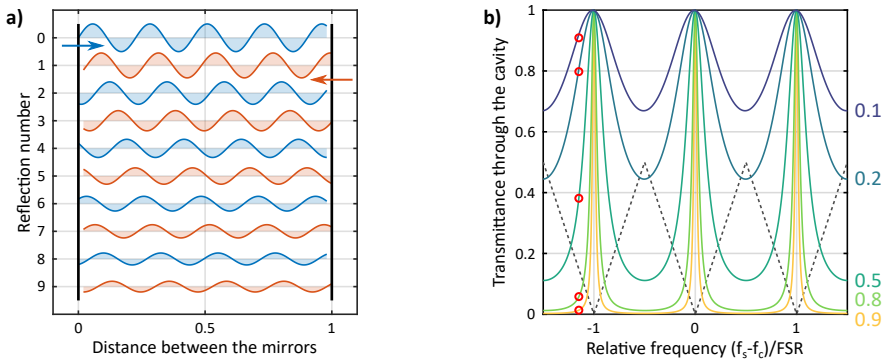


Figure 5.1: Numerical calculation of the Airy distribution. **a)** Individual reflections of the electric component of the electromagnetic field in the cavity. The wavelength does not quite match the spacing between the mirrors, and so the reflections can be seen to fall out of phase with one another. **b)** Numerically calculated Airy distribution for different values of the mirror reflectivity (indicated on the side of the figure), as calculated in Listing 5.1. The dashed line indicates the distance or the quantity $\frac{lf}{c}$ (cavity length divided by the wavelength) from the nearest integer.

the mirrors is low, a few reflections is all the persistence time any quantum of radiation will have. The higher the reflectivity of the mirrors, the more out-of-phase the radiation gets. Note that the analytical Airy distribution has not been plotted at it is visually identical to the numerically calculated one. The ratio between the width of the individual transmission peaks of the cavity, and the free spectral range is referred to as the finesse, Q-factor, or the frequency-to-bandwidth ratio of the cavity.

Cavity modes are a fundamental of much of gas-phase spectroscopy, but have not had much influence of condensed-phase measurements such as those which will be discussed here. The reasons for this are important to consider, and are as follows:

- A coupling of a wavefront-matched beam into the center of M1, excites only the M_{00} mode of the cavity. Any other excitation will additionally excite, to a greater of lesser degree, transverse modes which, as discussed, have different free spectral range.[125] The Airy distributions of different modes will not match up and will therefore constitute an additional range of wavelengths over which the cavity can be excited.[fd]
- Short pulses of light, such as those used in ring-down measurements, have very high bandwidth due to time-frequency uncertainty. Broadly speaking, a pulse of length t will have a bandwidth of t^{-1} which, if greater than the FSR of the cavity, guarantees the pulse will couple into at least one of the cavity modes.[fe]
- Condensed-phase samples tend to have very broad absorption bands due to, amongst others, molecule-solvent interactions giving rise to a multitude of possible electronic states, overall absorbance being the average of them.[ff] This means that features of interest tend to span multiples of the FSR and therefore be detectable regardless of where exactly they do occur. For example, the FSR-equivalent wavelength spacing of 500 nm light (green) in a 20 cm cavity is ca. 1×10^{-1} nm (1.5 GHz), while an absorption feature can span on the order of 100 nm.
- Further to the last point, the FSR is usually much smaller than the resolution of the detectors used in condensed-phase measurements. A typical spectrograph with a scientific camera will achieve a wavelength resolution of ca. 1 nm. All the light coming from ca. 1×10^3 modes which span each pixel will just get accumulated together.[fg]

[fd] This is, sometimes, done on purpose (e.g. in integrated cavity output spectroscopy) as it can lead to greater stability of the resonator at the price of lower transmission.

[fe] Narrow bandwidth of pulses, and therefore coupling into only a few cavity modes can cause high intensity noise in ring-down measurements. However, ring-down spectroscopy is insensitive to intensity fluctuations.

[ff] The absorption bands of species of interest in condensed phase are often confined to visible (wavelength range of ca. 300-800nm), near-UV (below down to ca. 180nm, the range of circular dichroism spectroscopy) and near-IR (up to ca. 2 micron) regions.

[fg] A resolution high-enough to observe cavity modes can, of course, be achieved with different style of detector e.g. Fourier-transform optical spectrum analyser which has a resolution of ca. 5 GHz.

All of the above points apply to the broadband CEAS measurements performed in this work. The BBCEAS apparatus uses a supercontinuum (SC) white light source to probe the absorbance of the sample inside of the cavity. The spatial coupling of the beam (adjustment of the injection point, angle, beam diameter, and focus) has to be optimised to a lesser degree than in high resolution gas-phase spectroscopy. Furthermore, even though the SC source is used akin to a CW source, its output is pulsed with frequency of ca. 70 MHz. This, combined with the broadband nature of the source, causes the light to couple into multitude of modes. The SC light is bright in the range of 400–800 nm (ca. 10^6 FSRs of a 20 cm long cavity), and the detector resolution is ca. 0.5 nm.

Cavity-enhanced absorption spectroscopy

The technique used in this work is not, however, ring-down spectroscopy, but its time-integrated variant – cavity-enhanced absorption spectroscopy (CEAS). The probe light entering the cavity in a CEAS experiment is not pulsed but originates from a continuous wave (CW) source. The sample of interest is placed inside of the cavity, and the the output of M2 is monitored. It turns out that the light output of such cavity can be very sensitive to tiny absorption changes of the sample placed therein. This sensitivity has, ultimately, the same origin as the sensitivity of the CRDS experiment.

To illustrate, consider the output of a CRDS cavity following an input of a single pulse of width $\mathrm{d}t$. Both M1 and M2 have reflectivity R, and the sample inside of the cavity has a fractional transmittance loss T. If the light input intensity is defined as 1, the ring-down trace (output of M2) has the form:

$$f(t) = T(1-R)^2 \begin{cases} \exp(-t/\tau) & t \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{5.5}$$

where τ is:

$$\tau = -\frac{2l}{c\ln(RT)} \tag{5.6}$$

derived by the same consideration which leads to Equation 5.1. If another pulse of width δt were to arrive at the input of the cavity with a delay of $\mathrm{d}t$, the output would be a sum of two exponential decays, the second one shifted forward in time by δt. A CW radiation can be considered as a continuous stream of such pulses, each with the delay k of an integer multiple of δt. The cavity output is, then, defined as a sum of all the resulting ring-down traces. In the limit of $\delta t \to 0$ this reduces to an integral over k:

$$I_{\text{out}} = \frac{\tau^{-1}}{1-R^2T^2}\int_0^\infty f(t-k)\,\mathrm{d}k = \frac{(1-R)^2\,T}{1-R^2T^2}\begin{cases} 1-\exp\left(-\frac{t}{\tau}\right) & t \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{5.7}$$

The scaling factor accounts for the finite nature of the pulses in CRDS – the integral over the ring-down trace ought to be equal to sum of all the pulses, and so:

$$\frac{\sum_{j=0}^{\infty}(RT)^{2n}}{\int_0^\infty \exp\left(-\frac{t}{\tau}\right)\mathrm{d}t} = \frac{-c\ln(RT)}{2l(1-R^2T^2)} = \frac{\tau^{-1}}{1-R^2T^2} \tag{5.8}$$

Equation 5.7 is an exponential growth of the cavity output with time constant equal to the ring-down time. The limiting behaviour of Equation 5.7 as $t \to \infty$ – the accumulated cavity output is:

$$I_{\text{out}}^{\infty}(T) = \lim_{t\to\infty} I_{\text{out}}(t,T) = \frac{(1-R)^2\,T}{1-R^2T^2} \tag{5.9}$$

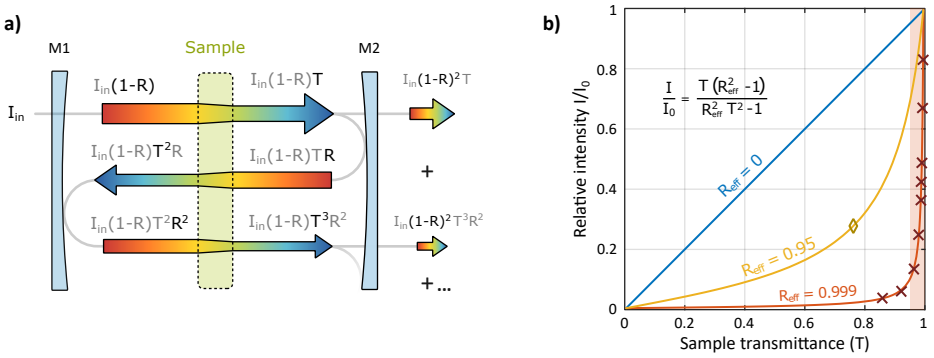


Figure 5.2: a) Schematic of the light roundtrip though the cavity, as described in the main text. The sum of the light leaking through the output mirror after an odd number of roundtrips is the integrated output light intensity of the cavity. **b)** Transfer function from the sample transmittance to the observed relative light intensity. The point were the gradient of the transfer function is 1 has been indicated in the $R_{\text{eff}} = 0.95$ curve with ⋄. An example set of calibration points measured has been shown under the $R_{\text{eff}} = 0.995$ curve.

Since CEAS employs detectors with much longer integration time than CRDS, this limiting value will be the intensity of light observed.[fh] The intensity output will, most often, be measured with respect to the output of the cavity with no sample present. This can be expressed as a ratio:

$$\frac{I_{\text{out}}^{\infty}(T_A T_S)}{I_{\text{out}}^{\infty}(T_S)} = \frac{I}{I_0} = \frac{T_A\left(1-R_{\text{eff}}^2\right)}{1-R_{\text{eff}}^2 T_A^2} = F\left(T_A|R_{\text{eff}}\right) \tag{5.10}$$

where $R_{\text{eff}} = R\,T_S$ is the effective reflectivity of the mirrors – scaled down by the scattering losses T_S inherent to the setup.[fi] Equation 5.10 will be used in the process of data analysis and the following discussion.

The ratio of measured and reference intensities I/I_0 is plotted in Figure 5.2b as function of the sample transmittance T_A. Equation 5.10 can be thought of as a transfer function F from the measurable quantity – the ratio of M2 output intensities with sample and with reference, to the quantity of interest – the transmittance (and therefore absorbance) of the sample itself. Crucially, this is a *non-linear* function of T_A. For large values of R, the transfer function is mostly flat, and it peaks sharply as it approaches $R = 1$. As $R \to 0$, Equation 5.10 approaches $I/I_0 = T$ – the transfer function for a UV-vis spectrometer. The gradient of the transfer function can be thought of as the sensitivity to changes in T_A since, when the gradient is large, a small change on the abscissa (transmittance) will yield a significant change on the ordinate (observed intensity). The CEAS experiment will, therefore, remap the dynamic range of the detector to a narrow range of sample transmittance around unity, and therefore enhance the sensitivity of the measurement to small changes in absorbance. Notably, if the effective reflectivity of the mirrors drops below the point where the gradient of the transfer function becomes smaller than 1, the sensitivity of the cavity experiment becomes *worse* than that of a single-pass experiment. The critical transmittance T_c of the sample, past which the sensitivity of the experiment is below 1, can be well described by:[fj]

$$T_c = \frac{9-8\sqrt{1-R_{\text{eff}}}-R_{\text{eff}}}{1+7R_{\text{eff}}} \tag{5.11}$$

For the BBCEAS experimental apparatus described later, the mirror reflectivity is $R_{\text{eff}} \approx 0.998$. The critical sample absorbance (in 1 cm path length cell) is therefore $-\log_{10}(T_c) \times 10 \approx 190\,\text{mOD}$.

The above analysis is useful for understanding how CEAS sensitivity arises, based on the theory of CRDS. A more typical way of analysing the cavity output intensity yields an identical result.[119] Briefly, considering

[fh] Any kinetics observed will be convolved with an exponential with time constant τ. Since the time resolution of the CEAS experiment is usually not the main concern, this can most often be disregarded.
[fi] Note that the square in the numerator in Equation 5.10 is *inside* the bracket, while in Equation 5.9 it is *outside*.
[fj] This equation has been found as the Padé approximant of the point when the derivative of Equation 5.10 equals 1.

the sum of output of the cavity after each pass, as shown in Figure 5.2a:

$$\begin{aligned} I_{\text{out}}(T) &= (1-R)^2\,T + (1-R)^2\,R^2\,T^3 + \dots \\ &= (1-R)^2\,T \sum_{j=0}^{\infty} (RT)^{2j} = \frac{(1-R)^2\,T}{1-R^2T^2} \end{aligned} \tag{5.12}$$

which is equivalent to Equation 5.9. This summation is sometimes referred to as the *superposition principle.*[126]

5.2 Resonator alignment

A proper alignment of an optical resonator is non-trivial, yet very important for cavity-based spectroscopy. The ease and stability of the alignment are, primarily, a function of the geometry of the mirrors. A cavity consisting of two plane parallel mirrors, for example, is infinitely unstable with respect to any perturbation of the mirrors of the beam inside of the cavity. That is to say, if the beam of light travels through such a cavity perfectly parallel to the cavity axis, any angular deflection of the beam will persist over the reflection and will result in the beam eventually leaving the cavity. The same applies for any angular deviation of the mirrors, although in this case the situation is worse, as the the misalignment will *compound* over the multiple passes of the beam and make it leave the cavity even sooner. This is not the case, for example, when one of the mirrors of such resonator is changed to a concave one.

Obviously a certain degree of misalignment, namely small enough that a pulse persistence time inside of the cavity is still much longer than the ring-down time, is acceptable and calling a plane-parallel resonator infinitely unstable may be considered a hyperbole. The ring-down time of cavity in a typical liquid-state measurement of a protein sample is on the order of 300 ns. A plane-parallel cavity with that ring-down time τ could tolerate an angular deviation θ of approximately $\theta \leq \frac{d}{10\tau c}$ where d is the mirror diameter and c is the speed of light.[fk] For standard mirror diameter of 1 in (ca. 2.5 cm), this is ca. 30 µrad – an positioning accuracy not easily achieved, not to mention maintained over long periods of time.[fl]

The stability of the cavities with different radii of curvature of M1 and M2 can be analysed rigorously with *analytical* ray-tracing. The quantity of interest turns out to be g – the geometric parameter of a mirror, defined as $g = 1 - \frac{l}{r}$, where r is the radius of curvature of the mirror in question. The cavity is stable if the product of the geometric parameters of the two mirrors is between 0 and 1:[127]

$$0 \leq g_1 g_2 \leq 1 \tag{5.13}$$

The condition for stability of an optical cavity as given by Equation 5.13 can be verified with *numerical* ray-tracing. The problem can be set up in 2D space parametrised by the complex plane. The mirrors of appropriate curvature are placed with their centres at $\pm 1 + 0i$ parallel to the imaginary axis. The mirror diameter is kept constant at $d = 1$. The ray of light can be propagated through space with a differential equation solver like `ode23` in Matlab. The beam is set to start between the mirrors and propagate until it either hits a mirror, or radius 1.5 circle centered on origin. In the former case, the reflection is calculated (up to `Nmax` reflections), while in the latter one the beam is deemed to leave the cavity. The problem for single pair of (g_1, g_2) values is solved as follows:

[fk] Consider a cavity of length l with mirrors of diameter d. A single pass though the cavity with beam deviation form parallel with the cavity axis of θ, displaces the beam laterally by $2l\tan\theta$. For a given ring-down time τ, the number of times the light passes though the cavity is approximately $5\tau\frac{c}{l}$ (since 5 time constants of an exponential decay corresponds to a decay by ca. 99 %). The 'tolerated' angles are those when the lateral displacement on all of the passes is smaller than the diameter of the mirror $\frac{d}{2l\tan\theta} \leq 5\tau\frac{c}{l}$. Furthermore, $\tan(\theta) \approx \theta$ for small θ. Hence $\theta \leq \frac{d}{10\tau c}$.

[fl] For comparison, high-quality mirror mounts from Thorlabs claim a 7 mrad/revolution resolution. By hand, one could achieve approximately 5° turning accuracy which still leaves one with 100 µrad resolution. Much better accuracy can be achieved with motorised mounts. High-end Physik Instrumente GmbH mounts with stepper motors claim 300 nrad step size.

```
Nmax = 100; % Maximum number of bounces
f = @(t,y) [y(2);0]; % Ray dynamics function (constant velocity)
options = odeset('RelTol',1e-6,'AbsTol',1e-6,'MaxStep',0.01);
ps = linspace(-1,1,100)/2; ps([1 end]) = []; % Launch points
M1 = make_mirror(-0.41,1); M2 = make_mirror(1,2);
options.Events = @(t,y) eventfun(t,y,M1,M2);
for j = 1:numel(ps)
    yc = [0+ps(j)*1i 1+0i]; % Starting position
    Y = yc(1); T = 0; % Store bounce positions and times
    for n = 1:Nmax
        [t,y] = ode23(f,[0 100],yc,options); % Raycast
        yc = impactmap(y(end,:),M1,M2);
        Y = [Y; yc(1)]; T = [T; t(end)+T(end)];
        if abs(yc(1))>=1.5, break, end % Beam escaped the cavity
    end
    N(j) = n; % Number of reflections for j'th launch point
end
```

Listing 5.2: Raytracing simulation of the cavity stability

The `make_mirror()` function creates a set of (complex) points defining the surface of each of the two mirrors. `eventfun()` is responsible for stopping the ode solver when the ray encounters a mirror, or escapes the cavity. Firstly the escape condition is calculated as `abs(yc)>=1.5` where `yc` is the current position of the light packet. Then the collision of the light packet with the mirrors is checked. The mirror is defined as an ordered set of points in $\mathbb{R}^2$. Two closest points to `yc` are found, and the right-angle (signed) distance from the line defined by these points to `yc` is calculated. The collision event is triggered if this distance becomes negative. Only collision with the mirror of the left/right half of the plane is checked, depending on where is the light packet located. Finally, `impactmap()` calculates the reflection of the light packet of the surface of the mirror. A random, normally-distributed angle error, with standard deviation of 1 mrad is added to the beam on each reflection.[fm] The listings for `make_mirror()`, `eventfun()`, and `impactmap()` can be found in A.7, A.8 and A.10 respectively. The launch position of the beam is kept in the center of the cavity, and its position through a set of points along the imaginary axis. The number of reflections for each position is finally recorded.

Running the above code for each pair of the geometrical factors for the two mirrors a range allows one to map the stability region of the optical cavity. The average number of reflections as a function of g_1 and g_2 is plotted in Figure 5.3a. Two simulations for the indicated points are also shown in Figure 5.3b. The stability region, as predicted by Equation 5.13 can be clearly seen. Additionally, resonators in which both mirrors have positive values of g (top right quartile) can be seen to be, on average, more stable than those with two g values (bottom-left quartile). Positive value of g corresponds to the focal lengths of the mirrors being longer than the length of the cavity ($r > l$). Interestingly, there is a region of stability in the top-left quadrant of the simulation, not reflected in the bottom-right one. The lack of symmetry about $g_1 = g_2$ axis of this simulation is attributed to the raycasting being done only towards M2, as opposed to in both directions. The region of stability in top-left quadrant of Figure 5.3a is attributed to formation of pseudo-stable folded, and bow-tie resonators within the two-mirror setup.

The BBCEAS experiment used in this work (described in detail in section 5.4) uses a ca. 20 cm cavity with mirrors of radius of curvature of ca. 2 m both. This results in $g = 0.9$ which indeed places it in the high-stability region. Note though, that the ratio of the mirror diameter to cavity length will affect the exact shape of high-stability region (but not the overall stability region as defined by Equation 5.13), and therefore this simulation is not fully representative of the experimental setup used.

[fm] This simulates small angle deviations in the beam due to scattering by dust on the surfaces of the mirrors, and small dust particles in the air. Furthermore, false-positives could be caused by the discretisation of the mirror surface into $N-1$ linear segments. Random reflection angle allows one to, effectively, average over multiple possible reflection positions.

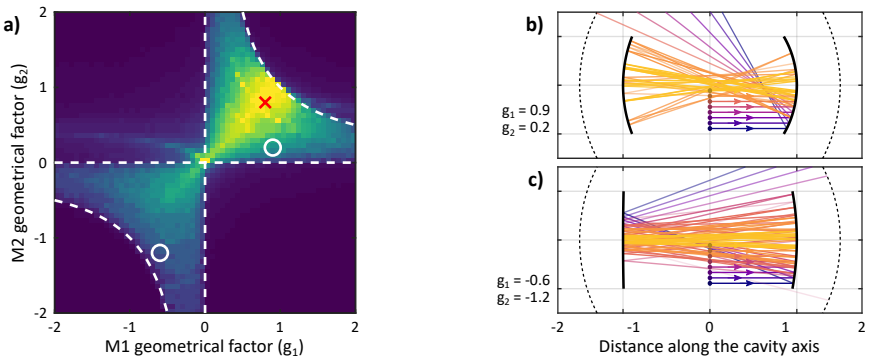


Figure 5.3: Stability condition for an optical resonator. **a)** Average number of reflections of a ray inside of the cavity, as a function of the geometrical factors of the two mirrors, as calculated in Listing 5.2. The dashed white lines indicate the theoretical stability region as given by Equation 5.13. The Red cross indicated the approximate position of the cavity used in this work. The two white circles indicate the positions of the example simulations showcased in b) and c). Note the unexpected region of stability it the top-left quadrant of the figure discussed in the main text. The colour axis, and the simulation cap at 100 bounces. **b) & c)** Resonator stability simulation for an (almost) concentric and hemispherical arrangement of two mirrors (b) and c) respectively). Different colours indicate consecutive starting points of the rays. The geometrical factors for these points are given in the figure. The dashed line indicates a radius 1.5 circle centered on the origin – the kill-plane past which the ray is considered to have left the cavity (see Listing A.8 for more details). Only launch points below the origin have been shown for the ease of viewing.

5.3 CEAS calibration

As discussed in section 5.1, the way Equation 5.10 has been derived in this work is not conventional. The way it will be used will also prove non-standard. The typical approach is to attribute all of the transmittance loss of the sample to the Beer-Lambert-like absorption, such that $T_S = 0$ and $T = T_A = \exp(-\alpha d)$ where d is the path-length of the absorbing sample, and α is the absorption coefficient. Equation 5.10 can then be rearranged for α and, after applying 1st order series expansion about $R_{\text{eff}} = 1$, expressed as:

$$\alpha \approx \frac{1}{d}\left(\frac{I_0}{I} - 1\right)(1 - R_{\text{eff}}) \tag{5.14}$$

The quantity $I_0/I - 1$ becomes the observable (note inverse of the I/I_0 fraction). The quantity $(1 - R_{\text{eff}})^{-1}$ is termed the *cavity enhancement factor* (CEF), as it quantifies the increase in the observable light attenuation by the sample as compared to a single pass experiment. Notably R_{eff} is often added as an afterthought, rather than included in the analysis from the very beginning.

The procedure suggested here is different. It is proposed that that F, as defined by Equation 5.10 is already well suited for interpreting the measured light intensity in terms of the sample's interaction with light. F is a transfer function from the fractional output of the cavity (ratio of two observables) to the sample transmittance losses (property of interest). The single unknown – R_{eff} can be calibrated for with a known absorber (hence it has been singled out as a parameter). The (fractional) cavity output light intensity of the sample of interest can then be easily related to its T by applying F in reverse – applying its functional inverse. The equation of F is:

$$F(T_A \,|\, R_{\text{eff}}) = \frac{T_A(1 - R_{\text{eff}})^2}{1 - R_{\text{eff}}^2 T_A^2} \tag{5.15}$$

The closed form of F's functional inverse is then:

$$F^{-1}(I_{\text{rel}} \,|\, R_{\text{eff}}) = \frac{R_{\text{eff}}^2 - 1 + \sqrt{1 + R_{\text{eff}}^2\left(4I_{\text{rel}}^2 - 2\right) + R_{\text{eff}}^4}}{2 I_{\text{rel}} R_{\text{eff}}^2} \tag{5.16}$$

where $I_{\text{rel}} = I/I_0$. Moreover, it is suggested that, instead of using a single known absorber to calibrate for R_{eff}, one uses multiple successive dilutions of the same stock absorber in order to collect multiple data points

along the $^{I}/_{I_0}$ vs. T curve. The value of R_{eff} can the be fitted in a least-square sense as opposed to inferred from a single measurement.

In a single-pass absorption experiment the transfer function from fractional light intensity to sample transmittance is a line with slope of 1. The higher the mirror reflectivity, the higher the slope of F close to $T = 1$ (see Figure 5.2b). The sensitivity of the cavity experiment for very low transmittance sample can therefore be approximated as the ratio of these two slopes – the infinitesimal cavity response to an infinitesimal absorber as compared to with a single pass. At $T = 1$:

$$\left.\frac{dF}{dT}\right|_{T=1} = \frac{1+R_{\text{eff}}^2}{1-R_{\text{eff}}^2} \tag{5.17}$$

Doing the series expansion about $R = 1$ and, in addition to higher order terms ignoring the -1/2 term (for simplicity, and because it is indeed small in comparison to typical values of the CEF), yields:

$$\left.\frac{dF}{dT}\right|_{T=1} \approx \frac{1}{1-R_{\text{eff}}} = \text{CEF} \tag{5.18}$$

Hence, the traditionally used, value of CEF turns out to be the slope of F at $T = 1$ (and for $R \to 1$). This links together the procedure proposed here with one typically used. Note, though, that this indicated that the CEF is an *upper bound* on the sensitivity of the experiment. A more realistic measure could be done by following the same procedure, but instead obtaining the gradient of the line going though the $\frac{dF}{dT} = 1$ point (see Figure 5.2b). This turns out to be:

$$Q \approx \frac{1}{\sqrt{1-R_{\text{eff}}}} = \sqrt{\text{CEF}} \tag{5.19}$$

where Q is a measure of sensitivity in 'units' of the gradient of F – fractional change in intensity per fractional change in sample's transmittance. This is very close (within a factor of $\sqrt{2}$) to the maximum theoretical signal-to-noise enhancement factor for absolute absorbance measurements, found by Fiedler *et al.* with similar assumptions.[128]

The proposed approach has the following benefits:

- It does not rely on interpreting T in terms of pure absorbance and instead treats T as the observed quantity. This way any assumptions about T can be decoupled from assumptions about the cavity itself. The assumption of $R_{\text{eff}} \to 1$ is therefore relaxed. This is especially applicable to measurements in the condensed phase where there are multiple additional sources of loss present.
- The proposed method relies on procedural rigidity as opposed to analytical mathematics. This means that the final result is not based on seemingly 'arbitrary' equation (Equation 5.14), but on an overall rigid procedure, every piece of which being conceptually simple. This makes the proposed method easier to reason about and understand.
- The use of multiple calibration points followed by fitting make this method less prone to accidental outliers, and it has been found that, over many months of measurements, it has given more consistent results.

Note also that the method used here works with T instead of α. Fractional loss, alongside of fractional reflectivity, is a natural way of expressing changes in light intensity in this model of the cavity.

Since F cannot be linearised, it has to be fitted to the measured data by numerical optimisation. As F is a function of single optimisation parameter – R_{eff}, the chosen method has been to employ a 1D golden section algorithm (Matlab's `fminbnd`). The following is a simplified excerpt from the analysis code which is responsible for fitting R_{eff} across all wavelengths.

```
Irel = bsxfun(@rdivide,I,I0); % Relative intensity of dye measurments
dye = interp1(wl_dye,dye,wl,'spline'); % Interp. reference to correct axis
dye = dye*10; % Correct for the path-length difference
dils = [200 100 25 7 4 3 2 1]; % Dilutions
T = 10.^(-bsxfun(@rdivide,dye,dils)); % Transmittance of dye measurments
% Irel and T are wavelength x n of dilutions matrices
for wli = 1:numel(wl)
    fun = @(reff) fit_Reff_optimfun(T(wli,:),Irel(wli,:),reff);
    Reff(wli) = fminbnd(fun,0.9,1);
end
```

Listing 5.3: Spectral fitting of R_{eff}

Where `fit_Reff_optimfun` is the optimisation function which expresses the distance between the fit and the data:

```
function diff = fit_Reff_optimfun(T,I,Reff)
Ifit = T.*(Reff.^2 -- 1)./(Reff.^2.*T.^2 -- 1); % Fitted intensity
dI = Ifit-I; diff = sum(dI(:).^2); % delta Irel
end
```

Listing 5.4: Optimisation function for R_{eff} fitting

`Irel` and `T` are matrices with dimensions of wavelength by number of dilutions. The fitting was found to be robust (due to bounded nature of the fitting parameter – $0 \leq R_{\text{eff}} \leq 1$) and take ca. 300 ms for all 1024 wavelength points (width of the CCD detector) with minimal code optimisation.

The optimisation function as stated in Listing 5.4 is attempting a direct fit of F – by minimising the square distance between measured I/I_0 and $F(T)$ (distance along the ordinate direction of Figure 5.4a) as a function of R_{eff}. This was successful (see Figure 5.4b) although it has been found to be prone to artefacts and preferentially sensitive to the low dilution measurements of the dye mix. A corresponding fit minimising the distance along the abscissa – distance between T and $F^{-1}(I/I_0)$ has been found to suffer the same problem by being overly sensitive to high dilutions. In order to improve this, the actual optimisation function used, optimises the (squared) *Euclidean* distance from measured I/I_0 to F. The evaluation of the distance from each of the data points to a nonlinear function F, however, would be computationally very expensive. Instead the Euclidean distance from data to F is approximated by a distance to the line defined by points $F(T)$ and $F^{-1}(I/I_0)$ (`Ifit` and `Tfit` respectively. See Figure 5.4). This yields a more reliable distance measure which is uniformly sensitive for both high and low dilutions of the reference dye mixture. The result if a much cleaner CEF free of artefacts (see Figure 5.4b).

5.4 BBCEAS for detection of MFEs

As aforementioned, the specific technique used in this work for the detection of the MFEs is broadband CEAS (BBCEAS). The specifics of the instrument, as well as challenges and aspects of data analysis used in this work for the detection of MFEs will be discussed here.

The entire experimental setup was assembled and aligned by the author. All experimental control and data analysis software was written by the author. The calibration and data analysis procedures have been developed in collaboration with Jessica Walton (MChem student 2018/19).

Details of the apparatus

The outline of the experiment can be seen in Figure 5.5a.

Mirrors The cavity mirrors are placed ca. 20 cm apart, and have 2 m radius of curvature. This places the cavity well within a stability region (as discussed in Section 5.2). The mirrors were supplied by Layertec

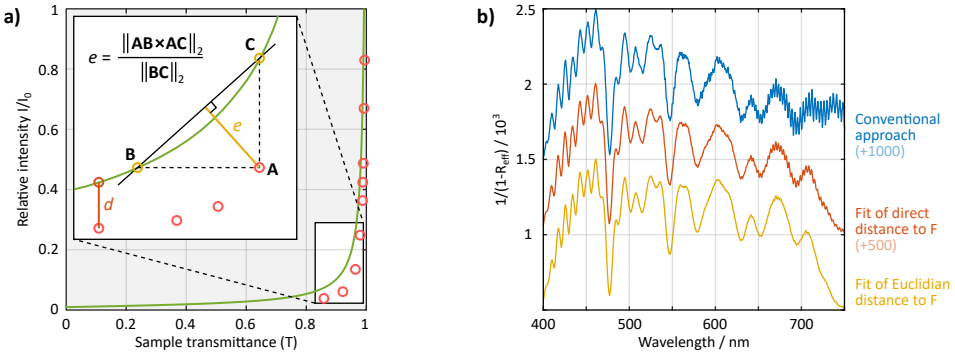


Figure 5.4: a) Example calibration data with $R_{\text{eff}} = 0.998$, acquired as described in section 5.5, and F with $R_{\text{eff}} = 0.995$. The insert shows the principle of obtaining the Euclidean (e) and direct (d) distance from data to F. The Euclidean distance is the height of the triangle ABC with vector **BC** as the base. The area of the triangle can be calculated as half of the magnitude of the cross product of vectors **AB** and **AC**. As the fit gets better, the distance e approaches the Euclidean shortest distance from point A to F. **b)** The comparison of different methods of obtaining the CEF. The same dataset has been analysed with all three methods. The conventional approach used Equation 5.14. Direct and euclidean fits optimise the respective distance from the data to F (Equation 5.15)

GmbH, and have a reflectivity of $R > 0.999$. This were confirmed by transmission measurements (see Figure B.4). The injection mirror (M1) is 1" in diameter, while the probe mirror (M2) is only ½" wide and fixed in a custom mount. This is done to allow the pump beam to reach the sample cell. Both mirrors are mounted in high stability mounts (Thorlabs KC1-PZ/M and KC1-M for M1 and M2 respectively) atop 1" posts to ensure long-term stability of the cavity.[fn] M2 mount is a piezo stage to allow for ±4 µm changes of its position with along the cavity axis. The reasoning for this will be discussed in Section 5.5.

Sample cell The sample cell (Hellma analytics, QS 165-1-40, discontinued) is placed in the center of the cavity in 1 mm pathlength precision quartz cell. The cell allows for both flowed and static samples. It is also fused with a cooling jacket which is connected to a recirculating chiller (Grant LT ecocool 100).[fo] The schematic of the sample cell can be seen in Figure B.5. The cell is mounted in 2" gimbal mount (Thorlabs GM200/M) with an aid of a custom-designed and 3D-printed mount (see Figure B.6 and B.7). The cell is set off-center to allow for cooling connections and access to the top of the cell for sample flow/insertion.[fp]

Cavity The cavity is partially enclosed in a 1" SM1 tubing with purge connectors (Thorlabs CPPC/M). A dry and filtered compressed air is flowed into the cavity at ca. 50 mLmin^{-1} (approx. the entire volume of the cavity per minute). This is done to apply a slight overpressure inside of the cavity and prevent any dust, or other microscatterers which would degrade the signal, entering the space between the mirrors. Furthermore, when performing measurements at low temperatures (below the dew point of the ambient air) the dry air lowers the humidity inside the cavity and prevents condensation on the faces of the sample cell.[fq] This is especially important for the measurements in Chapter 6, which were done at 5 °C. The SM1 tubes are connected to neither the mirrors nor the sample cell, to reduce vibrational coupling between them. Instead, they are friction fitted between the magnetic field coils. The cavity is set with its axis (the axis running through the center of the mirrors) at 15 cm above the surface of the optical table,[fr] and placed

[fn] The adjusters were changed to 100 thread per inch screws with bronze bushings (Thorlabs F19SS and F19SSP) and 1" adjustment knobs to allow for precision alignment.

[fo] The cooling liquid used is a mixture of commercial glycol (Sentinel X500) with deionised water 20% v/v, with an additive of corrosion inhibitor and biocide (Sentinel X100 and X700). The temperature of the sample cell was calibrated against the setting of the chiller (ca. 2 °C lost between the chiller and the sample cell.) Bar the first minute after start when the system rids itself of bubbles, it was found that the work of the pump in the recirculating chiller does not translate to the sample cell to any measurable degree.

[fp] This does mean that any angular adjustment of the sample cell introduces a lateral shift. This is, however, negligible within the adjustment angles used.

[fq] The dew point is ca. 15 °C in summer and 10 °C in winter corresponding to ca. 35 % and 27 % humidity as measured in the laboratory. The dry air drops the humidity to ca. 5 % which prevents any condensation.

[fr] This could be a space for future stability improvement – setting the cavity closer to the optical table which could minimise long-term shifts and vibrational coupling from the table to the optics. It is not, however, clear whether this is indeed the limiting factor for

inside of a small enclosure ca. 0.5 m on the side (this is not the laser safety enclosure around *the whole experiment*). This enclosure is also filled with dry air, and provides additional layer of protection for the cavity against dust and drafts which otherwise were found to induce vibrations in the mirrors.

Coils The magnetic field coils, as well as their power supply, are manufactured by the University of Oxford Physical and Theoretical Chemistry (PTCL) Electronic and Mechanical Workshops. The coil windings are 15 cm in diameter, and spaced apart by 7.5 cm (Helmholtz arrangement). The coil power supply is a constant-current supply which drives up to 7 A current at 16-bit resolution through the coils in either direction. This corresponds to a magnetic field of ca. ±315 G. The power supply is controlled by two voltage inputs – one for the instantaneous field and one for the constant offset, the latter being used for calibration and cancellation of long-term magnetisation due to repeated application of the magnetic fields (see Figure B.1b). This is achieved by constantly measuring the field with a 3D gaussmeter mounted inside of the coils, right next to the sample. The gaussmeter was manufactured by PTCL electronic workshops and calibrated against a commercial Gaussmeter (Lakeshore 425 with HST-4 probe) to measure 9.183 mV G^{-1} (108.9 G V^{-1}). With the full scale of ±5 V at 16-bit, this corresponds to ca. 160 mG resolution. The details of field calibration and control will be described in Section 5.5.

Probe source The cavity is probed by a white light from a supercontinuum source (NKT Photonics, SuperK EXU-6).[fs] The supercontinuum output is split into the IR and visible range by a set of hot/cold mirrors. The broadband visible power was measured at ca. 600 mW. The output of the splitter is launched from a fine positioning flexure (an older model of Thorlabs MAX603D) into free space. The output of the SC source is not a CW beam, but a train of 78 MHz, 700 ps pulses (see Figure B.8). Before the light couples into the cavity, it passes through a variable neutral density (VND) filter and a filter wheel (FW). The VND is comprised of a circular filter with variable thickness reflective coating (Newport 50Q04AV.2) mounted on a precision rotation stage (Thorlabs PRM1/MZ8). This is done such that the light intensity hitting the detector can be adjusted between the measurements. The position of the VND is recorded by the controlling software on each measurements and calibrated for in the final data (see Section 5.5). The FW allows for adjustments of the spectral shape of the light entering the cavity. The filter used for all the measurements in Chapter 6 is a heat-absorbing filter (Comar Optics, 716 GK 25) which drops the magnitude of the red-end of the spectrum and allows for more even illumination of the sample, as well as fuller use of the resolution of the detector.

Detector The light exiting the cavity is directed to the spectrograph through free space. This is achieved with a periscope assembly (Thorlabs RS99/M). The beam is coupled into the entrance slit of a spectrograph (Andor Shamrock 500i) with an achromatic doublet pair (Thorlabs MAP10).[ft] The spectrograph is of a Czerny-Turner type, with 150 lines/nm diffraction grating. The light is dispersed onto a 16 bit charge-coupled device (CCD) camera (Andor Newton 920). The sensor size is 1024×256 pixels (26 × 26 µm pixel size). The spectral dispersion due to the grating used corresponds to about 0.5 nm resolution on the detector. The light is dispersed in a ca. 400 µm tall streak across the bottom 30 rows of the CCD (ca. 700 µm). This is done such that the exposure time of the camera can be dropped and the time resolution increased. The shortest exposure time possible was found to be 1 ms (1 kHz sampling rate), but the actual exposure time used for this work was 3 ms (330 Hz sampling rate) in order to decrease the amount of light the sample is

current experiments. Furthermore, the height of the cavity is, dictated by the height of the SC fibre launch stage, as well as the size of the magnetic field coils used, and no reasonable way of changing these was found. Even if the magnetic field coils were to be set flush with the table, the considerations of residual magnetisation of the table, as well as distortion of the magnetic field by the presence of the neighbouring diamagnet (the table) would occur.

[fs] The supercontinuum generation is done by injecting high power nanosecond pulses from a 1064 nm microchip laser into a long photonic crystal fibre. [129] The accumulation of nonlinear effects inside of the fibre (self-phase modulation, Raman scattering, soliton fission, and four-wave mixing) lead to extreme pulse broadening and creation of light pulses spanning thousands of nanometres – the supercontinuum. [130]

[ft] A proper alignment of the periscope and the doublet was found to be exceptionally difficult, but necessary. A fibre coupling has been used in the past, but it was found to introduce a spectral distortion on top of the cavity output light intensity which was excessively sensitive to any vibration. Further, the throughput of the fiber-coupled cavity output was found to be poor and the alignment not stable on the timescales of the experiments performed.

exposed to, and the number of datapoints needed to be processed. All the pixels in the bottom 30 rows of the camera are added together and constitute a single spectrum. The CCD acquires all the data into an internal buffer, which is then read at the end of the acquisition. The wavelength axis of the camera (correspondence between the horizontal pixel number and the actual wavelength of light hitting it) is calibrated with a reference HgAr arc lamp (Newport, Oriel 6035). The detector is thermoelectrically cooled to –10 °C to reduce the dark noise counts (ca. 300 counts). Further cooling was found not to drop the noise floor appreciably, and take an extensively long time to reach equilibrium.[fu]

Pump source All the samples measured in this work (by the BBCEAS technique) are flavin-based, and hence have been excited at 450 nm by a diode laser (Oxxius LBX-450). The maximum power of the laser is 50 mW, but it was adjusted by an internal 500 MHz pulse-width modulation (PWM) to the desired value. The beam is focused with a 3 m radius lens and directed to the sample through two turning mirrors to allow for full control over the position and direction of the beam.[fv] The laser power was measured with power meter (Thorlabs S121C) which has been fixed in a flip-mount just before the beam enters the cavity for reproducibility of the measurement. As aforementioned, the pump beam enters the cavity at a sharp angle to the cavity axis, barely missing the M2. This is a purely geometrical consideration, as the sample cell cannot be excited from the side (see Figure B.5).

Timing control The timing of the entire experiment is controlled with a digital delay generator (DDG, Stanford Research Systems DG645). The DDG is responsible for control of the pump and probe shutters, as well as the CCD acquisition trigger and triggering of the waveform generator (Rigol DG1022Z). The latter is used in burst mode to control the instantaneous magnetic field value and create a 200 ms period square-wave burst of the magnetic field of various strengths (see Section 'Details of the MFE experiments performed' for more detail). The pump/probe shutters are manufactured by PTCL electronic and mechanical workshops, with emphasis on extremely low jitter and no opening or closing bounce (both of which were found to be an issue with Thorlabs SH1 and SH05 shutters). The shutter opening is spring-loaded while its closing is done with a DC motor. This is opposite to the way Thorlabs shutters work and is the source of the aforementioned improvements.[fw]

Control software The entire experiment is controlled with a custom written LabVIEW software (National Instruments, 2015 SP1). The software is responsible for setting correct timing of the DDG and the waveform generator, field calibration, periodic adjustment of the static offset of the magnetic field and data gathering after each acquisition. The field offset control, and the initial trigger to the DDG are done with the use of a basic AO module (National Instruments, NI-9263). The gaussmeter readings are collected with a similar AI module (National Instruments, NI-9215) at 1 kHz in 100 ms snapshots.[fx] The control software is split into four parts:

- The device interface keeps the connection to all the devices alive and performs other background tasks like the continuous reading and logging of the humidity and temperature sensor located in the cavity enclosure.
- The interface to the CCD camera is used to monitor the cavity output during alignment, CEF calibration and to perform other tests. It has certain custom-built options not available in the commercial Andor

[fu] Cooling to low temperatures also requires purging with dry air or nitrogen to remove moisture, and prevent condensation. Moreover, once cooled below ca. –20 °C, the detector cannot be simply turned off, but it must go through a careful warm-up sequence to prevent thermal shock and potential damage.

[fv] The pump/probe overlap of the beams is ensured with measurements of differential absorbance of a known sample. The pump and probe spot sizes on the sample are estimated at 1 mm and 500 μm respectively.

[fw] The spring-loaded opening of the shutter renders them non laser-safe if they were to be used by themselves. Hence, a second set of true safety shutters is also used on the experiment (not shown in Figure 5.5a).

[fx] The channels of the AI module are isolated, with separate analogue-to-digital (ADC) converters, as opposed to a single ADC multiplexed over many channels. This is important in measuring multiple signals with DC offset or from different impedance sources. In multiplexed AI devices the holding capacitors might not have enough time to discharge between the reads of the ADC which creates artefact signals from one channel onto the next. This is referred to as 'ghosting'.

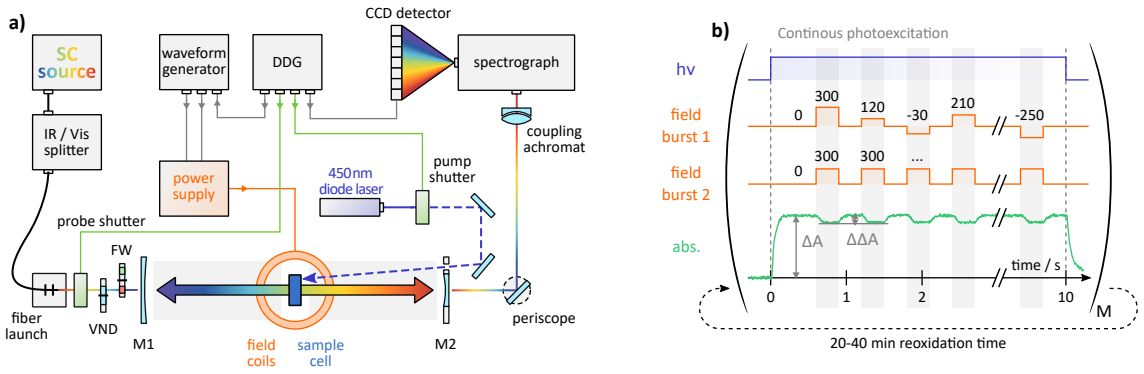


Figure 5.5: a) Outline of the BBCEAS experiment, as described in the main text in section 'Details of the apparatus'. Briefly, the cavity is injected with white light from a supercontinuum (SC) source. The output of the cavity is coupled into a spectrograph and dispersed onto a charge-coupled device (CCD) detector. The sample, which is placed inside the cavity is photoexcited with a diode laser. The timing of the opening/closing of the shutters, triggering of the magnetic field applied and the acquisition on the CCD is controlled with a digital delay generator (DDG). The controlling computer and data connections are not shown. **b)** Outline of the MFE experiment, as described in the main text in Section 'Details of the MFE experiments performed'. The photoexcitation (blue), whose onset marks the time $t = 0$, induces a change in the observed absorbance signal (ΔA). A subsequent burst of field-on field-off steps on magnetic field induces a secondary magnetic-field induced change in the observed ΔA signal ($\Delta\Delta$A). Two field-burst sequences used in this work are shown. The numbers next to the steps indicate (example) field values in G.

interface software. These include running statistics used to evaluate the stability of the cavity and control of the shutters and VND. The custom CCD interface, furthermore, does not clash with other running software, as it uses the device connection from the device interface above (unlike the Andor software).

- The various calibration programs are run only when said calibrations are needed. These programs are used for calibration of the VND angle, exposure time and, most importantly, the magnetic field. The field calibration procedure will be described in more detail in Section 5.5.
- The program performing the main BBCEAS experiment. This program calculates and sets up all of the timings on the CCD, DDG and the waveform generator, starts each acquisition, and collects and saves the data. It is also responsible for timing of the dark reoxidation period (from seconds to hours).

This partitioning of control software allows for very robust operation, as any crashes, bar those of LabVIEW or the device interface, do not shut down the entire experiment. Great care has been put into robustness of the device interface, and it has been found to run successfully for periods of, at the longest, about a month continuously.

Details of the MFE experiments performed

Historically, BBCEAS has been used as a purely steady-state technique.[131] The probe, as well as the pump light, would illuminate the sample continuously, and the magnetic field would be switched on and off to measure spectral change in in the light intensity. This would be then used with ground-state measurement acquired a-priori to calculate the MFE on the spectral absorbance of the sample. Albeit simple, this approach did not yield any insight into time evolution of either the sample absorbance, or its MFE. The sample was exposed to a lot of light which induced a lot of photodegradation. For more sensitive samples, or samples which could not be flown, this meant the system was not actually in a steady-state and would undergo degradation over the course of the experiments. Furthermore, any shifts in the alignment of the experiment, and the associated light loss would ruin the experiment as the initial reference would become invalid.

One of the aims of the work described in this chapter was to develop a new scheme for the the detection of the MFEs on the BBCEAS experiment, one which is more suitable for the protein samples samples investigated in Chapter 6. Multiple such schemes have been considered and attempted. The outline of two main ones used in this work can be seen in Figure 5.5b.

Field burst The main challenge with measurements of cryptochrome proteins is their slow reoxidation rate – on the timescale of minutes. This means that after only a brief exposure to sample to light, ca. 10 s, the sample needed to be be kept in the dark for ca. 20 min to return to ground state. All of the measurements of any field effects on the photoexcited states have to be performed within those 10 s. In order to maximise the use of that time, a magnetic field burst has been employed. Two burst schemes will be described here. Burst-2 will be the scheme used for most of this work, and is the one implemented currently. The burst-1 scheme has been described here as an important stepping stone in the development of the experimental procedure, and to highlight the experimental considerations which, ultimately, made it inviable for the measurement performed.

Burst-1 turns the magnetic field off and on in a random fashion. A selection of field values along the MARY curve is selected, shuffled, and interlaced with zeros. This sequence of field values is then applied to the sample within the photoexcitation period (Figure 5.5b). This is referred to as a single acquisition cycle, and it is repeated multiple times with the reoxidation time slotted in between. The resulting data is split into off-on-off steps and the *doubly-differential* absorbance $\Delta\Delta A$ for each of the field values is interpolated, akin to the description in Section 'Data slicing'. The $\Delta\Delta A$ can then be sorted according to the applied field value and averaged together for all the repetitions of the acquisition cycle. This method was found to be successful for the samples where the steady-state *of the magnetically sensitive component of the photocycle* is reached very quickly and no further evolution of the MFE occurs. The FMN/HEWL, other model systems as well as the *At*Cry WT protein (shown in Figure 6.2) measured were found to be just such systems. The data analysis procedure was, however, not straight forward, and the field controller used was not only limiting the field switching speed (to 200 ms step length), but was found to miss triggers and therefore fall out of step with the field acquisition. It also transpired that, if any evolution of the MFE on the timescale of the experiment would not be observed as the field steps of random magnitude would happen at random times though the acquisition cycle and then get averaged together. Hence, a different field-switching scheme has been implemented.

Burst-2 is a conceptually simpler scheme. The field value along the MARY curve is selected and then applied to the sample *repeatedly*, interlaced with zero field steps (burst-2 in Figure 5.5b). This is repeated for all the values along the MARY curve. This, in turn is repeated multiple times, split into off-on-off steps, and converted to $\Delta\Delta A$ and averaged as described in Section 'Data slicing'. The result can now be analysed as a function of the field index throughout the experiment, corresponding to the MFE evolution on the timescale of the field switching period. Since this method does not need a dedicated field controller to apply a shuffled field vector, but instead a single field value is used for each acquisition cycle, a waveform generator has been used to control the field burst. The field switching timing is now limited by the rise time of the coils (80 ms), and with carefully calibrated overcurrent can be as low as 50 ms period (25 ms step length). See Section 5.5 for more detail on field calibration and overcurrent. Furthermore, the waveform generator needs to be triggered only once (at the beginning of the burst) and so it will stay in step with the field switching as it relies only on internal timing for that purpose.[fy]

While allowing for time-resolved MFE measurements, the burst-2 scheme has some obvious *disadvantages* over burst-1. An acquisition of a N point MARY curve takes a single acquisition of N steps in burst-1, but N entire acquisition cycles in burst-2. If, however, the MFE is constant on the timescale of the experiment, the MARY curves from different times in burst-2 can indeed be averaged together to improve the S/N ratio. The time-dependence of the MFE on the 1 s timescale has indeed been observed for the *Er*Cry samples discussed in Chapter 6. The burst-2 scheme has been, therefore, used for all the measurements shown in this work, except for the earlier *At*Cry WT measurements shown in Figure 6.2, which has been acquired with the burst-1 sequence.

[fy] No detectable dephasing of internal clock references of different devices was found on the timescale of single acquisition. Nevertheless, all the devices have been synchronised to the DDG internal Rubidium clock timebase (less than 1 ns jitter on the timescale of the experiment).

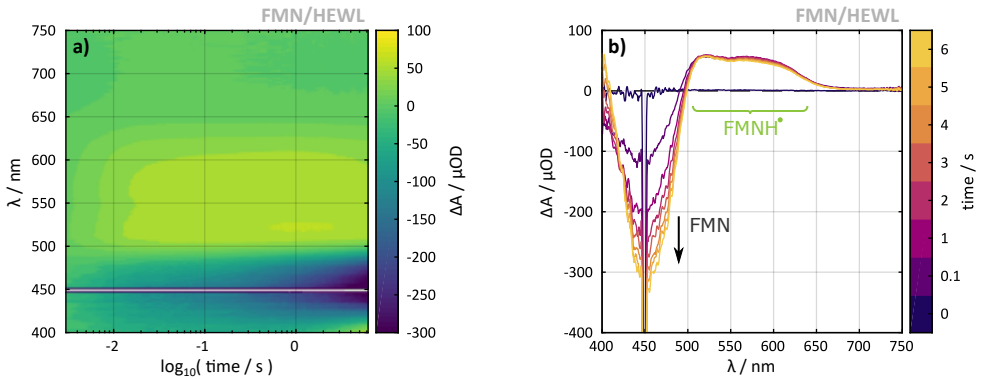


Figure 5.6: a) ΔA surface of FMN/HEWL. Average of 30 repetitions. The sample was flown through the sample cell with a infusion pump at 0.1 mLmin^{-1}. The pump power used was 5 mW. The sample was kept at 5 °C for comparison with cryptochrome samples measured. **b)** slices through the ΔA surface at selected times. The sharp feature at 450 nm is the scattered excitation light reaching the detector.

Performing the measurement in short acquisition sequences, as outline above has the added advantage of taking the measurement of the ground-state absorbance on each acquisition sequence. This helps correct for any alignment drifts and otherwise accumulated light loss due to samples' photodegradation and allows for monitoring of the immediate light-induced *differential* absorbance (ΔA) dynamics.

Data slicing

This section will outline the data analysis method – converting the raw measured intensity into the differential ΔA and doubly-differential ΔΔA absorbance change. The analysis of the burst-2 scheme will be discussed here. Burst-1 can be analysed analogously, but with an addition of a sorting step for the MFE steps.

The data is saved as 16-bit unsigned integer arrays in a hierarchical data format version 5 (HDF5).[132] The following procedure is repeated for each acquisition sequence (i.e. each field value) in each repeat of the experiment.

- Load the appropriate dataset – time × wavelength surface, its parameters and (timestamp, field value, VND angle, etc.)
- Subtract dark counts
- Scale the magnitude of the data by the (inverse) VND filter attenuation
- Convert raw intensity to relative intensity by either dividing by raw background measurement or a shape-corrected one. See Section 5.5 for details and reasoning behind the shape correction.
- Apply Equation 5.16 (inverse transfer function) to convert relative intensity to absolute transmittance (and therefore absorbance) of the sample
- Subtract the ground-state absorbance to convert the data to ΔA.
- The ΔA data is sliced into field-on and field-off segments akin to the way average fluorescence intensity has been analysed in Section 4.2.
- Fit the last ¼ (adjustable) of each pair of field-off steps with a line (not, as it were the case in the CLSM data, a quadratic) and use this as an interpolant for the intermediate field-on step. Hence get the ΔΔA.
- Optionally, the sliced background measurement – ΔΔK surface gets subtracted from the data. See Section 5.5 for more information and rationale behind the background correction step.

The result is time, wavelength and magnetic field resolved differential absorbance dataset. All of the above analysis is accompanied by the analysis of any associated uncertainties. Uncertainty in the linear fit of the sliced field-off steps is estimated by the residuals of the fit, rather than as described in Section 'Uncertainty of the optimum', as the latter was found to be too expensive (slow) for the number of fits needed to be performed per dataset.

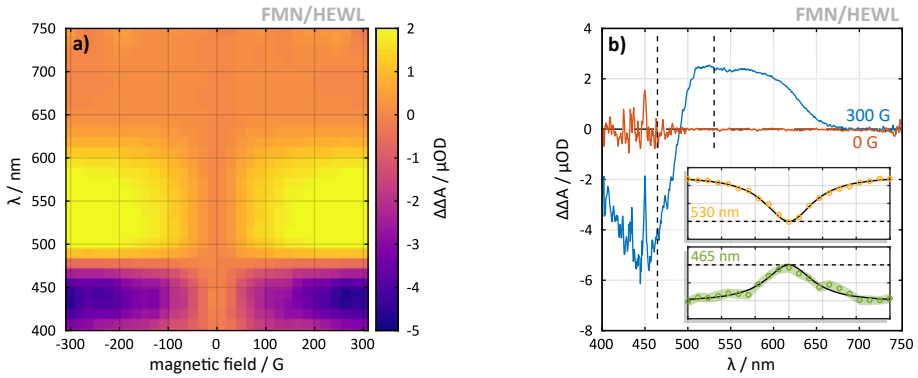


Figure 5.7: a) ΔΔA MARY surface of FMN/HEWL. 21 field points in 8 off-on-off steps. Average across 30 repetitions and, since no time-evolution was observed, across the 8 off-on-off steps. **b)** Slices through the ΔΔA at 0 and 300 G. The fitted MARY curve parameters for the $FMNH^{\bullet}$ region are: 4.15 ± 0.03 μOD and 84.0 ± 2.3 G, and for the FMN ground-state region: -2.54 ± 0.16 μOD and 86.9 ± 6.7 G.

Note that, nominally, absorbance is a unitless quantity. In this work, however, the units of optical density (OD) have been adopted. 1 OD corresponds to the absorbance of 1 – ten-fold extinction of light intensity upon passing through the absorber in question. 1 μOD is a corresponding extinction of light intensity by a approx. 1 part per million (ppm).

Differential absorbance An example differential absorbance (ΔA) dataset for FMN/HEWL system can be seen in Figure 5.6. The data shown is a time and wavelength resolved average surface for 30 repetitions of the experiment. The formation of the radical species ($FMNH^{\bullet}$) can be seen to occur on the timescale of 10 ms after the start of the illumination, while the ground-state (GS) bleach (FMN) grows slowly over the course of the experiment time (this is an example of light-induced dynamics not observable with historic methodology). The sensitivity of the experiment is nonlinear with the GS absorbance of the sample, as discussed in Section 5.1. This can be observed from the S/N ratio of the GS (400–500 nm) and the radical (500–650 nm) regions of the data (see Figure 5.6b). The radical region is mostly transparent, and only a small light-induced concentration of the $FMNH^{\bullet}$ forms. The transmittance of the probe beam in the radical region, when the sample is not photoexcited is, therefore, high and any changes in absorbance upon photoexcitation happen in the region of high gradient of the transfer function (see Figure 5.2). The GS region, on the other hand, is highly absorbing when not photoexcited. Any subsequent changes upon application of the photoexcitation light – depletion of the ground-state population of the absorber referred to as the *ground-state bleach* – happen in the region of a low gradient of the transfer function (see Figure 5.2). Any noise added atop the data will therefore see a low and high gradient in the radical and the GS region respectively, when going through the inverse transfer function during the data analysis process.

Doubly differential absorbance The ΔΔA MARY surface data for the dataset from Figure 5.6 can be seen in Figure 5.7a. 21 points along the MARY curve were chosen. The length of the burst was 8 steps. Since no time-dependence of the field effect was observed, the data shown is an average over all the burst steps. A clear positive and negative ΔΔA can be seen in the radical and GS region respectively. Since the species observed are the long-timescale accumulates of the MFE photocycle, this is indicative of a *triplet-born* radical pair, as expected for the FMN/HEWL system (see Section 1.5). Figure 5.7b shows the slice through the ΔΔA MARY surface at 300 and 0 G. The fitted MARY curves in the radical and GS region are shown in the insert. Both have been fitted with a single Lorentzian lineshape and, since the $B_{1/2}$ parameter (half width at half maximum) of both MARY curves match, one can reasonably assume that they originate from the same magnetosensitive mechanism (as expected for FMN/HEWL).

The comparison of the spectral shape of the ΔA and ΔΔA can yield interesting insights into the magnetosensitivity of the sample of interest. As mentioned, no time dependence of the ΔΔA was observed, while the

GS bleach in Figure 5.6b can be seen to keep growing. This suggests that the magnetosensitive part of the photocycle establishes quickly, most likely with the establishment of the steady state of $FMNH^{\bullet}$ radical i.e. 10 ms). There must be, therefore, an additional, not magnetosensitive, pathway of the depletion of the GS, which yields the increasing GS ΔA signal, while leaving the $\Delta\Delta$A one unchanged. Furthermore, one can see that the blue end of the $\Delta\Delta$A slice does not match the behaviour of the ΔA – $\Delta\Delta$A is negative while ΔA goes negative due to GS bleach and then again positive. This sign change could be attributed to eventual formation of an $FMN^{\bullet-}$ radical which has a strong absorption in this spectral region, and could therefore be part of the non-magnetosensitive pathway of depletion of the GS suggested above.[fz]

Fractional MFE The quantity often used for comparisons of the magnetic field effects, is the fractional (percentage) MFE. In the case of other absorption experiments, like transient absorption (TA) or CRDS, the %MFE is defined simply as $\frac{\Delta\Delta A}{\Delta A} \times 100$. There is, however, an issue when attempting to use the same definition in here: while $\Delta\Delta$A is known, what is the appropriate ΔA to reference it to? In the case of FMN/HEWL data above, one could simply divide the $\Delta\Delta$A and ΔA surfaces obtained. Since in the GS region $\Delta\Delta$A stays constant, and ΔA increases, the %MFE would seen to decrease over time, while staying constant in the radical region. Following the discussion above, one could, however, quite reasonably take the slice through the $\Delta\Delta$A surface at ca. 100 ms (since that is approximately when the steady-state of the magnetosensitive part of the photocycle gets established), and divide the entire $\Delta\Delta$A surface by this slice. This would yield %MFE which is constant in time. These two approaches clearly yield vastly different results and hence, whenever referring to the %MFE by means of this technique, it is crucial to specify: what is the %MFE defined *with respect to?* Due to this ambiguity, all the BBCEAS data in this work is shown as ΔA and $\Delta\Delta$A surfaces.

5.5 Experimental procedures

This section will outline details of specific experimental, and data analysis procedures used, as well as the rationale behind them.

Cavity alignment The gimbal mount of the sample cell, together with the adjusters of each of the mirrors provides the 6 degrees of freedom used to align the cavity. The coarse alignment is done by aligning back reflections through an iris placed between the filter wheel and M1. The fine alignment is done by maximising the light output of the cavity with respect to its 6 axes of freedom. The cavity is filled with water for alignment. An example of the raw light intensity measured at the output of the cavity can be seen in Figure 5.8b and Figure 5.9a. The light profile can be seen to extend through the range of most of the visible spectrum and be highly structured. This structure – or, at least, the major spikes seen in the figure – is attributed mostly to the spectral shape of the mirror reflectivity and the intensity profile of the SC source itself. This can be seen by making a comparison to Figure 5.4b where an example of spectrally resolved CEF can be seen. CEF is related to the mirror reflectivity R by the scattering losses T_S as $\text{CEF} = (1 - RT_S)$. Since T_S is not expected to show much spectral structure, most of the structure of the CEF can be attributed to the combined mirror reflectivity of the two mirrors.

CEF calibration The value of R_{eff} has been calibrated as described in Section 5.3. A broadband absorbing dye mixture of ~6 µM Acid Red 27 (Amaranth) ~8 µM Acid Green 25 and ~14 µM of Acid Green 1 (naphthol green B) was prepared.[ga] This mix has a broadband absorbance of approx. 0.4 OD over the 400-750nm range which can be easily and accurately measured on a benchtop UV-Vis spectrometer. The absorption spectra for these species as well as the dye mix can be seen in Figure 5.8. The dye mix was found to be stable over long time, easy to remove from the sample cell with distilled water (ultrapure MilliQ water, 18.2 MΩ,

[fz] The tyrosyl radicals also absorb strongly at 400 nm (not shown in Figure 1.11a). [133] The formation and eventual accumulation of a tyrosyl radical in the body of HEWL could be another explanation.

[ga] Note that the exact concentrations of individual dyes do not matter, as the overall absorbance of the mix will be measured anyway.

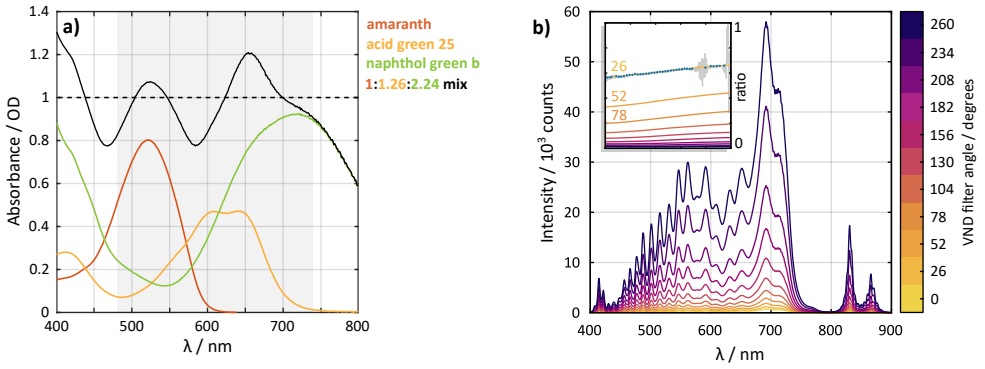


Figure 5.8: a) Absorption spectra of the dyes used for calibration of the BBCEAS experiment, and their linear combination in an approximate ratio used. Note that the calibration dye used is made up to ca. 0.4 mOD, and the sequentially diluted, as described in the main text, in order to obtain the calibration solutions. **b)** Intensity measurements used for the calibration of the VND filter position. The insert shows the corresponding intensity ratios. The points on the 26° line show the data selected for the global polynomial smoothing.

> 5 ppb total organic content) and caustic quartz cleaning solution (Hellma analytics, Hellmanex III). A series of dilutions of the stock dye mix is then prepared. As all of them come from the same stock, they all have the same, appropriately scaled, absorption spectrum. 10mL solutions of 1 part in 200, 100, 25, 7, 4, 3, 2 and 1 are prepared. Notably the low concentration regime is assumed, so that the effects of multiple interacting dye molecules changing the underlying absorption profile can be neglected.

The following measurements are made in order to calibrate the experiment:

- Background measurement (I_0) – light intensity transmitted through the cavity with the reference absorber and scattering losses inherent to the cavity present. In this work this is either water, water/glycerol mixture or the buffer solution for protein measurements.
- Dark measurement. This is the measurement of dark counts of the camera.[gb] It is smoothed by a global fit to a quadratic, and subtracted from all the other measurements made on the CCD camera before any other processing.
- Dye measurements (I) – the measurements of the light intensity output of the cavity for all the dilutions of the reference solution, starting with the most dilute.
- Dye reference. This is the measurement of the reference dye solution, made on a standard UV-Vis spectrometer, over the entire range of interest. Since it is made on a different detector to that used in the BBCEAS experiment, this measurement has to be interpolated onto the same wavelength axis, and cropped to the same range as the other calibration data.

The value of R_{eff} is then fitted across all the wavelength with the method described in Listing 5.3. The typical CEF is shown in Figure 5.4b. The calibration of R_{eff} is done before every measurement.

Background measurements The slicing of the data into steps and interpolation is a procedure akin to modulation/demodulation, as described in Section 2.1. The modem schemes are affected by the interference in the carrier band, since it does not get rejected by the LP filtering step of the demodulation. The same problem applies to the slicing described in Section 'Data slicing' – if the ΔΔA steps occur atop some other, time varying ΔA change, a false ΔΔA signal can be detected if the frequency components of the background ΔA fall within the field switching (modulation) frequency band. To mitigate this, a background measurement is taken at random points throughout the experiment. These background measurements are processed in *precisely the same way* as the absorbance data to give the differential and doubly background absorbance changes – ΔK and ΔΔK respectively. ΔΔK then gets subtracted from ΔΔA to remove the effects of

[gb] The dark counts are usually ca. 300 count in a 16-bit depth (max. 65 535 count) camera giving a noise floor of ca. 0.5 %

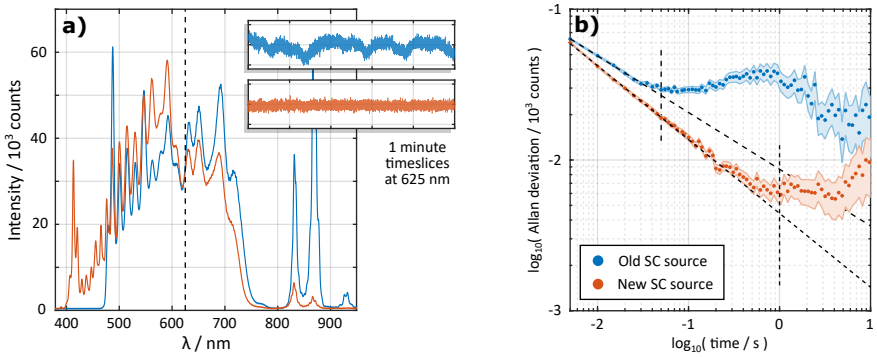


Figure 5.9: a) Intensity profile of the old and new supercontinuum source. 1 min timeslices at 625 nm are shown in the insert, showcasing the relative instability of the old SC source. The ordinate is displayed in the ±100 counts range. **b)** Allan deviation analysis of the time-slices shown in a). The initial gradients are fitted to a line on a log-scale. They evaluate to –0.38 and –0.49 for the old and new source respectively. The optimal stability times are 50 ms and 1 s, respectively (indicated with dashed vertical lines).

the background. A great care has been taken to 1) not introduce any spurious noise and 2) not distort the data and introduce false signals by means of this additional procedure.

VND calibration The variable neutral density filter position is read with each acquisition sequence, and used to correct the magnitude of the data as described in the previous section. The actual spectral optical density of the filter at each angle position is has been calibrated for by removing the cavity and measuring the SC continuum light intensity passing through the VND for different orientations (see Figure 5.8b). The transmittance of the filter is therefore calculated across the entire spectrum. Since the measurement is inherently noisy, especially in the regions where the measured light intensity is low, the region where the measurement of the optical density is valid is calculated based on the light transmittance of the filter, and smoothed by a global fit to a quadratic. The rotation stage on which the VND is mounted, is re-homed every time the device interface is turned on in order to ensure positioning accuracy. The calibration of the VND filter position has to be performed every time it, or any optics upstream from it are moved.

SC source change During the work performed on this experiment, the SC source was changed to a newer model. Figure 5.9a shows the intensity profiles for both sources. Note that the intensity profiles, although shown in the units of absolute counts, do not represent the actual brightness of the supercontinuum source. This is the case mostly because of a different setting of the VND filter.[gc] In practise the new SC source is about 2.5× brighter than the old one. This is known from specifications of the two devices, and has been confirmed by measuring the broadband radiant power at the output of the delivery fibre.[gd] The new source can also be seen to extend further into the blue end of the spectrum and, more importantly, show higher temporal stability. This can be seen in the insert where 1 min time slices of both datasets have been taken.

A more rigorous analysis of stability of a signal like that can be done with the aid of Allan deviation plot – the plot of the (square root) variance between length-n segments of data, as a function of n. The plot of Allan deviation of the two time-slices can be seen in Figure 5.9b. The Allan deviation plot can be interpreted as follows. The initial falling slope corresponds to fast noise – noise which quickly averages out when more than one sample is considered. The specific colour of the noise (slope of its power spectral density) will dictate exactly how well it averages and what is the exact initial slope of the Allan plot. Eventually, as the length of the segments considered increases, so does their variance since they begin to get more and more

[gc] There are other reasons. Firstly, the intensity profiles of the light passing *through the cavity* are shown. The details of alignment of the cavities for the two measurement were most certainly not the same, which definitely affected the output light intensity of the cavity. Secondly, much of the experimental setup was realigned and minor changes were introduced. The measured value of the intensity is a function of collection efficiency and light losses in the setup.

[gd] Broadband power can be measured with a uniformly absorbing thermal sensor.

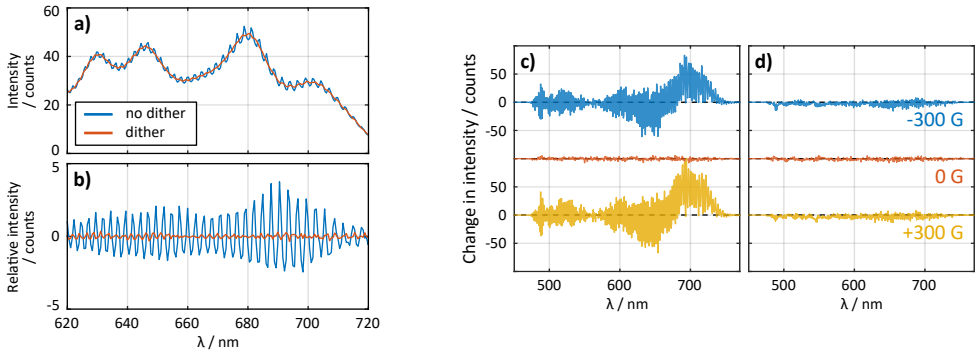


Figure 5.10: Influence of the M1 dither on the interference pattern in the transmitted light intensity. **a)** Close-up of the raw intensity spectrum measured. Note that the frequency of the observed comb-like pattern is changing with wavelength. **b)** The data from a), minus a baseline – 7p smoothing interpolant of the dithered data. **c) & d)** Basic field effect experiment performed on a cavity filled with water. c) is performed without dither, and d) with dither. M1 is dithered by ±4 µm at 2 kHz.

affected by the long-term drifts (slow noise).[ge] The optimal point between the two is usually taken as the optimal averaging time.[gf] This point can be seen to appear at much longer times, and lower noise floor for the new SC source, as expected based on the time-traces. Any periodic noise appears as a bump in the Allan deviation plot. It appears that the main contribution to the noise in the old SC source was just such a periodic (or pseudo-periodic) instability. Note that the error in the estimate of the Allan deviation increases at large segment length (due to there being fewer segments). The code used for calculation the Allan deviation, as well the associated uncertainty, can be found in Listing A.27.

M1 spoiler A pronounced comb-like pattern was observed for certain alignments of the cavity (see Figure 5.10a). The origin of this pattern is not clear, but it can most certainly be attributed to any combination of the following effects: 1) interference pattern due to differential pathways taken by the probe light through the cavity, 2) cavity mode mismatch, 3) formation of smaller sub-cavities between the mirrors and the reflective surface of the sample cell or 4) between the cell walls.[gg] Regardless of its origin, this pattern was found to be extremely sensitive to even the merest vibration of the elements of the cavity. This was disastrous for the MFE measurements, as the pattern was coherently influenced by the presence of the magnetic field (presumably inducing minute forces on any diamagnetic pieces of optics mounts). This is shown in Figure 5.10c. To mitigate the influence this effect has on the MFE measurement, the M1 was placed in a mount with piezo adjusters and dithered at 2 kHz by ±4 µm along the cavity axis. Since the dither is faster than the integrating time of the CCD detector (at least 1 ms), the comb-like structure averages over to a constant value. The effect of dithering on the transmitted light intensity can be seen in Figure 5.10b and on the MFE measurement in Figure 5.10d. The M1 ditherer is software-controlled and turned off during the dark reoxidation period.[gh]

Shape correction It was found that the spectral shape of the light intensity transmitted through the cavity can drift somewhat over the course of especially long experiments. This can happen due to thermal drifts affecting the nonlinear fibre of the SC source or minor shifts of the cavity itself. A section of the raw light intensity showcasing this drift over a 3 day measurement is shown in Figure 5.11a. As this, effectively, changes the shape of the reference measurement (I_0) over the course of the experiment, it will not be

[ge] Note the similarity between Allan deviation analysis and Fourier transforms – they both explore different components of the signals based on their timescales.

[gf] This is *not always* the optimal averaging time though. Sometimes, especially in differential measurements, the nature of the measurement in question has to be taken into account.

[gg] To test whether this indeed is an interference pattern, or whether is has something to do with the mode structure of the cavity, it was suggested that a very short coherence time light source could be used. The laser-driven light source (LDLS) has been proposed as a candidate. This would destroy the effect if it were to be due to an interface, but not if it were to be related to the cavity mode structure.

[gh] M1 ditherer is, also, turned on appropriately earlier than the start of the measurement in order to avoid interfering with it.

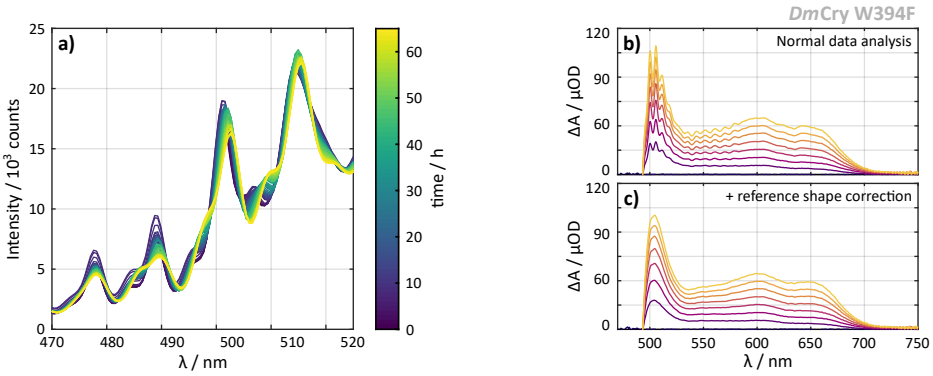


Figure 5.11: a) Demonstration of the intensity shape drift over the course of 3 days **b) & c)** Reference shape correction as applied to a dataset of *Dm*Cry W394F. b) shows ΔA slices as a result of normal data analysis described in Section 'Data slicing'. c) shows the result of the same procedure, augmented by the reference shape correction step.

corrected for by the background measurement on each acquisition, and will show up in the final ΔΔA data, as can be seen in Figure 5.11b. To (optionally) correct for this drift, the shape of the first GS $^{I}/_{I_0}$ was used as a reference, and the shape of all successive acquisitions was altered such that the *shape* (not the magnitude) matched that of the first acquisition GS $^{I}/_{I_0}$ (warped slightly with a 2nd order Lagrange polynomial). This was found to be very successful, as can be seen by comparing Figure 5.11b to 5.11c. Great care was taken not to alter the shape of the data unduly. Hence, this procedure was used only when necessary, and preceded by an analysis without any reference correction.

Burst calibration The following field burst calibration procedure has been developed:

- The 3D gaussmeter can be mounted in any orientation inside the magnetic field coils. The first step is therefore to determine the orientation of the magnetic field with respect to the internal coordinate system of the gaussmeter (equatorial x, y, and axial z). This is done by briefly applying a magnetic field of various magnitudes and measuring the corresponding readings for each. A line in 3-space, going through the origin, is fitted through these points and a unit vector along the line (in arbitrary direction) is established. The dot product of this unit vector with any measurement of the 3 field axes reported by the gaussmeter will be the *signed* magnitude of the applied magnetic field (sign indicating the direction).
- Secondly, the correspondence between the DC voltage and field is established. This is done in similar way to above, by applying a series of voltages and measuring the corresponding (signed) magnitude of the magnetic field. The table of values is not stored, but instead a *reverse polynomial* (2$^{\text{nd}}$ order) is fitted to the field vs. voltage data.[gi]
- Lastly, the magnetic field burst is calibrated. If the control voltage is switched in a instantaneous step, the power supply will immediately attempt to drive the updated value of current through the coils. Since the magnetic field coils have non-zero impedance, their response time (also referred to as the rise-time) will not be instantaneous. The rise-time of the coils used in the BBCEAS experiment has been measured at ca. 50 ms. Applying a slight overcurrent (increased electromotive force with which the power supply drives the current), allows for fast field switching (with step lengths of, at least 25 ms). The corresponding over (and under) control voltage is calibrated with a series of 5-step bursts. The magnitude of the field-on step for each control voltage is simply measured, while the magnitude of the undervoltage is adjusted with a bisection rootfinding algorithm until the field-off

[gi] When considering a measurement of a dependent (x) versus an independent (y) variable, a *forward* polynomial is that which when evaluated with the independent variable (x) approximated the dependent one (y). The reverse polynomial does the opposite – approximates the independent variable given the dependent one. In this case, the question of interest is: '*Given that this is the magnetic field I want to apply, what is the control voltage I need to use?*' rather than: '*If I apply this control voltage, what will the magnetic field be?*'.

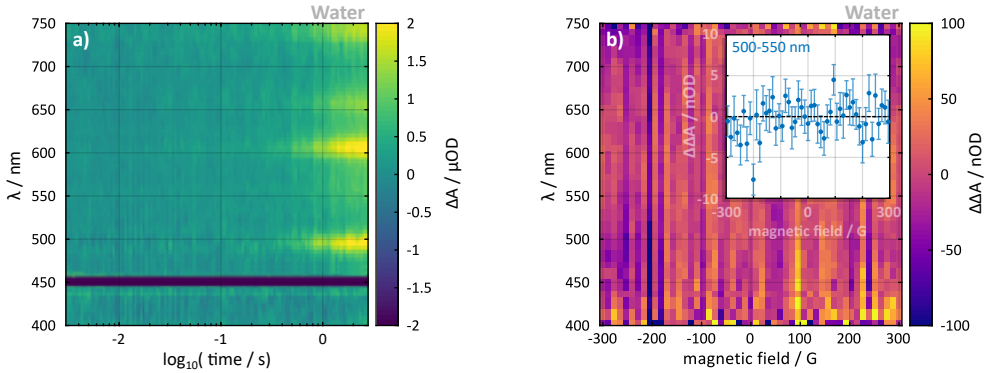


Figure 5.12: Blank measurement of water. **a)** ΔA surface – average of 40 repetitions. Cavity shifts (probably due to flow) induce small artefacts which can be seen at ca. 500 and 600 nm at times past 1 s. **b)** ΔΔA surface. 51 field points measured across the entire range. The insert shows an average response in the 500–550 nm range, since this will be the range used in later measurements in Chapter 6 as the $Trp^{\bullet}$ radical band.

step does indeed correspond to zero field. The magnitudes of both voltages are, similarly, fitted with reverse polynomials.

All the fitted polynomial coefficients, as well as the unit vector are stored in the calibration table and used by the experiment. Note that the calibration has to be done anew for each value of the field switching period. Overall, the calibration procedure is fully automatic, and takes approximately 10 min. An example acquisition of a field burst, along with an estimate of the high and low level can be seen in Figure B.1a.

The field offset controller – one used for small adjustments to correct for any hysteresis – was implemented as a simple integral controller (increase in control voltage proportional to the integral difference between the setpoint and the plant variable).[gj] This controller was not active all the time, but instead was turned on until a zero field was achieved every 2 min, or after each acquisition, whichever was faster. The field zeroing takes, on average, 2 s.

5.6 Blank measurements

In order to asses the limits of the experiment, and to make sure no artefacts are induced by the magnetic field, a series of blank measurements was performed.

Water blank The cavity is calibrated, as normal, and left with MilliQ water flowing through it at 1 $mLmin^{-1}$. The experiment is then performed as normal – with photoexcitation (3 mW), field switching and 'photorecovery' period of 20 s. The result is analysed in the same manner as a typical dataset. The ΔA surface can be seen in Figure 5.12a. Some spectral artefacts with magnitude of, at most, 1 μOD develop in the ΔA surface at ca. 1 s due to random cavity shifts. The 'bleach' at 450 nm is the scatter of the pump laser which is picked up by the detector. The ΔΔA surface shown in Figure 5.12b exhibits no spectral features, and a noise level of ca. 50 nOD. The 500–550 nm wavelength average, which has been shown in the insert of Figure 5.12b, is, again, featureless, with the detection limit of ±5 nOD. Note that the noise in the ΔΔA surface is distributed non-uniformly between the two field and wavelength dimension. This is because each field value is acquired separately, while all the wavelength points are acquired at the same time.[gk]

GOx blank The measurement of the water blank is useful but, potentially, unrealistic for the assessment of the detection limit on ΔΔA since there is no ΔA to begin with. For that reason, a flavoprotein glucose

[gj] The terms 'setpoint' (SP), 'plant variable' (PV) and 'manipulated variable' (MV) stem from control theory. PV is the property measured (e.g. magnetic field). SP is the desired value of the PV (in this case 0). MV is the property which can be manipulated by the controller (in this case the offset control voltage).

[gk] For that reason the wavelength averaging does not decrease the noise level by $N^{-1/2}$ as expected from averaging of independent noise samples, but by the effective number of degrees of freedom.

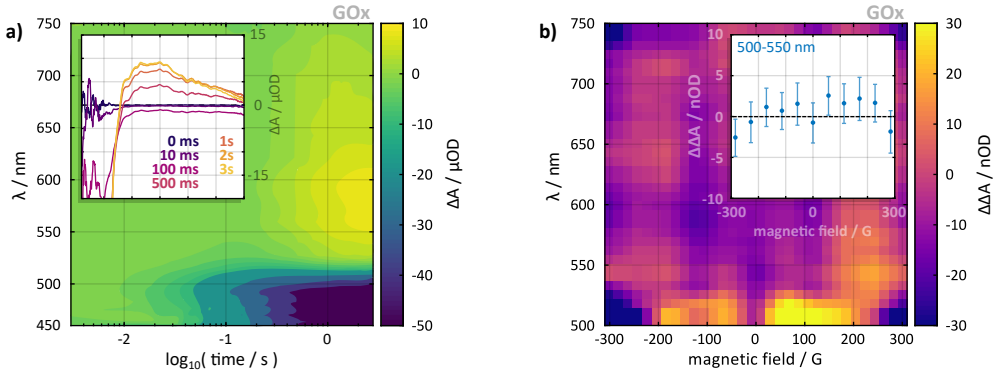


Figure 5.13: Blank measurement of GOx. **a)** ΔA surface – average of 221 repetitions. 10 s delay between each acquisition cycle. 45 mW pump power. The insert shows the wavelength slices through the ΔA surface at the indicated times. The abscissa of the insert runs over the same wavelength range as the main graph. **b)** ΔΔA surface. 11 field points measured. The insert shows an average response in the 500–550 nm range. The sample was prepared in a way analogous to that described in Chapter 6 – with GS absorbance of 200 mOD (17 μM), 20 % glycerol v/v, 100 μM pH 6 phosphate buffer, with no addition of potassium ferricyanide. The ΔΔA spectrum of this data at 300 G has also been plotted alongside other measurements in Figure 6.11 as a reference.

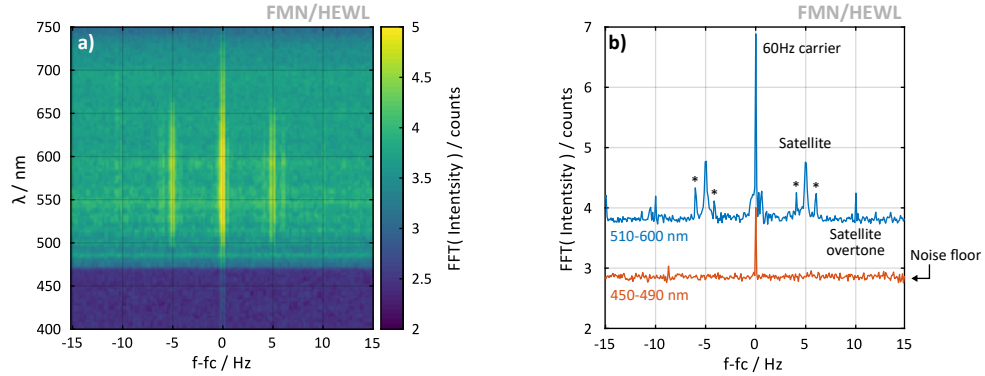


Figure 5.14: Satellite detected MFE on FMN/HEWL **a)** Wavelength-resolved Fourier transform of the cavity output fluorescence intensity, centered on the pump-laser optical chopper frequency (carrier frequency, $f_c = 60$ Hz). The ±5 Hz satellite peaks can be seen in the 500–650 nm range (FMNH$^\bullet$ region). The spectra have been acquired with 1 kHz sampling frequency (maximum framerate of the CCD) over the period of 10 s (0.1 Hz frequency resolution) **b)** Average frequency spectrum in two selected wavelength ranges. The noise floor in both ranges is different – the spectra have not been shifted along the ordinate. The 1 Hz satellite peak sidebands of unknown origin are indicated with ∗.

oxidase (GOx) from *Aspergillus niger* (black mould) was also measured. GOx binds a FAD cofactor, but has no residues which are able to support the ET chain and therefore support any efficient development of RP and magnetosensitivity.[134;135] The ΔA surface of GOx can be seen in Figure 5.13a. The GS bleach followed by the accumulation of broadly absorbing FADH$^\bullet$ species can be seen. The rate of accumulation of FADH$^\bullet$ is much lower than that of the ground state which suggests very inefficient ET from any neighbouring resides – as expected. Interestingly, FADH$^\bullet$ absorption is extremely broad – extending past the upper detection window (750 nm). This is, however, expected as the FAD in GOx sits in a very different binding pocket as compared to cryptochromes.[gl] The ΔΔA surface shows no MFE down to ca. 15 nOD, except for a feature in the 500–550 nm range. When averaged, however, this feature does fall within the ±5 nOD noise floor and therefore cannot be considered significant.

5.7 Satellite MFE

As discussed in Section 'Data slicing', the fractional MFE defined as ΔΔA/ΔA depends on ones choice of ΔA. A double modulation experiment – of both field and the pump laser – was performed in an attempt to

[gl] In cryptochromes the FAD is bound non-covalently and it is in a folded conformation. In GOx it is bound covalently, and in an extended configuration.

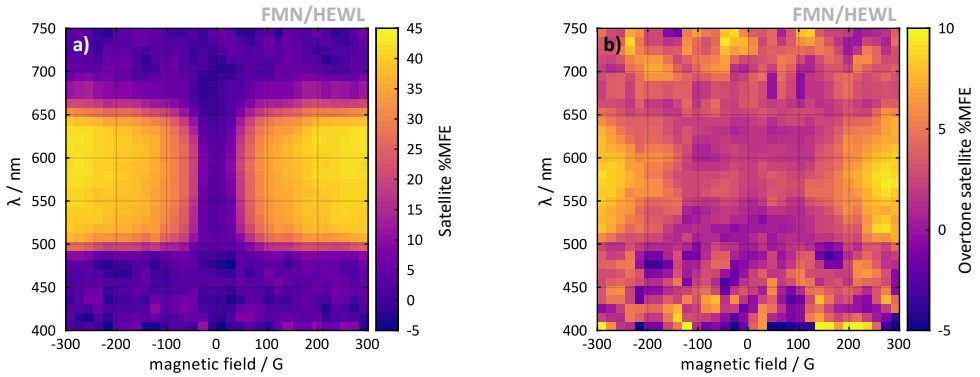


Figure 5.15: Satellite and overtone MARY surfaces of FMN/HEWL **a)** Wavelength-resolved ratio of the integrated –5 Hz satellite peak to the carrier peak. **b)** Wavelength-resolved ratio of the integrated –10 Hz overtone satellite peak to the carrier peak. The integral is centered on the peak of interest and runs over ±0.2 Hz range. Noise floor is subtracted from the spectra before integration.

understand the behaviour of ΔA, and its magnetosensitive fraction. The pump beam was modulated with an optical chopper (50 % duty cycle) at 60 Hz. The field was switched on and off, as before, at 200 ms period correspond to 5 Hz. Any field effect on the photoexcited state will, therefore appear as a 5 Hz amplitude modulation of the 60 Hz carrier. This frequency mixing will lead to a magnetosensitive satellite signal appearing around the carrier.

This experiment was attempted on the FMN/HEWL system. The frequency mixing was indeed detected, and can be seen in Figure 5.14. The carrier, which is proportional to the instantaneous ΔA at the timescale of 60 Hz (ca. 17 ms), can be seen across most of the spectrum. The satellite peak, which is proportional to the ΔΔA, appears only in the magnetosensitive region which corresponds to the $FMNH^{\bullet}$ radical (compare with Figure 5.6b). Due to the square-wave the modulation of the field (and therefore the harmonic distortion of the base 5 Hz frequency), both harmonic and overtone satellite peaks appear in the frequency spectrum (see Figure 5.14b).

The noise-floor of the spectra was estimated based on the bands between the signals (1–3 Hz, 7–9 Hz and 11–14 Hz) and subtracted from the frequency spectra. The integral (±0.2 Hz) of the peaks of interest was taken – the carrier, the satellite and the overtone satellite. The ratio of the satellite to carrier peaks can be considered as the ratio of ΔΔA to the magnetosensitive ΔA fraction on the timescale of the carrier (17 ms). The plot of such satellite-detected %MFE can be seen in Figure 5.15a. The shape of the MFE can be seen to match very well the $FMNH^{\bullet}$ region, but no GS MFE appears. This suggests that the GS MFE either develops on a longer timescale, or is not detectable with this technique. In either case, more research on it is needed, as it could provide an insight into the timescale of the development of the MFE by varying the carrier frequency. Note that this technique, as applied here, does not allow one to detect the *sign* of the MFE. This, however, could be achieved with a phase-sensitive detection akin to the lock-in methodology used in Chapter 3.

A similar plot for the overtone-satellite %MFE can be seen in Figure 5.15b. This signal also matches the $FMNH^{\bullet}$ region, but is much smaller and broader. This is, again, expected as the harmonic distortion of the signal leading to the overtone satellite peaks will be partially induced by the non-linearity of the MARY curve. If the phase-sensitive detection were to be employed, the relative sign of the positive and negative MFE lobes is expected to have opposite sign (akin to the ModMARY signal in Chapter 3).

A 1 Hz satellite on the magnetosensitive peaks has also been observed. Its exact origin is not known, but it is most likely related to the poor quality of the field controller used at the time of the measurement – the same field controller used for the burst-1 which was found to miss triggers. Such missed triggers could introduce additional frequency components into the field modulation. Alternatively, the plate of the chopper could

have a 1 Hz sideband if the chopper plate was to be warped. This, however would introduce the sideband on the carrier, not the MFE satellite peak.

This method has not been used in any other measurements performed in this thesis, but has been described here as both a potential avenue of future research, and to aid understanding of the experimental procedures developed in this chapter.

5.8 Outlook

As described, the SC source is, in fact, a pulsed light source used in pseudo-CW mode. An electro-optic switch could be employed to pulse-pick either the seed SC laser or its visible output pulses, and therefore use them for time-resolved measurement. Such a time- and wavelength-resolved measurement has been attempted in the past with use of 8 PMTs.[131] This has, however, resulted in poor light throughput and wavelength resolution. The PMTs did not have enough time-resolution to detect cavity ring-downs and therefore had to be operated in CW-probe and pulsed-pump mode. All of these shortfalls could, in principle, be resolved with a use of a streak camera.[gm] Since the spectrograph employed currently in the experiment has a free output port, this could be done with minimal disruptions to the current layout of the experiment, and with an option of switching between different modes.

The light loss at the sample cell is another major shortfall of the BBCEAS experiment. This could be addressed by employing a microcavity[136] and a microfluidic system.[137] This would also decrease the volume of the sample required and, potentially, allow for measuring multiple microdroplets which would eliminate the need for long reoxidation times, since each droplet could be treated as an individual average and then discarded. Depending on the exact experimental design, this approach would require a faster field burst sequence and would potentially be more experimentally challenging.

[gm] A streak camera is a device which allows for fast scan of the input light pulse in the spatial dimension (streak), usually in a manner coherent with an external trigger signal. The light is scanned across a spatially-resolved detector, and so the position of the light in the output can be mapped to the input time.

Chapter 6

Study of animal cryptochrome magnetosensitivity

This chapter will show and discuss results obtained on a variety of cryptochrome proteins by means of the BBCEAS technique discussed in Chapter 5. The investigated proteins along with their nomenclature are tabulated in Figure 6.1. *Er*Cry and *Cl*Cry proteins, as well as their mutants were expressed by Jingjing Xu from the group of Henrik Mouritsen from (Oldenburg, Germany). *Gg*Cry and *Tg*Cry proteins were expressed by Marco Bassetto from the group of Peter Hore (Oxford). *At*Cry was provided by the Defence Advanced Research Project Agency (DARPA) and *Dm*Cry was expressed in the lab of Erick Schleicher (University of Freiburg, Germany). GOx has been purchased from Sigma Aldrich (Cat. No. 9001-37-0).

Protein name	Binomial nomenclature name of the source organism	Common Name(s)
*At*Cry	*Arabidopsis thaliana*	thale cress
*Dm*Cry	*Drosophila melanogaster*	fruit fly
*Er*Cry	*Erithacus rubecula*	European robin
*Gg*Cry *	*Gallus gallus*	red junglefowl / chicken
*Cl*Cry	*Columba livia*	rock dove / pigeon
*Tg*Cry *	*Taeniopygia guttata*	zebra finch
GOx	*Aspergillus niger*	black mould

Figure 6.1: Nomenclature of the protein samples studied. The abbreviation used, Latin name and common name used to refer to it are given. Abbreviation by itself will be used to refer to the cryptochrome type in general. When a specific sample will be referred to, its name will be proceeded by 'WT' for the wildtype samples and by the mutation code with respect to the WT for any mutants. The mutation code is, for example, M213N – Methionine in position 213 has been mutated to Asparagine. *Tg*Cry and *Gg*Cry samples used (indicated with $*$) have been found to have had minor mutation in their C-terminus tail due to an error in sequencing. Only the *Tg*Cry WT sample data has been since reproduced and it was found to be *identical* to previous measurements. Hence, the data for these have been used here instead of the wildtype samples.

All the samples measured were prepared in a similar manner. The 'stock' protein samples in their expression medium (usually 20 % v/v glycerol) were either prepared immediately, or frozen in a cryogenic freezer (–80 °C) for later use. To prepare a sample to be measured, the absorption spectrum of the sample was measured with a benchtop UV-Vis spectrometer (Agilent, Cary 60). The quality of the sample was assessed qualitatively by the shape of the GS absorption spectrum, the presence of any scattering background and any $FADH^{\bullet}$ which may have been present. If needed, the protein solution was reoxidised for ca. 2 h with potassium ferricyanide ($K_3[Fe(CN)]_6$, Sigma Aldrich 244023), centrifuged for ca. 90 min and then carefully pipetted off the top to remove any scatterers which may be present (e.g. aggregated proteins). The examination by UV-Vis was then repeated. The sample was kept at low temperatures (below 5 °C) throughout. The concentration of the protein was estimated by measuring the maximum absorption of the GS peak. Based on previous studies, an absorption of 0.7 OD at the 450 nm peak corresponds to ca. 60 µM protein. The protein was then diluted/concentrated, and exchanged to a common buffer to maintain constant glycerol percentage and the ionic strength of the solution. Both of these factors can affect the exact tertiary structure of the protein and therefore the radical pair (RP) dynamics.

All the cryptochrome samples were prepared in 20 % v/v, 100 µM Tris (tris(hydroxymethyl)aminomethane, Sigma Aldrich 252859) buffer at pH 8, with an additive of 100 µM sodium chloride (NaCl, Sigma Aldrich S9888). The buffer solutions were prepared very carefully at room-temperature and their pH has been

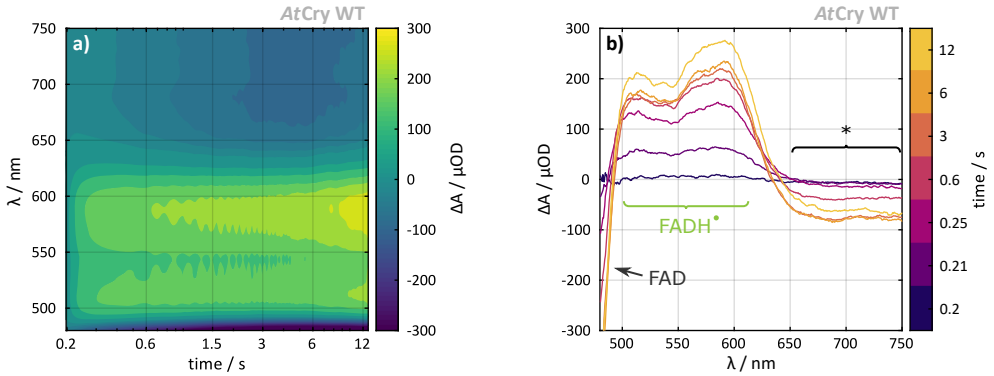


Figure 6.2: a) ΔA surface of *At*Cry WT. Average of 90 repetitions. The pump power used was 5 mW. The dark reoxidation time was 20 s. The vertical stripes in the ΔΔA signal are the MFE steps. **b)** Slices through the ΔA surface at the times indicated. The region of 'baseline shift' discussed in the main text is indicated with ∗.

carefully adjusted either with aq. 1 M HCl or 1 M NaOH (no more than 1 mL added per 200 mL of buffer prepared in a single batch).[gn] The *At*Cry sample was prepared at the ground-state absorbance of 200 mOD (17 μM), dilute *Dm*Cry W394F sample had a GS absorbance of 40 mOD (5 μM) and all the other samples were made up to 130 mOD (11 μM). The final volume of all the samples was 180–200 μL. GOx sample was prepared in 20 % glycerol v/v, 100 μM pH 6 phosphate buffer, with no addition of potassium ferricyanide.

All the samples, bar the dilute *Dm*Cry W394F sample measured in Section 6.1, were measured with 1 mM potassium ferricyanide to promote the reoxidation of the photoreduced species back to the ground state.[go] Potassium ferricyanide absorbs very strongly at 420 nm. The shoulder of this absorption does, therefore, obscure most of the ground-state absorption of the flavin. Hence, most of the data shown in this chapter will show only a small part of the GS ΔA and ΔΔA signals.

All the measurements were done with a static sample kept at 5 °C throughout the experiment. The sample cell was washed with the buffer solution before carefully pipetting in the sample and letting it rest for 30 min before starting the experiment. Unless otherwise indicated, the photoexcitation power used was 3 mW, and the samples were left in the dark for 20 min to reoxidise between acquisitions. With these conditions, the average experiment time was 1 week. Hence, the shape correction method described in Section 5.5 was used. The cavity enclosure was kept in dry air atmosphere for all the experiments. Since the sample was not covered, oxygen was still able to diffuse into the sample volume over the course of the experiment. No degassing of the solvent/sample prior to the measurements was done.

All MARY curve fits are performed using a single Lorentzian model, with prediction bounds estimated using a bootstrap method as discussed in Section 'Uncertainty of the optimum'. The exact fitting code used can be found in Listing A.28.

The X-ray structures and sequences of *At*Cry, *Dm*Cry, *Cl*Cry and GOx are known and published (PDB entries 4GU5, 1UC3, 6PU0 and 1CF3 respectively).[139–142] Only the sequences and homology models are presently available for *Er*Cry, *Tg*Cry and *Gg*Cry.

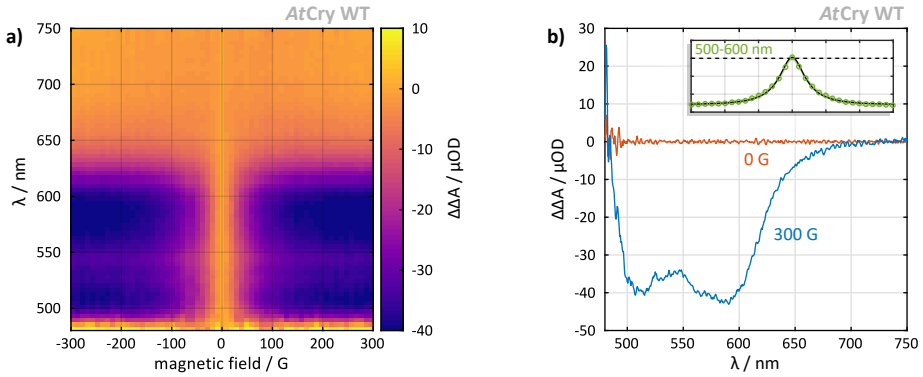


Figure 6.3: a) ΔΔA surface of *At*Cry WT. 61 field points were measured with the burst-1 sequence **b)** Slices through the ΔΔA surface at 0 and 300 G. The insert shows the average MARY curve in the 500 600 range ($FADH^{\bullet}$). The fitted MARY curve parameters are 39.20 ± 0.25 µOD and 46.7 ± 1.0 G.

6.1 Preliminary studies

***At*Cry** A plant cryptochrome 1 – *At*Cry has been measured as a preliminary study on an actual protein system. This is a widely-studied cryptochrome system which has been shown to have large singlet-born magnetic field effects, with the $FADH^{\bullet}$ radical being its long-timescale accumulant carrying the MFE. *At*Cry is therefore also shown as a contrast to other proteins measured, since it is the only one where the main RP termination pathway proceeded through $FADH^{\bullet}$ as a main intermediate.[143]

The ΔA surface for wild-type (WT) *At*Cry can be seen in Figure 6.2a, and the associated ΔΔA measurement in Figure 6.3. ΔA shows a clear and rapid formation of $FADH^{\bullet}$ (500–630 nm) alongside the GS bleach (<490 nm). The field effect originates from the $FADH^{\bullet}$ radical, as can be seen from the characteristic double-hump visible in both ΔA and ΔΔA spectra.

The time evolution of ΔA also shows a flat feature starting at 650 nm, and extending past the detection range (indicated with ∗ in Figure 6.2b). This feature is clearly coherent with the onset of photoexcitation, as any random baseline shifts would average out over the course of multiple acquisitions. If it were a feature originating from the absorption of chemical species, it would be hard to assign it to a known redox species of flavin. The only known species of flavin absorbing in that range is a molecular excited singlet ($^*F^1$) and a radical *cation* ($F^{\bullet+}$).[144] The former does not, in any circumstance, live long enough to accumulate, while the latter has never been observed in these systems. Another explanation, however, could indeed be a photoexcitation-coherent baseline shift due to, e.g. light-induced diffusion of a scatterer out of the probe volume. In either case, this feature has not been observed in the ΔΔA spectrum, suggesting it is not connected to the magnetosensitive branch of the photocycle.

Compare Figure 6.2a with Figure 5.6a – a similar measurement performed on FMN/HEWL. The neutral semiquinone radical species formation in FMN/HEWL is also rapid but levels off to a steady-state quickly. In *At*Cry, an initial establishment of a steady-state of $FADH^{\bullet}$ (on the timescale of c.a. 300 ms) is followed by a further slow increase. This suggests that a second pathway of formation of $FADH^{\bullet}$ exists, potentially in a misfolded fraction of the protein. This could be confirmed/revoked by the absence/presence of the associated change in ΔΔA signal respectively. This measurement was, however, the only one performed with the burst-1 sequence, as described in Section 5.4 and therefore no time-resolved MFE is available.

[gn] Note that the range of Tris buffer is ca. 8-9. In the past Tris solutions at lower pH was used, which defeats the purpose of using a buffer solution. An alternative buffering agent, similar to Tris, which buffers in the 6.3-7.3 range is the Bis-tris methane (BTM). The titration curves for both of these buffers have been shown in Figure B.9.

[go] The details of this process are not, in fact, known. It is almost certain, however, that the ferricyanide does not interact with the flavin moiety directly, as any such interaction would most likely induce a very strong spin-dephasing and therefore destroy the MFEs measured. It is proposed that the protein is reoxidised by the dissolved oxygen species present in solution. This forms free peroxide species (since it is a one-electron process) which, due to steric reasons, cannot reoxidise the flavin radicals.[138] The ferricyanide would, then, oxidise the protein indirectly by oxidising the peroxide species back to oxygen, and therefore increasing the free oxygen concentration in solution.

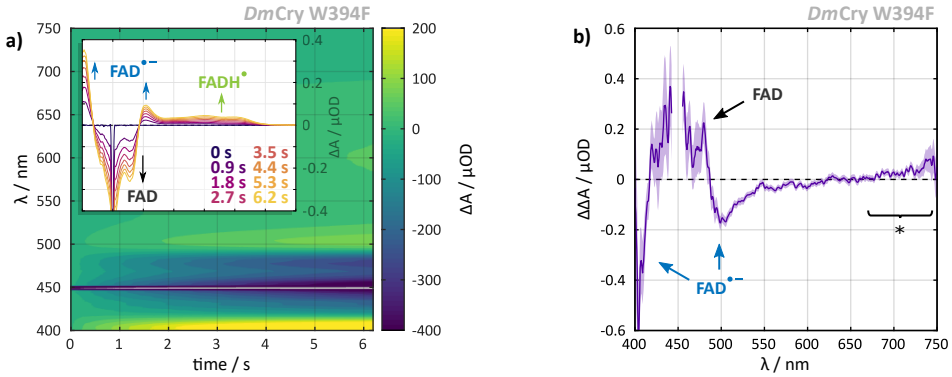


Figure 6.4: a) ΔA surface of dilute *Dm*Cry W394F, averaged across 66 experiment repeats. The insert shows slices through the ΔA surface at indicated times. The spectral features corresponding to different redox states of FAD are indicated. The details of the 500–750 nm region can be seen in Figure 5.11c. **b)** ΔΔA signal, averaged across 31 time points. The spectral features corresponding to different redox states of FAD are indicated. A slight artefact due to the fringe-pattern (see Section 5.5) can be seen in this data set, in the 670–750 nm range, and has been indicated with ∗.

Dilute *Dm*Cry Since the potassium ferricyanide used as a reoxidant in the measurements obscures the GS absorbance, the *At*Cry spectra are shown extending only down to 490 nm. This is not the full range of the SC source which extends all the way down to 400 nm and should therefore be able to observe the entire GS bleach region. To attempt to observe this, a dilute sample of a fruit-fly cryptochrome 1 (*Dm*Cry) with no reoxidant was measured. The protein used was not a wild-type, but a mutant. Like all animal cryptochromes studied here, *Dm*Cry WT has a tetrad of Trp residues, whereas *At*Cry has only a triad.[140] The Trp residues are usually referred to as Trp A, B, C and D for the 1st through 4th Trp in the ET chain, respectively. In the sample measured, the TrpD (W394) has been replaced with phenylalanine (Phe, F) – a structurally homologous, yet not ET active amino acid. This *increases* the MFE exhibited by the protein manyfold. Note that this data has been shown here as a preliminary measurement only. For more details and reasoning behind the W394F mutation see Section 6.2.

The ΔA surface of *Dm*Cry W394F over the entire 400–750 nm range of the SC source can be seen in Figure 6.4a. Due to the increased range of the probe, the formation of the entire GS bleach spectrum can indeed be seen. Since the sample is very dilute, the S/N ratio of the GS is very good and the characteristic shape of the GS FAD can be observed (compare with the reference spectrum in Figure 1.11a). The formation of $FAD^{\bullet-}$ and, on a slower timescale, $FADH^{\bullet}$ can also be observed. Notably, $FAD^{\bullet-}$ can be seen on both sided of the GS bleach region – at 500–515 nm range, as well as in the characteristic 400 nm region. The ΔΔA observed was very small (due to low concentration of the sample), but since it has shown no time evolution, it was averaged across time to increase the S/N ratio. This is shown in Figure 6.4b.

The RP termination reaction for *Dm*Cry is expected to involve the deprotonation of the radical centered on the tryptophan electron donor ($TrpH^{\bullet+} \rightarrow Trp^{\bullet}$), rather than protonation of the $FAD^{\bullet-}$ as it is the case in *At*Cry. The MFE can, therefore, be observed on the $FAD^{\bullet-}$ and the FAD ground state, but not on the $FADH^{\bullet}$ signal. This is in accordance with previous work. The $FADH^{\bullet}$ signal *can* however be observed in ΔA, which suggests its formation is part of the photocycle, but not the magnetosensitive branch. It it also likely, in the light of measurements from Section 6.2, that the protonation of the $FAD^{\bullet-}$ is a minor pathway (or a pathway of a minor fraction of the ensemble of the proteins with e.g. altered folding) which does indeed develop a field effect on $FADH^{\bullet}$ radical. Note also that, since no signal due to the $Trp^{\bullet}$ radical can be observed, its reduction back to the ground state must be rapid.

The dark reoxidation time used in this experiment was 1 h as opposed to the usual 20 min, since no iron ferricyanide was present which slowed the reoxidation rate, Furthermore, no MARY curve measurement was done, and, instead, the field values measured were just 0 and 300 G. This, combined with the very small

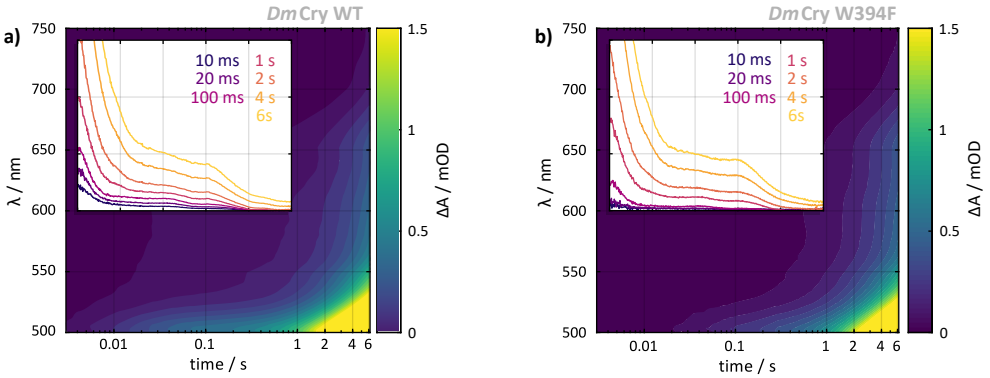


Figure 6.5: Comparison of ΔA surfaces of **a)** *Dm*Cry WT and **b)** *Dm*Cry W394F mutant. The slices at indicated times are shown in the inserts.

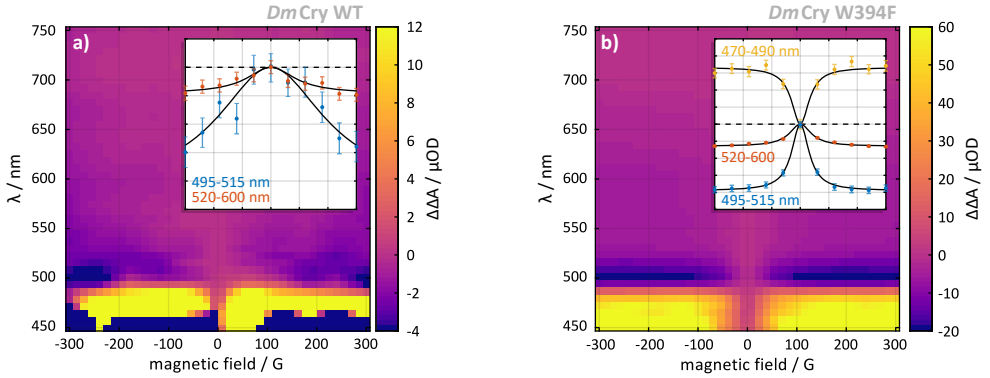


Figure 6.6: Comparison of ΔΔA surfaces of *Dm*Cry WT and *Dm*Cry W394F mutant. **a)** ΔΔA surface of *Dm*Cry WT. The average MARY curve in ranges 495–515 nm ($FAD^{\bullet-}$ shoulder) and 520–600 nm ($FADH^{\bullet}$) are shown in the insert. The fitted MARY curve parameters for the $FAD^{\bullet-}$ are: $-8.6^{+46}_{-1.5}$ μOD and 225^{+837}_{-47} G, and for the $FADH^{\bullet}$ they are: −1.8 ± 0.2 μOD and 101 ± 20.7 G. **b)** ΔΔA surface of *Dm*Cry W394F. The average MARY curve in ranges 470–490 nm (FAD ground state), 495–515 nm ($FAD^{\bullet-}$ shoulder) and 520–600 nm ($FADH^{\bullet}$) are shown in the insert. The fitted (single Lorentzian) MARY curve parameters for the FAD ground state are: 50.0 ± 1.45 μOD and 37.1 ± 5.86 G, for the $FAD^{\bullet-}$ they are: −19.5 ± 0.08 μOD and 36.6 ± 0.84 G, and for the $FADH^{\bullet}$ region they are: 6.5 ± 0.1 μOD and 46.1 ± 2.76 G.

magnitude of the ΔΔA obtained, justifies the use of reoxidant in other measurements.[gp] Further research into reoxidation chemistry of cryptochromes, as well as other non-optically active reoxidants would, however, be of great value as it would unlock a wider observable spectral range.[gq]

6.2 *Er*Cry and *Dm*Cry

The primary system of interest for this chapter is the Cry4 of the European robin (*Er*Cry). This protein has been recently expressed for the first time by Jingjing Xu, and it is of great interest to the avian magnetoreception. Indeed, many (most) behavioural experiments on migratory birds have been performed on European robins, and there is a wealth of evidence for robins' light-dependent magnetic sense.[14;145;146] The details of the magnetosensitivity of *Er*Cry will be investigated and compared with *Dm*Cry – a well known magnetosensitive cryptochrome from fruit fly.[1;38;42] The latter will be examined first, in order to establish a baseline.

Both samples have an ET chain consisting of four tryptophans. If the ET chain of the RP reaches the 4th Trp, the recombination probability of the RP decreases due to the radicals diffusing further apart in the protein body. This is, effectively, the case discussed in Section 1.4 (Figure 1.10a). The precise biological role of the

[gp] An experiment on this sample, but with a comparable number of averages and field points to those performed in Section 6.2 and 6.3 would take approx. 3 months as opposed to 1 week.

[gq] The key is to find an electron acceptor which does not stabilise the resulting charge by delocalisation, the latter usually leading to low lying electronic states and optical activity – undesirable absorbance. The biological cofactor – nicotinamide adenine dinucleotide (NAD), or simply nicotinamide itself could potentially fulfil that role.

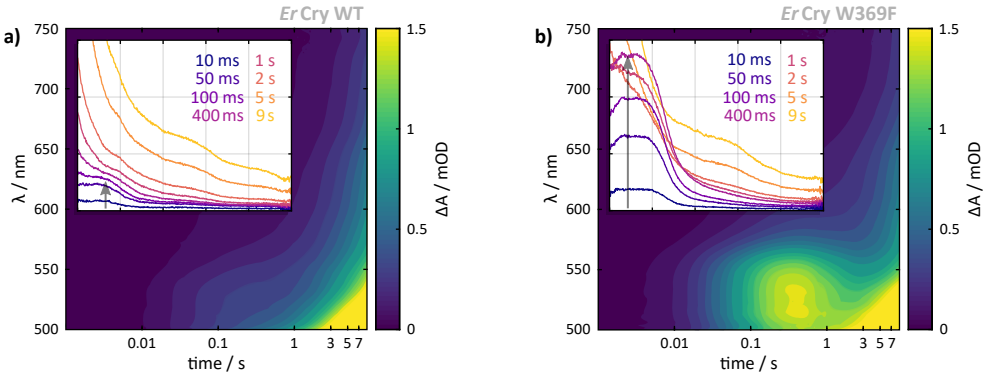


Figure 6.7: Comparison of ΔA surfaces of **a)** *Er*Cry WT and **b)** *Er*Cry W369F mutant. The slices at indicated times are shown in the inserts. The arrow indicate the initial rise of the absorptive feature at 500–550 nm attributed to $Trp^{\bullet}$ radical. Compare this figure with Figure 6.5.

4th Trp and the exact nature of the influence it has on the magnetosensitivity is not known. Two mutants in which TrpD has been replaced by phenylalanine (Phe, F) have therefore also been measured. As mentioned in Section 6.1, phenylalanine is structurally homologous to tryptophan, but it does not participate in ET.

***Dm*Cry** The ΔA surface of *Dm*Cry WT and *Dm*Cry W394F can be seen in Figure 6.5a and 6.5b respectively. The main feature of the ΔA surfaces is the steady accumulation of $FADH^{\bullet}$ radical in the 550–650 nm range and, what appears like, a red shoulder of the $FAD^{\bullet-}$ at wavelengths below 550 nm. A similar baseline behaviour will, in fact, be seen for all other cryptochrome samples measured. The corresponding ΔΔA surfaces can be seen in Figures 6.6a and 6.6b. The spectral shape of the MFE for both WT and mutant *Dm*Cry is the same as that observed for the dilute *Dm*Cry W394F sample in Figure 6.4 (albeit over a more limited wavelength range). In contrast to that measurement, however, the measurement performed here is resolved in the magnitude of the applied magnetic field. Hence, the MARY curves for different spectral regions can be seen in the inserts of Figures 6.6a and 6.6b.

The magnetic field effect of *Dm*Cry WT is, as expected, much smaller than *Dm*Cry W394F. The WT MARY curve can also be seen to be much broader than that of the mutant. This suggests an inherent change in the RP dynamics, since the width of the MARY curve is related to the hyperfine interactions which drive the spin mixing (see Figure 1.9b).[147] A simple explanation is that the hyperfine coupling the electron influenced by when residing on TrpC, rather than on TrpD, are vastly different. The spin density of the electron resides in the indole moiety of the tryptophan and its hyperfine interactions are strongly influenced by the secondary amine in the indole heterocycle.[148] If the amine in the 3rd and 4th tryptophans interact differently with their environments (either by hydrogen bonding or acid/base chemistry and therefore protonation/deprotonation), the spin density residing on the Trp could experience a different hyperfine coupling. Another possible explanation for varying widths of the MARY curve is that when TrpD is present, the electron spin density is either delocalised over both TrpC and TrpD, or the electron in hopping between the two Trp residues. In both cases, the result is a broadening of the MARY curve due to lifetime broadening of the ST spin sublevels.[31] This explanation of the $B_{1/2}$ difference between the mutants has the added advantage of explaining the exceptionally small MFE observed for the WT. When the electron hops between different Trp residues, random portion of the environment couples with the spin and induced its dephasing with respect to the other RP partner spin which is located only on a single moiety.

Note that the magnitude of the MFE, and therefore associated S/N ratio are relatively small for *Dm*Cry WT. The uncertainty of the associated fits is, therefore large. The differences in $B_{1/2}$ are, nevertheless, significant within the (bootstrapped) prediction bounds given. For the same reason no MARY curve for the ground state region (470–490 nm) of this dataset has been shown.

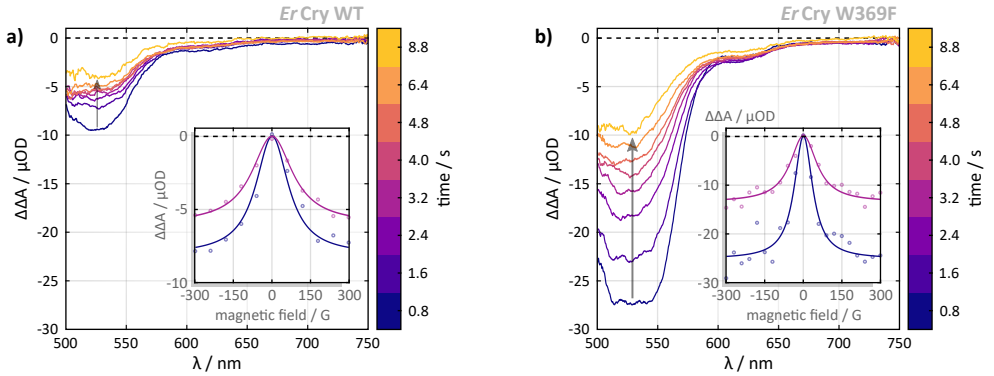


Figure 6.8: Comparison of ΔΔA surfaces of *Er*Cry WT and *Er*Cry W369F mutant. **a)** Wavelength slices of the ΔΔA surface of *Er*Cry WT at 300 G. The average MARY curves in the 500–550 nm range ($Trp^{\bullet}$) for early (575 ms) and late (2.58 s) times are shown in the insert. The fitted (single Lorentzian) MARY curve parameters for the early times are: 6.2 ± 0.3 μOD and 76.7 ± 8.7 G, and for the late times are: 4.6 ± 0.1 μOD and 95.9 ± 5.9 G. The full ΔΔA MARY surfaces can be seen in Figure B.10. **b)** Wavelength slices of the ΔΔA surface of *Er*Cry W369F at 300 G. The average MARY curves in the 500–550 nm range ($Trp^{\bullet}$) for early (575 ms) and late (2.58 s) times are shown in the insert. The fitted (single Lorentzian) MARY curve parameters for the early times are: 32.7 ± 1.0 μOD and 42.9 ± 4.5 G, and for the late times are: 17.6 ± 0.5 μOD and 62.3 ± 5.3 G. The full ΔΔA MARY surfaces can be seen in Figure B.11.

***Er*Cry** The wildtype and W369F (TrpD) mutants of *Er*Cry have been measured in the same manner as the *Dm*Cry proteins above, and their ΔA surfaces can be seen in Figure 6.7. The same background behaviour of $FADH^{\bullet}$ and $FAD^{\bullet-}$ can be seen in both cases. A striking difference can, however, be seen in the 500–550 nm region where a rise *and a subsequent fall* of an absorptive feature can be seen in both cases. Spectrally, this species has been assigned to a Trp neutral radical ($Trp^{\bullet}$, see Figure 1.11a). Due to its temporal behaviour, it must be an intermediate species or, at least, behave like an intermediate during the establishment of the photocycle (see, for example, the behaviour of species D in Figure 1.7b and of species C in Figure 1.10a). For a cryptochrome the formation and decay of a radical intermediate on this timescale has never been seen before.

Note that the red end of the ΔA surface (>700 nm) also rises in time, much more than the absorption of $FADH^{\bullet}$ alone can explain. Following the discussion on *At*Cry in Section 6.1, this is attributed to light-induced diffusion and/or photodegradation inducing an increase in scattering background. The exact nature of this process is not, however, known.

The ΔΔA spectra for both *Er*Cry proteins can be seen in Figure 6.8. In contrast to *Dm*Cry, the ΔΔA spectra of both *Er*Cry variants change in time and have therefore been displayed in a somewhat different style. The field effect can be seen to occur mostly on the $Trp^{\bullet}$ species, with a small but obvious shoulder in the 600–640 nm range which can, presumably, be attributed to the protonated Trp radical ($TrpH^{\bullet+}$, see Figure 1.11a). The $Trp^{\bullet}$ species the field effect is associated with is, presumably, the same transient species present in the ΔA spectra. Note that the timescale of the MFE measurement employed (100 ms field step length), and the delay of the start of the field burst, mean that the initial MFE point lies almost on the top of the $Trp^{\bullet}$ 'bump' which can be seen in ΔA. This means that the associated rise in the MFE cannot be seen in this data, confirming the association between the ΔA and ΔΔA. No MFE is observed on FAD species, however the $FAD^{\bullet-}$ cannot, in fact, be observed as it is masked by the $Trp^{\bullet}$ signal.

The relative change in ΔΔA between the *Er*Cry WT and *Er*Cry W369F is comparable to that observed in *Dm*Cry, with the field effect being larger for the W369F mutant. The magnitude of the *Er*Cry ΔΔA signals is much larger than any other animal cryptochrome measured, being matched only by the relative magnitude of the MFEs obtained on *At*Cry (see Figure 6.3). Note, however, that the concentration of *At*Cry WT is slightly larger than that of the samples measured here, and so it is possible that the actual fractional MFE is indeed larger for the *Er*Cry sample measured.

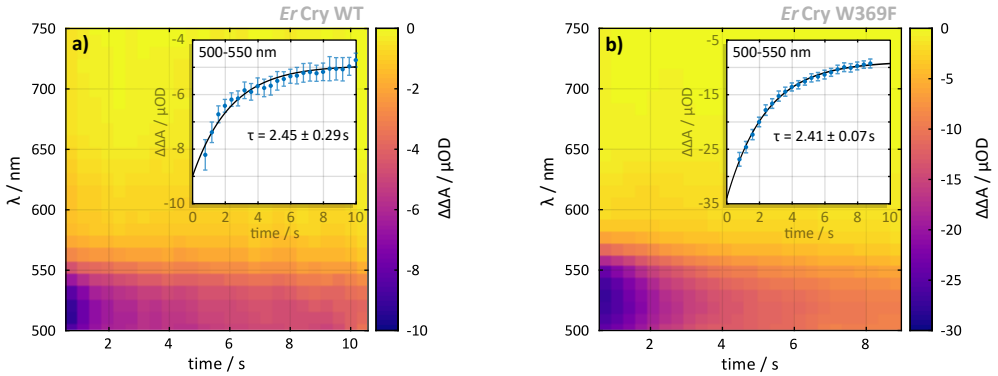


Figure 6.9: Comparison of ΔΔA time evolution of *Er*Cry WT and *Er*Cry W369F mutant. **a)** Time-resolved slice through the ΔΔA MARY of *Er*Cry WT at 300 G. The insert shows the time evolution of the average ΔΔA in the 500–550 nm range, fitted with a single exponential step model $(\alpha-\beta)\times\exp\left(-\frac{x}{\tau}\right)+\beta$ – an exponential growth from α to β with a time constant τ (see Listing A.15). The fitted parameters are $\tau = 2.40\pm0.29$ s, $\alpha = -9.7\pm0.4$ μOD and $\alpha = -4.3\pm0.1$ μOD. **b)** Time-resolved slice through the ΔΔA MARY of *Er*Cry W369F at 300 G. The insert shows the time evolution of the average ΔΔA in the 500–550 nm range, fitted with a single exponential step model. The fitted parameters are $\tau = 2.41\pm0.07$ s, $\alpha = -33.7\pm0.4$ μOD and $\alpha = -8.5\pm0.2$ μOD. In both cases the value of α can be thought of as the extrapolated value of MFE at time $t = 0$, and β as the asymptotic, steady-state MFE.

The MARY curves associated with the measurements can be seen in the inserts of Figures 6.8a and 6.8b. A similar trend to that discussed above in the context of *Dm*Cry can also be observed – the MARY curve for the WT is significantly broader than that on TrpD mutant. No significant time dependence of the width of the MARY curve was observed here although, in general, such effects have been seen, albeit on much faster timescales (ca. 1 μs).

A different view of the ΔΔA surface, and a single-exponential fit can be seen in Figure 6.9a and 6.9b. Although different in magnitude, the MFE can be seen to exhibit the same 'saturation' behaviour in both cases. The time dependence of the MFE can also be seen to be exactly the same – decaying to a constant level with an exponential time constant of ca. 2.4 s (see inserts in Figure 6.9). Importantly, it has been suggested that the temporal evolution of the MFE could be related to its accumulation over multiple field steps. To test this hypothesis, a similar experiment was performed with the field burst delayed by ca. 4 s. The observed ΔΔA 'begun' at a lower value and continued with the same time constant. This indicates that the observed change in the MFE is a manifestation of an underlying chemical change, rather than an accumulation of the effect of multiple field steps.[gr]

There are multiple aspects of the magnetosensitivity of *Er*Cry which are unique to it. The MFE on an accumulated $Trp^{\bullet}$ species has not been previously observed in any other cryptochrome. The same goes for the transient ΔA signal of $Trp^{\bullet}$ observed. No strong explanation of this behaviour of *Er*Cry can be given here, as much more research and modelling needs to be done to understand the details of underlying ET, RP interactions and chemical kinetics. A working theory, however, is based on differential redox and proton transfer rates for the 3rd and 4th Trp. Since TrpD is exposed to the outside of the protein, it can interact with the solvent much more readily than the 3rd Trp, and therefore will most likely become reduced back to the GS much faster than the 3rd Trp. It is, therefore, suggested that the MFE observed in both WT and W369F mutant originates, in fact, from the 3rd Trp. The rate of redox and proton transfer of the 3rd Trp are, then, what gives rise to its accumulation and eventual settlement to a steady-state. This is accompanied by the evolution of the associated MFE. At present, no explanation as to the exact role of the 4th Trp in WT can be provided. It is possible TrpD is somehow involved in the signal transduction, but since very little is known about that aspect of magnetosensitivity of cryptochromes, no further suggestions can be given.

[gr] A similar experiment with 400 ms long field steps has also yielded a correspondingly longer set of field steps in ΔΔA. No MFE dynamics within a single step was observed, although this could have been prohibited by the S/N ratio of a single acquisition.

Since the Cry4 molecules, which are suggested to be responsible for the magnetoreception, are located in the retina,[gs] it is reasonable to assume the perception of the magnetic field is connected to the visual cortex. The magnetic clues are required by the bird only periodically, not permanently. The saturation behaviour of the MFE could, therefore, be an evolutionary adaptation which decreases the number of signals needed to be processed by the visual cortex and not obscure the visual information.

A transient absorption experiment (e.g. TA, CRDS) takes a major portion of the population of the ground state, pumps it to the excited state and monitors the evolution of the species as they make their way back to the ground state through transient intermediates. The timescales of the processes probed in such a measurement are necessarily short. Furthermore, such measurement can often fail at measuring either long-lived intermediates, or pathways with small quantum yield which may, nevertheless, accumulate on longer timescales. Conversely, in BBCEAS, and in other in steady-state and light-induced dynamics experiments, such pathways are in fact the main observable. This means that such experiments ought to be considered complimentary – providing orthogonal pieces of information about the system of interest. For a much more in-depth analysis of the *Er*Cry magnetosensitivity, the data presented here has to considered together with such measurements. This has been done in the publication resulting, in part, from this work, [16] and will be explored further in the future.

[gs] The exact localisation of the Cry4 within the cells is not known. It it theorised that, to fulfil their role as a magnetoreceptor and a *compass*, they ought to be immobilised in a cellular membrane.

6.3 Avian mutants

The final study of this work concentrates on the influence of the protein environment on the development of the magnetic field effects observed on *Er*Cry proteins.

The European robin is part of the passerine branch of the phylogenic tree (order Passeriformes, perching songbirds). Most passerines are migratory, but there is a sub-branch which has lost its magnetic sense. A careful examination of the differences between the members of this branch shows a high correlation between the evolutionary mutation of certain aminoacids and the loss of magnetic sense. Specifically, migratory species seem to posses a cysteine (C) in position 317 and a lysine (K) in position 320. Both of these amino acids are in positions which allow interaction with the RP species – C317 can interact with the terminal ET chain Trp (TrpD), while K320 is in the vicinity of the FAD. The importance of these residues has been confirmed by preliminary computer simulations (personal communication with Marco Bassetto).

Selected cryptochrome sequences and their mutants have been measured and are shown here. The cryptochromes selected are all Cry4 variants from various non-migratory species: zebra finch (non-migratory passerine), pigeon and chicken.[gt] The latter two are not passerines, but their sequence is very similar to that of *Er*Cry, also bar the position 317 and 320 amino acids. The summary of the WT and mutant proteins measured can be seen in Figure 6.10. All the ΔA and ΔΔA surfaces, as well as the associated MARY curves have been shown in the figures indicated.

Protein name	Amino acid at position 317	Amino acid at position 320	|ΔΔA| / μOD	$B_{1/2}$ / G	Figure
*Er*Cry WT	Cys (C)	Lys (K)	6.2 ± 0.3 4.6 ± 0.1	76.7 ± 8.7 95.9 ± 5.9	6.7a, 6.8a B.10
*Gg*Cry WT	Arg (R)	Glu (E)	1.2 ± 0.1	121 ± 12	6.12
*Cl*Cry WT	His (H)	Glu (E)	0.8 ± 0.1	154 ± 17	6.13
*Tg*Cry WT	Ser (S)	Glu (E)	0.8 ± 0.1	92 ± 10	6.14
*Gg*Cry R317C	Cys (C)	Glu (E)	2.2 ± 0.1	122 ± 6	6.15
*Gg*Cry R317C+E320K	Cys (C)	Lys (K)	0.2 ± 0.1	51 ± 18	6.17
*Tg*Cry S317C	Cys (C)	Glu (E)	0.5 ± 0.1	60 ± 6.5	6.16

Figure 6.10: Comparison of wildtype robin, chicken, pigeon and zebra finch cryptochromes and their selected mutants. The amino acids in position 317 and 320, as well as the magnitude of ΔΔA signal and $B_{1/2}$ in the 500–550 nm range are shown. The two entries for *Er*Cry WT are for early (0.7 s) and late times (10.3 s). The last column gives references to the figures showing the corresponding datasets.

The spectrotemporal behaviour of all the samples measured was found to be very similar, and akin to that described for *Er*Cry WT. The presence of the $Trp^{\bullet}$ band in the 500–550 nm $Trp^{\bullet}$ range has been seen for all the proteins to greater of lesser degree (see e.g. Figure 6.13). The ΔΔA for all the samples was found to also be centered around the $Trp^{\bullet}$ region. The comparison of the ΔΔA spectra of all of the proteins at 300 G can be seen in Figure 6.11. Note that, since all the samples have the same concentration of protein and have been measured in the same conditions (and the same excitation power as well as pump-probe overlap), they can be compared with one another directly. The measurements shown for *Tg*Cry WT have been reproduced two more times on samples from different batches, prepared independently of one another and in a freshly aligned and calibrated experiment each time. This has yielded identical results each time, and suggests that the methodology used (e.g. the sample preparation and cavity alignment) is capable of delivering reproducible results and resolving the differences between the ΔΔA signals shown.

Firstly, none of the cryptochromes of non-migratory origin shows any time dependence of the MFE, unlike *Er*Cry WT (not to mention *Er*Cry W369F), although in certain cases it is possible that it simply cannot be

[gt] Chickens and pigeons are not passerines. They belong to the order of Galliformes and Columbiformes (doves) respectively.

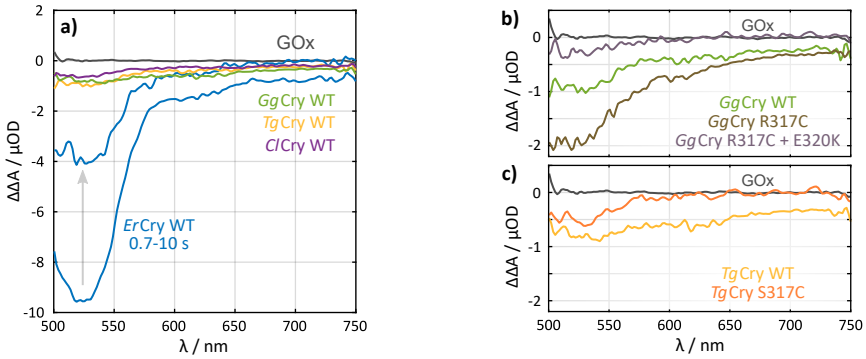


Figure 6.11: a) Comparison of ΔΔA spectra of *Er*Cry WT, *Gg*Cry WT, *Cl*Cry WT and *Tg*Cry WT. ΔΔA spectrum of GOx is shown as a blank, for reference. Since ΔΔA of *Er*Cry WT changes in time, its spectra are shown at early (700 ms) and late (10 s) times. **b)** Comparison of ΔΔA spectra of *Gg*Cry WT, *Gg*Cry R317C and *Gg*Cry R317C+E320K. ΔΔA spectrum of GOx is shown as a blank. **c)** Comparison of ΔΔA spectra of *Tg*Cry WT with *Tg*Cry S317C. ΔΔA spectrum of GOx is shown as a blank.

resolved due to very small magnitude of the ΔΔA signal associated. Secondly, the MFEs observed for these species are much smaller than the MFE of *Er*Cry WT. These differences highlight the importance and uniqueness of the effects observed on *Er*Cry, but by themselves give little insight into the role of the 317 and 320 residues. Figure 6.11b compares the three *Gg*Cry proteins measured, and Figure 6.11c does the same for the two *Tg*Cry proteins.

The mutation of the Arginine 317 in *Gg*Cry doubles the observed field effect. This suggests this amino-acid is indeed strongly associated with magnetosensitivity. Its role is, however, most certainly non-trivial as a similar mutation in *Tg*Cry decreases the MFE. The *Tg*Cry WT protein also shows a broad red shoulder which, as mentioned, has been reproduced on a couple of occasions. This shoulder disappears when the Serine 317 is mutated into a Cysteine, and the field effect becomes centered only on the $Trp^{\bullet}$ band. Since residue 317 is located close to TrpD, which already has been shown to have an enormous effect on the MFE, it is suggested that the C317 interacts with the terminal Trp. The nature and effect of this interaction are presently under investigation. It is possible the Cysteine stabilises the spin density on TrpD (or the motion of the residue) and therefore increases the observed field effect (by increasing the lifetime of the RP). In the light of the fact that removal of TrpD increased the observed field effect manyfold, it is more likely that the Cysteine pulls the TrpD residue away for the main Trp tetrad and therefore increases the fraction of spin population remaining of TrpC and/or decreasing the rate of TrpC→TrpD hopping (if that is indeed what occurs in these proteins).

Finally, a double mutation of *Gg*Cry was attempted in order to bring *Gg*Cry in line with *Er*Cry (see Figure 6.10). This was not, however, successful and the resulting field effect can be seen to drop to almost nothing. This could be due to misfolding caused by the double mutation, but could also be a real effect, and indicate an interplay between the 320 position amino acid which neighbours the FAD and some other residue. Also, the mutation performed: Glu→Lys is one of the most strenuous for the protein, since glutamic acid is one of the most net-negative residues while lysine one of the most net-positive ones. This, however, does not distinguish between the misfolding and FAD interaction and the major change in the coulombic environment could either change the redox and ET properties of FAD, or induce misfolding of the protein.

The measurements presented here are the beginning of the extensive study of such non-tryptophan interactions of the cryptochrome residues with the RP. The field effects of the mutated proteins shown above also give insight into just how fine of a balance of many factors is achieved by the evolution of *Er*Cry proteins.

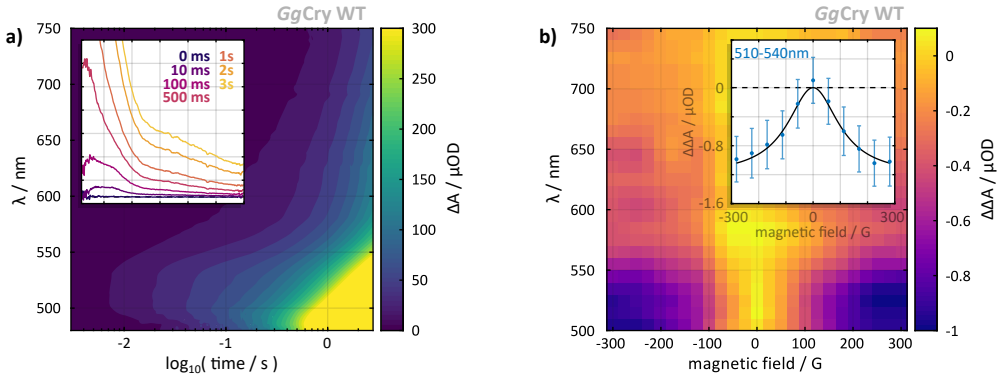


Figure 6.12: a) ΔA surface of *Gg*Cry WT, averaged across 15 experiment repeats. The wavelength-slices at selected times are shown in the insert. **b)** ΔΔA surface of *Gg*Cry WT, averaged across 12 time points. The insert shown the average MARY curve in 500–550 nm range. The fitted (single Lorentzian) MARY curve parameters are: 1150 ± 65 nOD and 121.0 ± 11.7 G

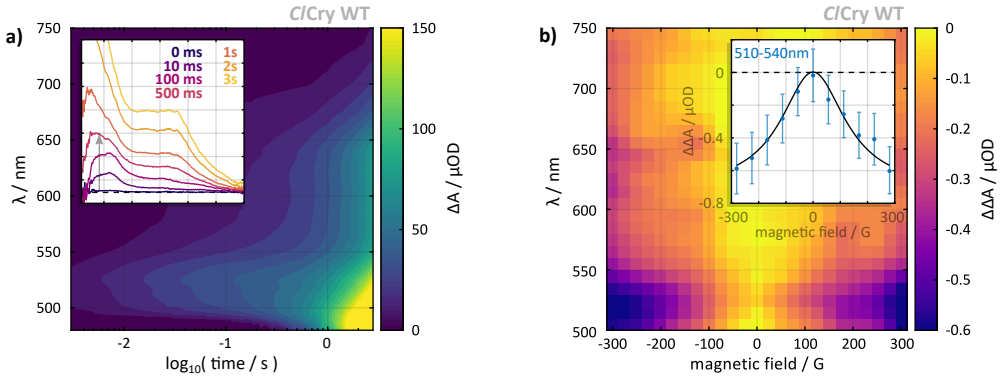


Figure 6.13: a) ΔA surface of *Cl*Cry WT, averaged across 17 experiment repeats. The wavelength-slices at selected times are shown in the insert. The arrow indicates fast initial formation of $Trp^{\bullet}$ radical. **b)** ΔΔA surface of *Cl*Cry WT, averaged across 12 time points. The insert shown the average MARY curve in 500–550 nm range. The fitted (single Lorentzian) MARY curve parameters are: 770 ± 63 nOD and 153.8 ± 16.7 G.

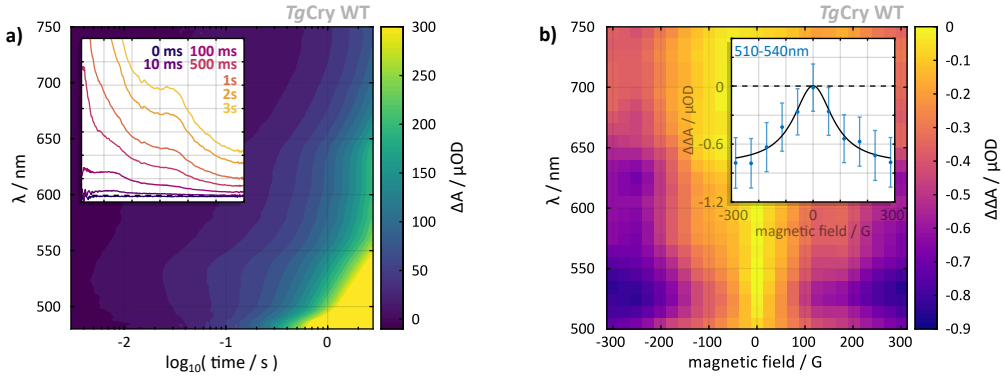


Figure 6.14: a) ΔA surface of *Tg*Cry WT, averaged across 28 experiment repeats. The wavelength-slices at selected times are shown in the insert. **b)** ΔΔA surface of *Tg*Cry WT, averaged across 12 time points. The insert shown the average MARY curve in 500–550 nm range. The fitted (single Lorentzian) MARY curve parameters are: 813 ± 36 nOD and $91.6^{+11.6}_{-7.7}$ G

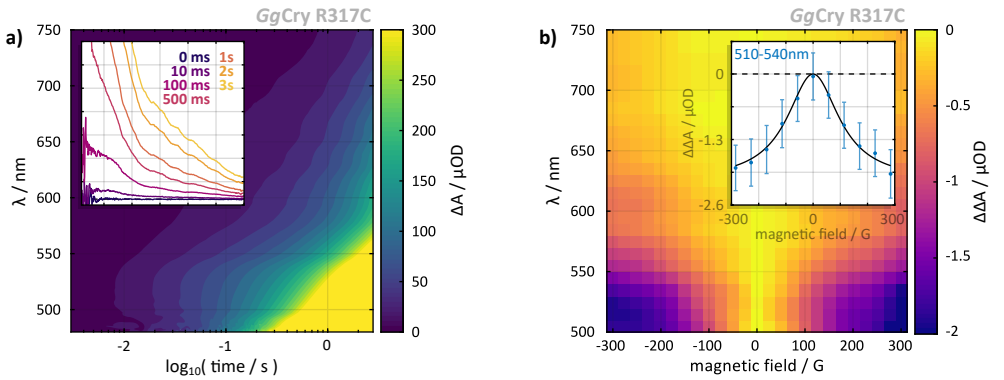


Figure 6.15: a) ΔA surface of *Gg*Cry R317C, averaged across 33 experiment repeats. The wavelength-slices at selected times are shown in the insert. **b)** ΔΔA surface of *Gg*Cry R317C, averaged across 12 time points. The insert shown the average MARY curve in 500–550 nm range. The fitted (single Lorentzian) MARY curve parameters are: 2200 ± 72 nOD and 121.8 ± 6.4 G

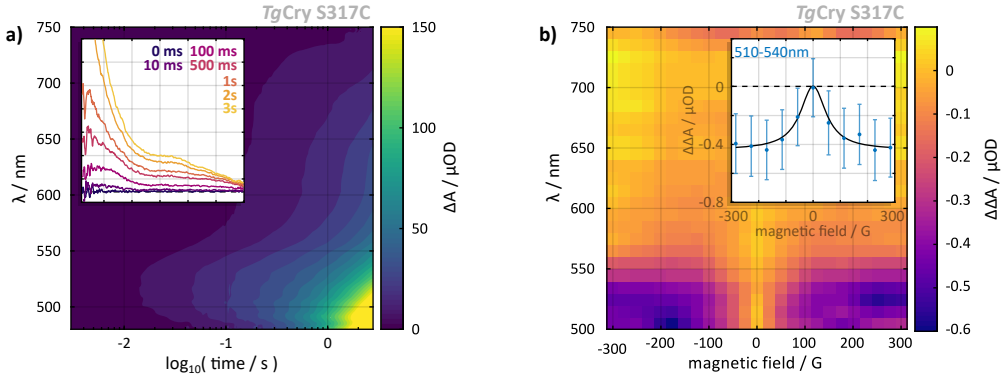


Figure 6.16: a) ΔA surface of *Tg*Cry S317C, averaged across 21 experiment repeats. The wavelength-slices at selected times are shown in the insert. **b)** ΔΔA surface of *Tg*Cry S317C, averaged across 12 time points. The insert shown the average MARY curve in 500–550 nm range. The fitted (single Lorentzian) MARY curve parameters are: 496 ± 15 nOD and 59.6 ± 6.5 G

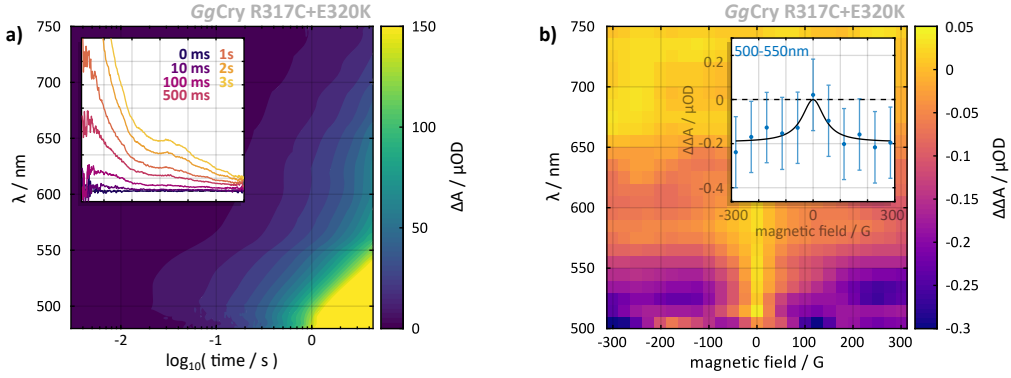


Figure 6.17: a) ΔA surface of *Gg*Cry R317C+E320K, averaged across 16 experiment repeats. The wavelength-slices at selected times are shown in the insert. **b)** ΔΔA surface of *Gg*Cry R317C+E320K, averaged across 20 time points. The insert shown the average MARY curve in 500–550 nm range. The fitted (single Lorentzian) MARY curve parameters are: 194 ± 21 nOD and 51.2 ± 18.4 G

6.4 Conclusions and outlook

Measurements of cryptochrome proteins presented in this chapter are amongst the first performed on the *Er*Cry proteins, as well as many other non-migratory birds and their mutants. The comparison between *Er*Cry and *Dm*Cry reveals never-before seen photoinduced dynamics and slow-timescale evolution of the (substantial) MFE observed on *Er*Cry. The accumulated magnetosensitive species is shown to be different from *Dm*Cry, and it is assigned to the $Trp^{\bullet}$ radical.[gu] The comparison between *Er*Cry WT and other non-migratory avian mutants shows a stark difference – the magnetosensitive species in non-migratory cryptochromes also appears as the $Trp^{\bullet}$ radical, but the magnitude of the MFE does not match that of the *Er*Cry WT. Further study of various mutants suggests the influence of the neighbouring residues in the position 317 and 320 on both the magnitude and $B_{1/2}$ value of the resultant MARY curve. Following experiments have also suggested that the appearance of the magnetosensitive shoulder at 560–650 nm is significant (compare, for example, Figure 6.14b and 6.15b). Although its behaviour is not yet interpreted, it is preliminarily assigned to the $TrpH^{\bullet+}$ radical.[gv]

In order to elucidate the mechanism of cryptochrome magnetosensitivity, further investigation is clearly required. The following aspects are suggested for the most-immediate future investigation:

- The MFE of the avian cryptochromes is clearly exhibited in the accumulated $Trp^{\bullet}$ (and $TrpH^{\bullet+}$) species. What current experiments cannot, however, elucidate is which tryptophan residue (or combination of a few) along the ET chain is responsible for magnetosensitivity. TrpA and TrpB are too short-lived to support a magnetic sense, and therefore the discrimination has to be made between TrpC and TrpD. Other experiments and simulations could allow that. An EPR technique – out of phase electron spin echo envelope modulation (OOP-ESEEM) allows the measurement of the distance between the constituent spins in the RP.[gw] The exact nature of the tryptophan could be elucidated by selective mutations or its environment and/or three-pulse correlation absorption spectroscopy, [150] as well as with a non-photonic approach like the hydrogen ↔ deuterium exchange mass spectroscopy (HDx-MS). [151]

- It is clearly important to understand the long-timescale behaviour of the cryptochromes under illumination, but it is also important to understand the link between the short (TA), intermediate (BBCEAS) and long (irradiation) experiments. For example; long timescale photoreduction shows $FADH^{\bullet}$ while the short timescale ΔA indicates $Trp^{\bullet}$ radicals and the blue shoulder which is potentially $FAD^{\bullet-}$. The two experiments are not in contradiction (BBCEAS looks at short timescale and ca. 10 µOD absorptions while irradiation in UV-Vis looks at minute timescale and absorptions on the order of 1 OD) but the chemistry in between is currently hard to probe. The BBCEAS experiment could be run to longer timescales, but it lacks the dynamic range of concentrations to be able to detect the initial formation of $Trp^{\bullet}$ as well as long-timescale accumulation of FAD. The dynamic range could be improved at the price of sensitivity – by using lower reflectivity mirrors. Another approach would be to devise a mixed irradiation and pump/probe experiment, and therefore attempt to probe the fast dynamics of the species slowly accumulated under photoillumination. An equivalent experiment could also be done with slow irradiation and fast time-resolved fluorescence spectroscopy.[gx]

[gu] The role of the $Trp^{\bullet}$ radical and its redox has also been seen in the circadian clock activity of the cryptochromes. [149]

[gv] The shoulder could also be assigned to the blue-shifted $FADH^{\bullet}$ radical (see Figure 1.11). The exact position and shape of the accumulated $FADH^{\bullet}$ radical spectrum is, however, known from long-timescale irradiation experiments and its shape *does not* match the shoulder observed in the BBCEAS data.

[gw] The nuclear interactions around each of the tryptophans are, also, expected to be different. This could be probed with an electron nuclear double resonance (ENDOR) experiment.

[gx] This mixed experiment would not work with time-correlated single-photon counting (slow irradiation vs. millions of single events required for TCSPC), but would most-likely work with the application of fluorescence upconversion spectroscopy since it operates on a pump/probe principle. [152]

- Further experiments (EPR, BBCEAS, TA, etc.) and simulations (homology modelling, molecular dynamics, QM-MM modelling of electron transfer) on *Gg*Cry, *Tg*Cry and *Cl*Cry are needed to understand the difference between the migratory and non-migratory avian cryptochromes.
- The kinetics of photo-accumulation of *Er*Cry WT and *Er*Cry W369F, as well and those of the resulting field effects are never-before seen, and therefore need to be carefully simulated and understood properly in the context of the possible reaction kinetics, and its implications. Some preliminary work in this direction has already been done.[153]
- A large limitation of the current experiments on proteins are the sub-optimal sample conditions. The main areas of improvement are the reoxidising agent (one which does not absorb in the visible region, unlike potassium ferricyanide presently used) and the exact excitation wavelength used (see discussion in Section 4.5). In general very little is known about the mechanisms of cryptochrome redox and photodegradation. Even though not central to the magnetic sense, they are clearly important to understand in order to make further progress in the cryptochrome MFE research.

BACK MATTER

Acknowledgements

For their vicarious contributions to this work, I would like to thank:

- My parents, Jola and Peter Konowalczyk, thank you for supporting me in whoever I want to be and whatever I want to do. I know it is not easy sometimes. Thank you for giving me education, and thank you for trying to show me wisdom, regardless of how much I sometimes resist.
- My supervisor, Christiane Timmel, thank you for putting up with me for me for so many years. More seriously though, thank you for supporting my both in my scientific development and in my personal life. It is the most wonderful thing to know (not 'think', 'know'!) that you have someone on your side, indisputably, regardless of how much you may happen to argue about one particular aspect of reality or another.
- My grandmother, Ada Konowalczyk, thank you for always being my friend and guardian.
- Stuart and Peter, thank you for being my co-supervisors, as well as the perfect compliment to the good cop / bad cop dynamic with Chris. Of course, we all know who the good cop is…
- All of Timmel group, past and present, for keeping up the amazing and friendly spirit throughout all my time with you. You are all mad, in the best way possible.
- Dr Beata Paściak, for sparking my love of chemistry. I do not know where would I be, nor what would I do now if it were not for you. Thank you, from the bottom of my heart.
- Dr Peter McKerchar and Mr Ian Pattison, for keeping the momentum of my passion for chemistry going, and helping me get to Oxford.
- Prof. Nathan Kutz, Prof. Nick Trefethen, Prof. Gilbert Strang and late Prof. Patrick Winston, for being my (remote) tutors of computer science and numerical mathematics.
- Bea Prentiss, Dan Claff, Julia McDonnell, Jordan Wheeler, Christoph Weis, Sam Hutton, Kaja Molińska and all the other Komrades and everyone from the Oxford Salsa scene who has brought me so much joy. Al Centro!
- Alex O'Bryan-Tear, Ziggy Yinda, Jess Ockendon, Martin Newman, Gregg Bayes-Brown, Leila Al-Izzi, Francesca Elizabeth, Inigo Lapwood, Ariane Laurent-Smith, Yanni Van der Pant, Blaine Willian Rogers, David Leon, Emily Aemas, Tristan Thomas Serjeant and everyone else from the HoD community. You have been nothing but wonderful to me in ten-thousand-and-one little ways you might not even realise. Delphi love!
- Ross Atkins, for being my friend and a fiend.
- Timothy Powell, John Adams, Phillip Hurst, Neville Baker, Robert Howcroft and Kevin Valentine from PTCL electronic workshops. I am honestly *happy* when some part of the experiment breaks, because I get to go and chat with you guys. First rule of the ping-pong table is that you do not speak of the ping-pong table…
- Martin Wall and Paul Mitchell, for always smiling.
- Adrian Potter, Amy Welmers, for being amazing coaches and friends.
- Adam Fineberg and Chrissie Potter, for being the source of ceaseless cheer and good spirit, both in your own special ways.
- Naomi Bull, for being my dance partner and a friend. You have made me feel awesome and invincible.
- Iris Geens, for being a supportive and wonderful friend, even if we sometimes end up not talking for months while living, effectively, 10 min away from one another. Also, for trusting me to dance with you and making me spit my lungs out on HOFAH. *It was so good!*.

- Olivia FVE and Jonathan Storey, for working with me on the ModMARY experiment, proof-reading the publication manuscript and being a fantastic student and mentor, respectively.
- Tilo Zollitch, for the late-night games of Magic, freezing our digits off in the sea in winter, solving riddles and endless scientific curiosity.
- Captain Hanna Theaker, for great boat trips and inducting me into your crew.
- Jessica Walton, for working with me on the BBCEAS experiment and putting up with my ramblings.
- Christian Kerpal, for being a fantastic rubber duck with really interesting stories about wine.
- Sabine Richert, for laughing at my stupid jokes and being a wonderful friend and mentor.
- Victoire Déjean, for being a great friend and an amazing discussion partner. Also for late nights spent working and being completely 'high' on tiredness together.
- Lauren Jarocha, for keeping my spirit up when I really needed it. Also, for showing me how to align the Nd:YAG even though you knew I was never going to use it.
- Matthew Golesworthy, for keeping the wet lab clean and blowing up his own experiment (multiple times) to distract the bosses from my problems. As agreed, we will never tell anyone…
- Kevin Henbest, for being the force of good and kindness in this world.
- Gabriel Moise, for teaching me more about quantum mechanics than *almost* anyone else. Also, for understanding my childish excitement upon discovering an esoteric way of calculating something.
- Laura Raczyńska, for being my longest-lasting friend and always sticking by me.
- Jamie Gravell, for the ATS trips, and late nights spent editing. You are a worthy *young blood* of the Timmel group. Drinks?
- Alice Bowen, for forgiving me for throwing out her party appliances, and even taking me to dinner and cheering me up when I was anxious.
- Hugh Lindley, for having breakfasts with me. Also, for keeping their mess strictly confined to their own room, and in general being a great flatmate and an amazing friend.
- Julian Hatt, for being a friend.
- Jonathan Burton, for being a wonderful tutor and teaching me to do 'reasonable chemical things'. This really stuck with me more than you can imagine. Apparently it works in physical chemistry too. Who would have thought?
- Maya Heine, Tommy Pitcher, Ruby Lyons and Andrew Markham, for the Brexit stickers, origami paper, amazing dancing and actually using git, respectively.
- John McGrady, for telling me that, for him 'the only way to really understand something is to program it'. For me too, as it turns out.
- Marco Bassetto, for teaching me more than I ever knew I would be interested in about biochemistry.
- Grant Ritchie and Ilya Kuprov, for being the most efficient and kindest examiners I could have imagined.[gy]

I would like to add, that in the above list I have referred to many people as my friends. The word 'friend' means different things to different people and, even within the context of the above list it means something else when referring to one, than when referring to another. Suffice to say, in no context have I used this word lightly. Thank you all.

[gy] This acknowledgement has been added *after* the viva, during the (minor) corrections.

Bibliography

[1] Dean M.W. Sheppard, Jing Li, Kevin B. Henbest, Simon R.T. Neil, Kiminori Maeda, Jonathan Storey, Erik Schleicher, Till Biskup, Ryan Rodriguez, Stefan Weber, Peter J. Hore, Christiane R. Timmel, and Stuart R. MacKenzie. Millitesla magnetic field effects on the photocycle of an animal cryptochrome. *Scientific Reports*, 7:1–7, 2017. | ii, 145

[2] Daniel R. Kattnig, Emrys W. Evans, Victoire Déjean, Charlotte A. Dodson, Mark I. Wallace, Stuart R. Mackenzie, Christiane R Timmel, and Hore P. J. Chemical amplification of magnetic field effects relevant to avian magnetoreception. *Nat Chem*, 8(4):384–391, 2016. | 43, 96, 106, 109

[3] Connie X. Wang, Isaac A. Hilburn, Daw An Wu, Yuki Mizuhara, Christopher P. Cousté, Jacob N.H. Abrahams, Sam E. Bernstein, Ayumu Matani, Shinsuke Shimojo, and Joseph L. Kirschvink. Transduction of the geomagnetic field as evidenced from alpha-band activity in the human brain. *eNeuro*, 6(2), 2019. | ii, 95

[4] Leah R. Weiss, Sam L. Bayliss, Felix Kraffert, Karl J. Thorley, John E. Anthony, Robert Bittl, Richard H. Friend, Akshay Rao, Neil C. Greenham, and Jan Behrends. Strongly exchange-coupled triplet pairs in an organic semiconductor. *Nature Physics*, 13(2):176–181, 2017. | ii

[5] Hermann Kraus, Sebastian Bange, Felix Frunder, Ullrich Scherf, Christoph Boehme, and John M. Lupton. Visualizing the radical-pair mechanism of molecular magnetic field effects by magnetic resonance induced electrofluorescence to electrophosphorescence interconversion. *Physical Review B*, 95(24), 2017. | ii

[6] Chris Sampson, Robert H. Keens, and Daniel R. Kattnig. On the magnetosensitivity of lipid peroxidation: Two-: versus three-radical dynamics. *Physical Chemistry Chemical Physics*, 21(25):13526–13538, 2019. | ii, 95

[7] Christian Kerpal, Sabine Richert, Jonathan G. Storey, Smitha Pillai, Paul A. Liddell, Devens Gust, Stuart R. Mackenzie, P. J. Hore, and Christiane R. Timmel. Chemical compass behaviour at microtesla magnetic fields strengthens the radical pair hypothesis of avian magnetoreception. *Nature Communications*, 10(1):1–7, 2019. | ii, 26

[8] Kiminori Maeda, Kevin B. Henbest, Filippo Cintolesi, Ilya Kuprov, Christopher T. Rodgers, Paul A. Liddell, Devens Gust, Christiane R. Timmel, and Peter J. Hore. Chemical compass model of avian magnetoreception. *Nature*, 453(7193):387–390, 2008. | ii

[9] Charles B. Grissom. Magnetic field effects in biology: A survey of possible mechanisms with emphasis on radical-pair recombination. *Chemical Reviews*, 95(1):3–24, 1995. | ii

[10] Christopher T. Rodgers and Peter J. Hore. Chemical magnetoreception in birds: The radical pair mechanism. *Proceedings of the National Academy of Sciences of the United States of America*, 106(2):353–360, 2009. | ii

[11] Vladimir N. Binhi and Frank S. Prato. Rotations of macromolecules affect nonspecific biological responses to magnetic fields. *Scientific Reports*, 8(1):1–11, 2018. | ii

[12] Marcin Konowalczyk, Olivia Van Der Elst, Jonathan Storey, and Christiane R. Timmel. Development of lock-in based overtone Modulated MARY spectroscopy for detection of weak magnetic field effects, 2020. Manuscript in preparation. | ii, 80

[13] Victoire Déjean, Marcin Konowalczyk, Jamie Gravell, Matthew Golesworthy, Catlin Gunn, Nils Pompe, Olivia Van Der Elst, K. J. Tan, Mark Oxborrow, Dirk G. L. Aarts, Stuart R. Mackenzie, Peter J. Hore, and Christiane R. Timmel. Detection of Magnetic Field Effects by Confocal Microscopy, 2020. Manuscript submitted for review in *Chemical Science*. | iii, 96, 97, 101, 104, 106

[14] Peter J. Hore and Henrik Mouritsen. The radical-pair mechanism of magnetoreception. *Annual Review of Biophysics*, 45(1):299–344, 2016. | iii, 95, 145

[15] Andrea Moller, Sven Sagasser, Wolfgang Wiltschko, and Bernd Schierwater. Retinal cryptochrome in a migratory passerine bird: A possible transducer for the avian magnetic compass. *Naturwissenschaften*, 91(12):585–588, 2004. | iii

[16] Jingjing Xu, Lauren Jarocha, Tilo Zollitsch, Marcin Konowalczyk, Matthew Golesworthy, Daniel J. C. Sowood, Maike Herrmann, Hang Yin, Yogarany Chelliah, and Can Xie. Cryptochrome 4 from a night-migratory songbird has the magnetic properties required of a light-dependent magnetoreceptor, 2020. Manuscript submitted for review in *Nature*. | iii, 30, 149

[17] Patrick Winston. 6.034 Artificial Intelligence. `ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-034-artificial-intelligence-fall-2010/`; accessed 16 January 2020. | iv, 55, 56

[18] Peter W. Atkins and Ronald S. Friedman. *Molecular Quantum Mechanics*. Oxford University Press, 5 edition, 2011. | 1

[19] David E. Manolopoulos. Operators and Commutators, 2019. Oxford MChem supplementary lecture series 'Quantum Chemistry'. | 1

[20] Richard N. Zare. *Angular Momentum*. Wiley & Sons, Inc., 1988. | 1, 3

[21] A. H. Reddoch. Systematic perturbations of the EPR spectra of anthracene and azulene anions in solution. *The Journal of Chemical Physics*, 43(1):225–234, 1965. | 10

[22] Ilya Kuprov. Density operator formalism. Part of a lecture course on elementary topics in Spin Dynamics; `spindynamics.org`; accessed 22 April 2020. | 14

[23] Strang Gilbert and Cleve Moler. Differential Equations, 2015. | 14

[24] Ilya Kuprov. Defeating the Matrix. *Journal of Magnetic Resonance*, 306:75–79, 2019. | 14

[25] Christopher C. Moser, Jonathan M. Keske, Kurt Warncke, Ramy S. Farid, and P. Leslie Dutton. Nature of biological electron transfer. *Nature*, 355(6363):796–802, 1992. | 17, 17, 18

[26] Harry B. Gray and Jay R. Winkler. Electron Transfer in proteins. *Current Opinion in Chemical Biology*, 3(5):612–618, 1996. | 18

[27] Radek Erban, Jonathan Chapman, and Philip Maini. A practical guide to stochastic simulations of reaction-diffusion processes, 2007. | 19, 111

[28] Jonathan Storey. *Magnetic Field Effects on Electron-Hole Recombination Reactions - Theoretical and Experimental Developments*. Doctoral thesis, University of Oxford, 2014. | 23

[29] Marcin Konowalczyk and Gabriel Moise. Liouville-von-Neumann-Matlab. `github.com/MarcinKonowalczyk/Liouville-von-Neumann-Matlab`; accessed 11 November 2019. | 24

[30] Christiane R. Timmel. *Magnetic Field Effects on Radical Pair Reactions*. Doctoral thesis, University of Oxford, 1998. | 24

[31] K. Schulten and A. Weller. Exploring fast electron transfer processes by magnetic fields. *Biophysical Journal*, 24(1):295–305, 1978. | 25, 146

[32] Bin Liu, Hongtao Liu, Dongping Zhong, and Chentao Lin. Searching for a photocycle of the cryptochrome photoreceptors. *Current Opinion in Plant Biology*, 13(5):578–586, 2010. | 28

[33] Sonja Solar, Nikola Getoff, Parminder S. Surdhar, David A. Armstrong, and Ajit Singh. Oxidation of tryptophan and N-methylindole by $N_3^\bullet$, $Br_2^{\bullet-}$, and $(SCN)_2^{\bullet-}$ radicals in light-and heavy-water solutions: A pulse radiolysis study. *Journal of Physical Chemistry*, 95(9):3639–3643, 1991. | 28

[34] Jeremiah E. Miller, Cristian Grădinaru, Brian R. Crane, Angel J. Di Bilio, William A. Wehbi, Sun Un, Jay R. Winkler, and Harry B. Gray. Spectroscopy and Reactivity of a Photogenerated Tryptophan Radical in a Structurally Defined Protein Environment. *Journal of the American Chemical Society*, 125(47):14220–14221, 2003. | 28

[35] Javier Galbán, Isabel Sanz-Vicente, Jesús Navarro, and Susana De Marcos. The intrinsic fluorescence of FAD and its application in analytical chemistry: A review. *Methods and Applications in Fluorescence*, 4(4), 2016. | 28

[36] Donat P. Häder and Giulio Jori. *Flavins Photochemistry and Photobiology*. RSC publishing, 2013. | 28
[37] José I. García, Milagros Medina, Javier Sancho, Pablo J. Alonso, Carlos Gómez-Moreno, José A. Mayoral, and Jesús I. Martínez. Theoretical analysis of the electron spin density distribution of the Flavin semiquinone isoalloxazine ring within model protein environments. *Journal of Physical Chemistry A*, 106(18):4729–4735, 2002. | 28
[38] Daniel Nohr, Sophie Franz, Ryan Rodriguez, Bernd Paulus, Lars-Oliver Essen, Stefan Weber, and Erik Schleicher. Extended Electron-Transfer in Animal Cryptochromes Mediated by a Tetrad of Aromatic Amino Acids. *Biophysical Journal*, 111(2):301–311, 2016. | 29, 30, 145
[39] T Miura, Kiminori Maeda, and T Arai. Effect of coulomb interaction on the dynamics of the radical pair in the system of flavin mononucleotide and hen egg-white lysozyme (HEWL) studied by a magnetic field effect. *J. Phys. Chem. B*, 107(26):6474–6478, 2003. | 30
[40] Jiawei Wang, Miroslawa Dauter, Randy Alkire, Andrzej Joachimiak, and Zbigniew Dauter. Triclinic lysozyme at 0.65 Å resolution. *Acta Crystallographica Section D: Biological Crystallography*, 63(12):1254–1268, 2007. | 30
[41] Margaret Ahmad, Jose A. Jarillo, Olga Smirnova, and Anthony R. Cashmore. Cryptochrome blue-light photoreceptors of Arabidopsis implicated in phototropism. *Nature*, 392(6677):720–723, 1998. | 30
[42] Roger Jan Kutta, Nataliya Archipowa, and Nigel Shaun Scrutton. The sacrificial inactivation of the blue-light photosensor cryptochrome from: Drosophila melanogaster. *Physical Chemistry Chemical Physics*, 20(45):28767–28776, 2018. | 30, 145
[43] Paul J. Nahin. *Dr. Euler's fabulous formula*. Princeton University Press, 2006. | 32
[44] Brian Hayes. *Foolproof, and Other Mathematical Meditations*. MIT Press, 2017. | 34, 48
[45] Aurélie Jost and Rainer Heintzmann. Superresolution Multidimensional Imaging with Structured Illumination Microscopy. *Annual Review of Materials Research*, 43(1):261–282, 2013. | 35
[46] David L Hershberger and Nevada City. Controlled Envelope Single Sideband. *QEX*, pages 3–13, 2014. | 37
[47] James W. Cooley and John W. Tukey. An Algorithm for the Machine Calculation of Complex Fourier Series. *Mathematics of Computation*, 19(90):297, 1965. | 38
[48] Steven W. Smith. *Digital Signal Processing: A Practical Guide for Engineers and Scientists*. IDC Technology, 2 edition, 1997. | 38
[49] Martin Knapp-Cordes and Bill McKeeman. Improvements to tic and toc Functions for Measuring Absolute Elapsed Time Performance in MATLAB - MATLAB & Simulink. Technical report, MathWorks, 2011. | 38
[50] Matt Parker. *Humble Pi*. Penguin books, 2019. | 38
[51] Marcin Konowalczyk. randnd, 2017. `github.com/MarcinKonowalczyk/randnd`; accessed 09 January 2020. | 43, 178
[52] Eric W. Weisstein. Tanc Function. `mathworld.wolfram.com/TancFunction.html`; accessed 22 April 2020. | 46
[53] Lloyd Nicholas Trefethen. Maxims About Numerical Mathematics, Computers, Science, and Life. *SIAM news*, 1998. | 46, 52
[54] Fundamental Physical Constants. `physics.nist.gov/cuu/Constants/index.html`; accessed 16 January 2020. | 46
[55] Matt Parker. *Things to make and do in the fourth dimension*. Penguin books, 2015. | 48
[56] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, 2nd edition, 2018. | 50, 50
[57] Optimization SciPy v1.4.1 Reference Guide. `docs.scipy.org/doc/scipy/reference/tutorial/optimize.html`; accessed 16 January 2020. | 50, 55
[58] J. A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, 1965. | 51, 51, 92
[59] Marius Muja and David G. Lowe. Scalable nearest neighbor algorithms for high dimensional data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(11):2227–2240, 2014. | 52
[60] Eric W. Weisstein. Global Optimization. `mathworld.wolfram.com/GlobalOptimization.html`. | 55

[61] Gregory S. Hornby, Al Globus, Derek S. Linden, and Jason D. Lohn. Automated antenna design with evolutionary algorithms. *Collection of Technical Papers - Space 2006 Conference*, 1:445–452, 2006. | 56

[62] Nicolás García-Pedrajas, Domingo Ortiz-Boyer, and César Hervás-Martínez. An alternative approach for neural network evolution with a genetic algorithm: Crossover by combinatorial optimization. *Neural Networks*, 19(4):514–528, 2006. | 56

[63] Firas Khatib, Seth Cooper, Michael D. Tyka, Kefan Xu, Ilya Makedon, Zoran Popović, David Baker, and Foldit Players. Algorithm discovery by protein folding game players. *Proceedings of the National Academy of Sciences of the United States of America*, 108(47):18949–18953, 2011.

[64] Ron Unger and John Moult. Genetic algorithms for protein folding simulations. *Journal of Molecular Biology*, 231(1):75–81, 1993. | 56

[65] Abraham. Savitzky and M. J. E. Golay. Smoothing and Differentiation of Data by Simplified Least Squares Procedures. *Analytical Chemistry*, 36(8):1627–1639, 1964. | 59, 101

[66] J. N. Lyness and Cleve B. Moler. Numerical Differentiation of Analytic Functions. *SIAM Journal on Numerical Analysis*, 4(2):202–210, 1967. | 66

[67] Cleve B. Moler. Complex Step Differentiation » Cleve's Corner: Cleve Moler on Mathematics and Computing - MATLAB & Simulink, 2013. | 66

[68] James Christian and Holly Middleton-Spencer. The magnetic pendulum. *Chalkdust*, pages 9–14, 2018. | 75

[69] Steven Henry Strogatz. *Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering*. Taylor & Francis Group, 2nd edition, 2015. | 76

[70] S. N. Batchelor, C. W. M. Kay, K. A. Mclauchlan, and I. A. Shkrob. Time-resolved and modulation methods in the study of the effects of magnetic-fields on the yields of free-radical reactions. *J. Phys. Chem.*, pages 13250–13258, 1993. | 80, 81, 93

[71] Kelly-Anne Ferguson. *An Investigation of isotropic and anisotropic magnetic field effects in fluorescent systems*. PhD thesis, University of Oxford, 2013. | 80, 80, 92, 92, 94

[72] M. Justinek, G. Grampp, S. Landgraf, Peter J. Hore, and N. N. Lukzen. Electron self-exchange kinetics determined by mary spectroscopy: theory and experiment. *Journal of the American Chemical Society*, 126(17):5635–46, may 2004. | 80, 92, 94

[73] Christopher J. Wedge. *Radiofrequency Magnetic Field Effects on Radical Pair Recombination Reactions*. D.phil, University of Oxford, 2009. | 80, 81, 92

[74] Kunal Pal, Daniel R. Kattnig, Günter Grampp, and Stephan Landgraf. Experimental observation of preferential solvation on a radical ion pair using mary spectroscopy. *Physical chemistry chemical physics*, 14(9):3155–61, mar 2012. | 80, 81

[75] Christiane R. Timmel and Kevin B. Henbest. A study of spin chemistry in weak magnetic fields. *Philosophical Transactions of the Royal Society A*, 362(1825):2573–2589, dec 2004. | 80, 93

[76] Nan Yang and Adam E. Cohen. Optical imaging through scattering media via magnetically modulated fluorescence. *Opt. Express*, 18(25):25461–25467, 2010. | 80

[77] Joshua P. Beardmore, Lewis M. Antill, and Jonathan R. Woodward. Optical absorption and magnetic field effect based imaging of transient radicals. *Angewandte Chemie International Edition*, 54(29):8494–8497, 2015. | 80

[78] Jeffrey N. Anker, Caleb J. Behrend, Hongmei Huang, and Raoul Kopelman. Magnetically-modulated optical nanoprobes (magmoons) and systems. *Journal of Magnetism and Magnetic Materials*, 293(1):655 – 662, 2005. Proceedings of the Fifth International Conference on Scientific and Clinical Applications of Magnetic Carriers. | 80

[79] Randall M. Erb, Joshua J. Martin, Rasam Soheilian, Chunzhou Pan, and Jabulani R. Barber. Actuating soft matter with magnetic torque. *Advanced Functional Materials*, 26(22):3859–3880, 2016. | 80

[80] Ove Axner, Patrick Ehlers, Aleksandra Foltynowicz, Isak Silander, and Junyang Wang. *NICE-OHMS—Frequency Modulation Cavity-Enhanced Spectroscopy—Principles and Performance*, pages 211–251. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014. | 81

[81] M. L. Meade. Advances in lock-in amplifiers. *J. Phys. E: Sci. Instrum.*, 15:395–403, 1982. | 81
[82] Zürich Instruments. Principles of lock-in detection and the state of the art zürich instruments. Technical report, Zürich Instruments, 2016.
[83] AMTEK Advanced Measurment Technology Inc. Low level optical detection using lock-in amplifier techniques. Technical report, AMTEK Advanced Measurment Technology Inc., 2008.
[84] Stanford Research Systems Inc. About lock-in amplifiers. Technical report, Stanford Research Systems Inc., 2011. | 81, 82
[85] C.A. Hamilton, J.P. Hewitt, K.A. McLauchlan, and U.E. Steiner. High resolution studies of the effects of magnetic fields on chemical reactions. *Molecular Physics*, 65(2):423–438, 1988. | 81, 92
[86] Paul Horowitz and Winfield Hill. *The Art Of Electronics*. Cambridge University Press, 3rd edition, 2015. | 81
[87] S. E. Tavares. A comparison of integration and low-pass filtering. *IEEE Transactions on Instrumentation and Measurement*, 15:33–38, 1966. | 81
[88] Peter J. Hore, Jonathan Jones, and Stephen Wimperis. *NMR: The Toolkit*. Oxford University Press, 2nd edition, 2015. | 82
[89] Mark Tseitlin, Sandra S. Eaton, and Gareth R. Eaton. Reconstruction of the first-derivative epr spectrum from multiple harmonics of the field-modulated continuous wave signal. *Journal of Magnetic Resonance*, 209(2):277–281, 2011. | 84, 94
[90] National Instruments. Extract single tone information vi. Part of the Signal Processing Toolbox, LabVIEW 2015, Online documentation accessed 2019. | 89
[91] J. G. Ziegler and N. B. Nichols. Optimum settings for automatic controllers. *InTech*, 42(6):94–100, 1995. | 90
[92] Philip. M. Dixon. Bootstrap resampling. *Encyclopedia of Environmetrics*, 2006. | 92
[93] James Carpenter and John Bithell. Bootstrap confidence intervals: when, which, what? a practical guide for medical statisticians. *Statistics in Medicine*, 19:1141–1164, 2000.
[94] S. Huet, A. Bouvier, M.-A. Poursat, and E. Jolivet. *Statistical Tools for Nonlinear Regression*. Springer US, 2nd edition, 2010. | 92, 94
[95] MathWorks Inc. Matlab 2017a, 2017. | 92
[96] Susan L. Mattes and Samir Farid. Exciplexes and electron transfer reactions. *Science*, 226(4677):917–921, 1984. | 92, 92
[97] Günter Grampp, M. Justinek, and Stephan Landgraf. Magnetic field effects on the pyrene-dicyanobenzene system: determination of electron self-exchange rates by mary spectroscopy. *Molecular Physics*, 100(8):1063–1070, apr 2002. | 92
[98] C. J. Wedge, Christopher T. Rodgers, Stuart A. Norman, Neville Baker, Kiminori Maeda, Kevin B. Henbest, C. R. Timmel, and P. J. Hore. Radiofrequency polarization effects in low-field electron paramagnetic resonance. *Physical Chemistry Chemical Physics*, 11(31):6573–6579, 2009. | 92
[99] Daniel R. Kattnig, Arnulf Rosspeintner, and Günter Grampp. Magnetic field effects on exciplex-forming systems: the effect on the locally excited fluorophore and its dependence on free energy. *Physical chemistry chemical physics*, 13(8):3446–60, feb 2011. | 92
[100] M. E. Sigman, P. F. Schuler, M. M. Ghosh, and R. T. Dabestani. Mechanism of pyrene photochemical oxidation in aqueous and surfactant solutions. *Environmental Science & Technology*, 32(24):3980–3985, 1998. | 93
[101] Alan M. Lewis, Thomas P. Fay, David E. Manolopoulos, Christian Kerpal, Sabine Richert, and Christiane R. Timmel. On the low magnetic field effect in radical pair reactions. *Journal of Chemical Physics*, 149(3), 2018. | 94
[102] Sabine Richert, Arnulf Rosspeintner, Stephan Landgraf, Günter Grampp, Eric Vauthey, and Daniel R. Kattnig. Time-resolved magnetic field effects distinguish loose ion pairs from exciplexes. *Journal of the American Chemical Society*, 135(40):15144–52, oct 2013. | 94

[103] Jukka Juutilainen, Mikko Herrala, Jukka Luukkonen, Jonne Naarala, and P. J. Hore. Magnetocarcinogenesis: Is there a mechanism for carcinogenic effects of weak magnetic fields? *Proceedings of the Royal Society B*, 285(1879), 2018. | 95

[104] Catlin Gunn. *Design and application of sensitive detection methods for the study of biological systems*. Doctoral thesis, University of Oxford, 2015. | 96

[105] Klaus Suhling, James A. Levitt, Pei Hua Chung, Marina K. Kuimova, and Gokhan Yahioglu. Fluorescence lifetime imaging of molecular rotors in living cells. *Journal of Visualized Experiments*, 1(60), 2012. | 103

[106] Xu Wang, Fengling Song, and Xiaojun Peng. A versatile fluorescent probe for imaging viscosity and hypochlorite in living cells. *Dyes and Pigments*, 2016. | 103

[107] B. W. Matthews. Solvent content of protein crystals. *Journal of Molecular Biology*, 1968. | 104

[108] Hiroyasu Tabe, Satoshi Abe, Tatsuo Hikage, Susumu Kitagawa, and Takafumi Ueno. Porous protein crystals as catalytic vessels for organometallic complexes. *Asian Chemical Editorial Society*, 9(5):1373–1378, 2014. | 104

[109] Theodore Tzedakis, Kane Cheikhou, Roche Jerôme, Groenen Serrano Karine, and Reynes Olivier. Electrochemical study in both classical cell and microreactors of flavin adenine dinucleotide as a redox mediator for NADH regeneration. *Electrochimica Acta*, 55(7):2286–2294, 2010. | 104

[110] A. Cvetkovic, A. J. J. Straathof, D. N. Hanlon, S. Van Der Zwaag, R. Krishna, and L. a M Van Der Wielen. Quantifying anisotropic solute transport in protein crystals using 3D laser scanning confocal microscopy visualization. *Biotechnology and Bioengineering*, 86(4):389–398, 2004. | 104

[111] O. D. Velev, E. W. Kaler, and a. M. Lenhoff. Surfactant Diffusion into Lysozyme Crystal Matrices Investigated by Quantitative Fluorescence Microscopy. *The Journal of Physical Chemistry B*, 104(39):9267–9275, 2000. | 104

[112] Muhammad A. Sheraz, Sadia H. Kazi, Sofia Ahmed, Kiran Qadeer, Marium F. Khan, and Iqbal Ahmad. Multicomponent spectrometric analysis of riboflavin and photoproducts and their kinetic applications. *Central European Journal of Chemistry*, 12(6):635–642, 2014. | 105

[113] Emrys W Evans. *Making physics work for biology : using spin chemistry to understand protein mechanism and structure*. Doctoral thesis, University of Oxford, 2016. | 106

[114] Derek Ainslie Jackson. The spherical fabry-perot interferometer as an instrument of high resolving power for use with external or with internal atomic beams. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 263(1314):289–308, 1961. | 113

[115] Anthony O'Keefe and David A. G. Deacon. Cavity ring-down optical spectrometer for absorption measurements using pulsed laser sources. *Review of Scientific Instruments*, 59(12):2544–2551, 1988. | 113

[116] Richard Engeln, Giel Berden, Rudy Peeters, and Gerard Meijer. Cavity enhanced absorption and cavity enhanced magnetic rotation spectroscopy. *Review of Scientific Instruments*, 69(11):3763–3769, 1998. | 113

[117] Anthony O'Keefe. Integrated cavity output analysis of ultra-weak absorption. *Chemical Physics Letters*, 293(5):331 – 336, 1998. | 113

[118] A. A. Ruth, S. Dixneuf, and R. Raghunandan. *Broadband Cavity-Enhanced Absorption Spectroscopy with Incoherent Light*, pages 485–517. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014. | 113

[119] S Fiedler. Incoherent broad-band cavity-enhanced absorption spectroscopy. *Chemical Physics Letters*, 371(3-4):284–294, 2003. | 113, 119

[120] J. M. Langridge, T. Laurila, R. S. Watt, R. L. Jones, C. F. Kaminski, and J. Hult. Cavity enhanced absorption spectroscopy of multiple trace gas species using a supercontinuum radiation source. *Opt. Express*, 16(14):10178 – 10188, 2008. | 113

[121] Daniele Romanini, Irène Ventrillard, Guillaume Méjean, Jérôme Morville, and Erik Kerstel. *Introduction to Cavity Enhanced Absorption Spectroscopy*, pages 1–60. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014. | 113, 113

[122] Thomas M. Crawford. Error Sources In The Ring Down Optical Cavity Decay Time Mirror Reflectometer. In Susanne C. Stotlar, editor, *Southwest Conf. on Optics '85*, volume 0540, pages 295 – 302. International Society for Optics and Photonics, SPIE, 1985. | 113

[123] H. Waechter, J. Litman, A. H. Cheung, J. A. Barnes, and H.-P. Loock. Chemical sensing using fiber cavity ring-down spectroscopy. *Sensors*, 10:1716 – 1742, 2010. | 114

[124] MathWorks Inc. `bsxfun` documentation. `uk.mathworks.com/help/matlab/ref/bsxfun.html`; accessed 14 January 2020. | 116

[125] Dana Z. Anderson. Alignment of resonant optical cavities. *Appl. Opt.*, 23(17):2944 – 2949, 1984. | 117

[126] Kevin K. Lehmann and Daniele Romanini. The superposition principle and cavity ring-down spectroscopy. *Journal of Chemical Physics*, 105(23):10263–10277, 1996. | 120

[127] Mikhail Mazurenka, Andrew J. Orr-Ewing, Robert Peverall, and Grant A. D. Ritchie. Cavity ring-down and cavity enhanced spectroscopy using diode lasers. *Annu. Rep. Prog. Chem., Sect. C: Phys. Chem.*, 101:100 — 142, 2005. | 120

[128] Sven E. Fiedler, Achim Hese, and Albert a. Ruth. Incoherent broad-band cavity-enhanced absorption spectroscopy of liquids. *Review of Scientific Instruments*, 76(2):023107, 2005. | 123

[129] NKT Photonics Inc. Supercontinuum Generation in Photonics Crystal Fibers. Technical report, NTK Photonics Inc., 2009. | 126

[130] John M. Dudley, Goëry Genty, and Stéphane Coen. Supercontinuum generation in photonic crystal fiber. *Reviews of Modern Physics*, 78(4):1135–1184, 2006. | 126

[131] Dean M.W. Sheppard. *Cavity Enhanced Detection of Biologically Relevant Magnetic Field Effects*. Doctoral thesis, University of Oxford, 2016. | 128, 140

[132] Sandeep Koranne and Sandeep Koranne. Hierarchical Data Format 5. In *Handbook of Open Source Tools*, pages 191–200. Springer US, 2011. | 130

[133] Bernd Giese, Min Wang, Jian Gao, Moritz Stoltz, Pavel Müller, and Michael Graber. Electron relay race in peptides. *Journal of Organic Chemistry*, 74(10):3621–3625, 2009. | 132

[134] V. Leskovac, S. Trivić, G. Wohlfahrt, J. Kandrač, and D. Peričin. Glucose oxidase from Aspergillus niger: The mechanism of action with molecular oxygen, quinones, and one-electron acceptors. *International Journal of Biochemistry and Cell Biology*, 37(4):731–750, 2005. | 138

[135] Lipsa Nag, Andras Lukacs, and Marten H. Vos. Short-Lived Radical Intermediates in the Photochemistry of Glucose Oxidase. *ChemPhysChem*, 20(14):1793–1798, 2019. | 138

[136] A. A. P. Trichet, J. Foster, N. E. Omori, D. James, P. R. Dolan, G. M. Hughes, C. Vallance, and J. M. Smith. Open-access optical microcavities for lab-on-a-chip refractive index sensing. *Lab Chip*, 14:4244–4249, 2014. | 140

[137] Simon R. T. Neil, Cathy M. Rushworth, Claire Vallance, and Stuart R. Mackenzie. Broadband cavity-enhanced absorption spectroscopy for real time, in situ spectral analysis of microfluidic droplets. *Lab Chip*, 11:3953–3955, 2011. | 140

[138] Pavel Müller and Margaret Ahmad. Light-activated cryptochrome reacts with molecular oxygen to form a flavin–superoxide radical pair consistent with magnetoreception. *The Journal of Biological Chemistry*, 286:21033 – 21040, 2011. | 143

[139] Colin Levy, Brian D. Zoltowski, Alex R. Jones, Anand T. Vaidya, Deniz Top, Joanne Widom, Michael W. Young, Nigel S. Scrutton, Brian R. Crane, and David Leys. Updated structure of Drosophila cryptochrome, 2013. | 142

[140] Chad A. Brautigam, Barbara S. Smith, Zhiquan Ma, Maya Palnitkar, Diana R. Tomchick, Mischa Machius, and Johann Deisenhofer. Structure of the photolyase-like domain of cryptochrome 1 from Arabidopsis thaliana. *Proceedings of the National Academy of Sciences of the United States of America*, 101(33):12142–12147, 2004. | 144

[141] Brian D. Zoltowski, Yogarany Chelliah, Anushka Wickramaratne, Lauren Jarocha, Nischal Karki, Wei Xu, Henrik Mouritsen, Peter J. Hore, Ryan E. Hibbs, Carla B. Green, and Joseph S. Takahashi. Chemical and

structural analysis of a photoactive vertebrate cryptochrome from pigeon. *Proceedings of the National Academy of Sciences of the United States of America*, 116(39):19449–19457, 2019.

[142] Gerd Wohlfahrt, Susanne Witt, Jörg Hendle, Dietmar Schomburg, Henryk M. Kalisz, and Hans Jürgen Hecht. 1.8 and 1.9 Å resolution structures of the Penicillium amagasakiense and Aspergillus niger glucose oxidases as a basis for modelling substrate complexes. *Acta Crystallographica Section D: Biological Crystallography*, 55(5):969–977, 1999. | 142

[143] Kiminori Maeda, Alexander J. Robinson, Kevin B. Henbest, Hannah J. Hogben, Till Biskup, Margaret Ahmad, Erik Schleicher, Stefan Weber, Christiane R. Timmel, and Peter J. Hore. Magnetically sensitive light-induced reactions in cryptochrome are consistent with its proposed role as a magnetoreceptor. *Proceedings of the National Academy of Sciences*, 109(13):4774–4779, 2012. | 143

[144] Heng Li, T. B. Melø, and K. Razi Naqvi. Triplets, radical cations and neutral semiquinone radicals of lumiflavin and riboflavin: An overhaul of previous pump-probe data and new multichannel absolute absorption spectra. *Journal of Photochemistry and Photobiology B: Biology*, 106(1):34–39, 2012. | 143

[145] Miriam Liedvogel and Henrik Mouritsen. Cryptochromes – a potential magnetoreceptor: what do we know and what do we want to know? *Journal of the Royal Society Interface*, 7(Suppl 2):S147–62, 2010. | 145

[146] Henrik Mouritsen. Long-distance navigation and magnetoreception in migratory animals. *Nature*, 558(7708):50–59, 2018. | 145

[147] A. Weller, F. Nolting, and H. Staerk. A quantitative interpretation of the magnetic field effect on hyperfine-coupling-induced triplet fromation from radical ion pairs. *Chemical Physics Letters*, 96(1):24–27, 1983. | 146

[148] Rebecca Pogni, M. Camilla Baratto, Christian Teutloff, Stefania Giansanti, Francisco J. Ruiz-Dueñas, Thomas Choinowski, Klaus Piontek, Angel T. Martínez, Friedhelm Lendzian, and Riccardo Basosi. A tryptophan neutral radical in the oxidized state of versatile peroxidase from Pleurotus eryngii: A combined multifrequency EPR and density functional theory study. *Journal of Biological Chemistry*, 281(14):9517–9526, 2006. | 146

[149] Changfan Lin, Deniz Top, Craig C. Manahan, Michael W. Young, and Brian R. Crane. Circadian clock activity of cryptochrome relies on tryptophan-mediated photoreduction. *Proceedings of the National Academy of Sciences*, 115(15):3822–3827, 2018. | 154

[150] Stephen Menke, Alexandre Cheminal, Patrick Conaghan, Niva Ran, Neil Greehnam, Guillermo Bazan, Thuc-Quyen Nguyen, Akshay Rao Ananth, and Richard Friend. Order enables efficient electron-hole separation at an organic heterojunction with a small energy loss. *Nature Communications*, 9, 2018. | 154

[151] Glenn R. Masson, John E. Burke, Natalie G. Ahn, and Anand *et al.* Recommendations for performing, interpreting and reporting hydrogen deuterium exchange mass spectrometry (hdx-ms) experiments. *Nature Methods*, 16(7):595–602, 2019. | 154

[152] Haik Chosrowjan, Seiji Taniguchi, and Fumio Tanaka. Ultrafast fluorescence upconversion technique and its applications to proteins. *The FEBS Journal*, 282(16):3003–3015, 2015. | 154

[153] Jessica Ruth Walton. *Investigating Avian Magnetoreception Using Broadband Cavity-Enhanced Absorption Spectroscopy*. Masters thesis, University of Oxford, 2019. | 155

[154] Marcin Konowalczyk. allan2, 2018. `uk.mathworks.com/matlabcentral/fileexchange/65754-allan2`; accessed 09 January 2020. | 187

List of Figures

1.1 Electronic α and β spin states of a free electron in an external magnetic field. 4
1.2 Hamiltonian for an unpaired electron and with spin-½ nucleus in an external magnetic field 11
1.3 Zeeman interaction of the e-n spin system, and spin in product and coupled basis 11
1.4 Representation of coupled spin basis as linear combinations of product basis. 12
1.5 Overlap of two harmonic potential wavefunctions and ET rate as predicted by Marcus theory 16
1.6 Details of the Franck-Condon overlap for selected ΔG values. 18
1.7 Reaction kinetics of ET with reoxidation, and a simple photocycle steady-state behaviour. . 21
1.8 Hamiltonian matrix for the interaction of two electronic, and one nuclear spin, in mixed basis. 23
1.9 Energy levels of coupled spin states and singlet yield, in an external magnetic field. 24
1.10 Demonstration of the kinetics of MFE accumulation 27
1.11 Absorption spectra of selected redox stated of FAD and Trp 28
1.12 Generalised photocycle of flavin-based RP chemistry, and the ET chain of *Dm*Cry WT. 29

2.1 Sub-Nyquist and super-Nyquist reconstruction and Nyquist 'reflection' 34
2.2 Signal downsampling with and without aliasing 36
2.3 Circularly periodic view of a signal from the perspective of Fourier transform 37
2.4 Performance of the FFT algorithm 39
2.5 Correct ($N-1$) and incorrect (edge-to-edge) sampling of a single tone signal. 39
2.6 Effects of windowing of 3 Hz and 4.5 Hz signals on scalloping and sidelobe magnitude. 40
2.7 Convolution of two tones 42
2.8 Modulation/Demodulation and its effect on low-frequency noise. 44
2.9 Gabor transform of a signal with changing frequency content 46
2.10 Simple Lorentzian fit and associated loss function hypersurface 48
2.11 Comparison of gradient descent, Nelder-Mead simplex and random search 52
2.12 Basins of attraction of a hypersurface with multiple local minima 54
2.13 Flattop window construction by optimisation 56
2.14 Hypersurfaces for the flattop window construction problem 57
2.15 Regularisation of numerical differentiation and linear regression 60
2.16 Eigenvalue solution by optimisation and asymmetric prediction bounds estimate. 62
2.17 Solution to the Van der Pol IVP with different timestepping methods 68
2.18 Reflecting oscillation of a Gaussian pulse 71
2.19 Propagation of an electric wave through a dielectric and the resulting resonant structure . . 72
2.20 Solution to 1D time-independent Schrödinger equation for a particle in a potential well. . . 75
2.21 Magnetic pendulum and emergence of chaos 76
2.22 Demonstration of correlation and the central limit theorem 78

3.1 MARY curve as a transfer function from magnetic field to fluorescence intensity 81
3.2 Log local error (LLE) in successive Taylor expansions of a Lorentzian lineshape 85
3.3 Pictorial representation of the detection kernel D and plot of the modulation kernels 86
3.4 Schematic of the ModMARY apparatus 89
3.5 Simultaneously fitted 1^{st} and 2^{nd} harmonic data for PyH_{10} and PyD_{10} 91
3.6 Photoscheme of the pyrene/1,3-dicyanobenzene system. 92
3.7 Reconstruction of the MARY curves of PyH_{10} and PyD_{10} 93

3.8 Points of interest of the fits 94

4.1 The principle of confocal microscopy 97
4.2 Representative average pixel intensity and outline of the step interpolation methodology . . 98
4.3 Fluorescence images of FMN and crystalline HEWL. FOV over an entire crystal. 103
4.4 Raw and shape-corrected L2 slices through the crystal 104
4.5 Comparison of the spatiotemporally resolved fluorescence intensity and %MFE of the crystal.105
4.6 Time evolution of the L1 slice of the MFE 105
4.7 Time evolution of the pixel intensity change inside and outside of the crystal 107
4.8 Average MFE step in two ROIs – inside and outside of the crystal 107
4.9 Fluorescence images of FMN and crystalline HEWL. FOV just inside of the crystal body. . . . 108
4.10 L1 slices through the fluorescence intensity 108
4.11 L1 slices through spatially-resolved %MFE of the data 109
4.12 Time evolution of the average pixel intensity of the entire frame relative to the centerline . . 109
4.13 Spatially-resolved EMFE 110
4.14 Fitted EMFE for the path, wave and wake regions 111

5.1 Numerical calculation of the Airy distribution 117
5.2 Schematic of the light roundtrip though the cavity, and transfer function T to I/I_0 119
5.3 Stability condition for an optical resonator 122
5.4 Euclidean distance measure for R_{eff} fitting, and the resulting CEF 125
5.5 Outline of the BBCEAS apparatus and the MFE experiment 128
5.6 ΔA surface of FMN/HEWL 130
5.7 ΔΔA MARY surface of FMN/HEWL 131
5.8 Calibration dye absorption spectrum and VND filter angle calibration 133
5.9 Comparison of the old and new supercontinuum sources spectral shape and stability 134
5.10 Influence of the M1 dither on the transmitted light intensity and MFE measurement. 135
5.11 Intensity shape drift, and shape correction. 136
5.12 ΔA and ΔΔA surfaces of water (blank) 137
5.13 ΔA and ΔΔA surfaces of GOx (blank) 138
5.14 Satellite detected MFE on FMN/HEWL 138
5.15 Satellite and overtone MARY surfaces of FMN/HEWL 139

6.1 Nomenclature of the protein samples studied 141
6.2 ΔA surface of *At*Cry WT 142
6.3 ΔΔA surface of *At*Cry WT 143
6.4 ΔA and ΔΔA surfaces of dilute *Dm*Cry W394F 144
6.5 Comparison of ΔA surfaces of *Dm*Cry WT and *Dm*Cry W394F mutant. 145
6.6 Comparison of ΔΔA surfaces of *Dm*Cry WT and *Dm*Cry W394F mutant. 145
6.7 Comparison of ΔA surfaces of *Er*Cry WT and *Er*Cry W369F mutant. 146
6.8 Comparison of ΔΔA surfaces of *Er*Cry WT and *Er*Cry W369F mutant. 147
6.9 Comparison of ΔΔA time evolution of *Er*Cry WT and *Er*Cry W369F mutant. 148
6.10 Comparison of avian cryptochromes and their C317 and K320 mutants. 150
6.11 Comparison of ΔΔA of avian cryptochromes and their selected mutants. 151
6.12 ΔA and ΔΔA surfaces of *Gg*Cry 152
6.13 ΔA and ΔΔA surfaces of *Cl*Cry 152
6.14 ΔA and ΔΔA surfaces of *Tg*Cry 152
6.15 ΔA and ΔΔA surfaces of *Gg*Cry R317C 153
6.16 ΔA and ΔΔA surfaces of *Tg*Cry S317C 153
6.17 ΔA and ΔΔA surfaces of *Gg*Cry R317C+E320K 153

B.1 Field burst and measured field hysteresis 192
B.2 2D excitation/emission spectrum of Py /DCB 192
B.3 Mode width and stability plotted on the M1/M2 radius axis. 193
B.4 Measured mirror transmittance 193
B.5 Model of the sample cell used 193
B.6 Model of the cell mount 194
B.7 Model of the cell clamp 194
B.8 Streak camera image of a SC source pulse 194
B.9 Acid titration curves of Tris and BTM buffer solutions 195
B.10 ΔΔA surfaces of *Er*Cry WT at early and late times 195
B.11 ΔΔA surfaces of *Er*Cry W369F at early and late times 195

List of Listings

1.1 Diagonalisation of the Hamiltonian for a simple spin system . . . 10
1.2 Franck-Condon overlap integral of harmonic oscillator wavefunctions. . . . 17
1.3 Simple A→B kinetics simulation. . . . 19
1.4 Electron transfer with reoxidation simulation. . . . 20
1.5 Photocycle kinetics simulation . . . 21
1.6 Singlet yield as a function of magnetic field . . . 23
1.7 Demonstration of the kinetics of MFE accumulation . . . 26

2.1 FFT reversibility. . . . 34
2.2 Demonstrations of sampling theorem. . . . 35
2.3 Demonstration of signal downsampling. . . . 35
2.4 Construction of the fft frequency axis. . . . 37
2.5 Construction of the fft half-frequency spectrum and axis. . . . 37
2.6 Reconstruction of signal from single-sided spectrum. . . . 37
2.7 Timing and accuracy benchmark of the FFT algorithm. . . . 38
2.8 $N+1$ and edge-to-edge timesampling of a signal. . . . 39
2.9 Convolution theorem verification . . . 42
2.10 Modulation/demodulation of a signal. . . . 43
2.11 Gabor transform of a signal with changing frequency content . . . 45
2.12 Loss function hypersurface for a Lorentzian fit . . . 47
2.13 Basic gradient descent algorithm. . . . 49
2.14 Central difference scheme. . . . 50
2.15 Optimisation with `fminunc` . . . 51
2.16 Simplified Nelder-Mead algorithm. . . . 51
2.17 Simple random search optimisation algorithm. . . . 52
2.18 Basins of attraction of the Nelder-Mead algorithm. . . . 53
2.19 Create a 3-parameter window scaled to unit height. . . . 54
2.20 Loss function for creating a simple flattop window. . . . 55
2.21 Multistart global optimisation. . . . 55
2.22 Example of an ill-posed problem. . . . 57
2.23 Better formulation of the same problem. . . . 58
2.24 Optimisation solution to the numerical derivative problem. . . . 59
2.25 Regularised solution to the numerical derivative problem. . . . 59
2.26 Regularised least-squares fit. . . . 59
2.27 Linear regresion data with outliers. . . . 59
2.28 Linear regresion models. . . . 60
2.29 Eigenvalues of a random 3×3 matrix by optimisation. . . . 61
2.30 Fit uncertainty based on the Hessian. . . . 62
2.31 Fit uncertainty based on bootstrap resampling of the residulas. . . . 63
2.32 Central tendency and upper/lower spread estimation. . . . 63
2.33 Numerical derivative methods. . . . 66
2.34 Setup of the Van der Pol equation . . . 67

2.35 Forward Euler timestepping . . . 67
2.36 Backward Euler timestepping. . . . 68
2.37 Variable timestep + FE. . . . 68
2.38 Crank-Nicolson timestepping. . . . 69
2.39 4th order Runge-Kutta timestepping . . . 69
2.40 EM radiation setup . . . 70
2.41 Driving force. . . . 71
2.42 Reflecting gaussian. . . . 71
2.43 EM radiation ODE function . . . 71
2.44 Quantum harmonic oscillator setup. . . . 74
2.45 Shooting method. . . . 74
2.46 Magnetic pendulum. . . . 76
2.47 Calculation of uncertainty of %MFE . . . 79
2.48 Sample noise from an arbitrary PDF . . . 79

4.1 Average pixel intensity of an ROI of each frame in a stack. . . . 98
4.2 Example code for obtaining spatiotemporally-resolved MFE. . . . 100
4.3 Determination of the fluorescence baseline. . . . 101
4.4 Obtaining slices through each frame of the stack of frames. . . . 101
4.5 Correcting the shape of slices according to a model. . . . 102
4.6 Correcting the shape of an entire tstack such that the first frame is flat. . . . 102

5.1 Numerical calculation of the Airy distribution . . . 116
5.2 Raytracing simulation of the cavity stability . . . 121
5.3 Spectral fitting of R_{eff} . . . 124
5.4 Optimisation function for R_{eff} fitting . . . 124

A.1 Comparison of `abs()` and `real()` for rounding after `ifft(fft(...))` of a real signal. . . . 175
A.2 Comparison of fft and a naïve Fourier transform calculation. . . . 175
A.3 Single sided, windowed FFT and an accompanying frequency axis. . . . 176
A.4 Random search optimisation function. . . . 176
A.5 Random coloured N-dimentional noise . . . 177
A.6 Eigenvalues of a large random 50×50 matrix by optimisation. . . . 178
A.7 Mirror maker for the raytracing simulation . . . 179
A.8 Raytracing event handler . . . 179
A.9 Ray reflection calculation . . . 179
A.10 Ray impact mapping . . . 180
A.11 Fast polynomial fitting. . . . 180
A.12 Fast weighted polynomial fitting. . . . 180
A.13 Fast polynomial evaluation. . . . 181
A.14 `sigmoid_step` . . . 181
A.15 `exponential_step` . . . 182
A.16 `gaussian_lineshape` . . . 182
A.17 `lorentzian_lineshape` . . . 182
A.18 `asymmetric_gaussian_lineshape` . . . 182
A.19 `havercosine_lineshape` . . . 182
A.20 `sech_lineshape` . . . 182
A.21 `exponential_step_wave` . . . 183
A.22 Harmonic distortion of a single tone by a Lorentzian. . . . 183
A.23 Log local error estimate for a Lorentzian lineshape. . . . 184

A.24 Calculate a correctly scaled version of the modulation kernel. 185
A.25 Numerical derivative of the transfer function 185
A.26 Convolution of the transfer function model specified. 185
A.27 Allan deviation 186
A.28 Single Lorentzian fit with uncertainty calculation. 187
A.29 Input argument parser. 189
A.30 Liouville von Neumann simulation. 189

APPENDIX

Appendix A

Additional code listings

Any additional code listings referenced throughout the text are listed in this appendix.

A.1 Additional examples for FFT

```
y = randn(1,1e3); % random input
fun1 = @(x) abs(ifft(fft(x))); % absolute value
fun2 = @(x) real(ifft(fft(x))); % real part
N = 200;
x = 1:N;
for n = x
    yn1 = y; yn2 = y;
    for j = 1:n % Apply 'fun' n times
        yn1 = fun1(yn1); yn2 = fun2(yn2);
    end
    e1(n) = max(y-yn1)/eps; e2(n) = max(y-yn2)/eps;
end
%% Gradients of each of the errors
g1 = (e1)/(1:N); g2 = (e2)/(1:N);
%% Plot
figure(1); clf;
h = plot(x,e1,'-',x,e2,'-'); hold on;
plot(x,x*g1,'k--',x,x*g2,'k--');
grid on; box on;
xlabel('times ifft(fft(x)) applied');
ylabel('max numerical roundoff error / eps');
title(sprintf('g1 = %.2f, g2 = %.2f',g1,g2));
legend(h,'abs(...)','real(...)','location','northwest');
```

Listing A.1: Comparison of using `abs()` and `real()` for rounding after taking the `ifft` of an `fft` of a real signal. The output it not exactly the same as the input due to numerical roundoff errors (it may even be complex). These errors cancel out better when using abs which can be seen well when applying `ifft(fft(x))` multiple times.

```
N = 64; % Number of sampled points
t = linspace(0,2*pi,N); % time axis / s
f = 2; % Frequency / Hz
y = @(x) sin(2*pi*f*x) .* exp(-x); % Signal
% Exact Fourier transfrom of the signal
Y = @(v) (2*f*pi) ./ (4*f.^2*pi^2 - (1i-v*2*pi).^2);
%% Construct frequency axis
dt = t(2)-t(1); % Time sampling
v = linspace(0,1-1/N,N); v = v-v(floor(N/2)+1); % Frequency indices
v = v/dt; % Frequency axis / Hz
%% Calculate fourier transforms
fast = fftshift(fft(y(t)));a
naive = sum(bsxfun(@times,y(t),exp(2 * pi * 1i .* v' .* (t-t(1)))),2)';
exact = Y(v)/dt; % Divide by dt to match fft
%% Plot
% Dense axes for nice plots
t_dense = linspace(0,2*pi,1e3); v_dense = linspace(min(v),max(v),1e3);
figure(1); clf;
```

```
subplot(2,1,1);
h = plot(t_dense,y(t_dense),'k-',t,y(t),'o');
grid on; box on; xlabel('t'); ylabel('y'); xlim(minmax(t));
title('Signal'); legend(h,'Signal','Sampled points');
subplot(2,1,2);
h = plot(v,abs(fast),'x',v,abs(naive),'o',v,abs(exact),'d'); hold on;
plot(v_dense,abs(Y(v_dense)/dt),'k-');
grid on; box on; xlabel('f'); ylabel('FT\{ y \}');
title('Fourier Transform'); legend(h,'FFT','Naive','Exact');
```

Listing A.2: Comparison of the fft and naïve approach to calculating the Fourier Transform. The signal is a single frequency oscillation starting at t=0 and decaying exponentially with time constant of 1. The exact value has been obtained with Mathematica. The difference between the `fft` and the exact value arises from finite sampling frequency and time.

```
function [k,yf] = fft_simple(dt,y,window)
%% [k,yf] = fft_simple(dt,y,window)
% Takes a fourier transform of a signal y with time point spacing dt
% INPUT
%  dt - time axis spacing (1/fs where fs is the sampling frequency)
%  y - signal to fourier transform
%  window - type of window to apply:
%           'boxcar' - no window
%           'hann' - default, Hanning window
%           'flattop' - useful for precise magnitude estimation
% OUTPUT
%  k - wavevectors (frequency axis)
%  yf - single-sided fourier transform of y
% Written by Marcin Konowalczyk
% Timmel Group @ Oxford University
y = y(:)'; % Make sure y is a row vector
n = numel(y);
%% Default to hann window
if nargin < 3 || isempty(window), window = 'hann'; end
%% Window signal
switch window
    case 'hann'
        w = hann(n)' / 0.5;
    case 'flattop'
        w = flattopwin(n)' / 0.21557895;
    case {'boxcar','none'}
        w = ones(1,n) / 1; % null window
end
%% Calcualte single-sided FFT
yf = fft(y.*w)./n; % Scaled FFT
yf = abs(yf); % Amplitude spectrum
yf = yf(1:ceil(n/2)); % Take only half of the psd
if mod(x,2) % Nyquist frequency does not exist
    k = linspace(0,1/2-1/n/2,ceil(n/2))./dt;
else
    k = linspace(0,1/2,n/2)./dt;
end
end
```

Listing A.3: Single sided, windowed FFT and an accompanying frequency axis.

A.2 Random search optimisation

```
function [pf,fval,path] = random_search(f,p0,tol,step_size)
```

```
if nargin < 3 || isempty(tol), tol = 1e-4; end
if nargin < 4 || isempty(step_size), step_size = 1; end
stall = 0; path = p0; % Initialise
pf = p0; fval = f(p0); % Starting point p0
for j = 1:1000 % Limit number of iterations to 1e3
    dp = step_size.*randn(1,2);
    pnew = pf + dp; fnew = f(pnew);
        if fnew < fval
            pf = pnew; fval = fnew; stall = 0;
            path = [path; pf];  %#ok<AGROW>
        elseif stall >= 15 % Standing in one place for too long
            stall = 0; step_size = step_size/2;
        else
            stall = stall + 1;
        end
        if step_size < tol; break; end
    end
end
```

Listing A.4: Random search optimisation function.

A.3 Random coloured noise

```
function X = randnd(beta,varargin)
%% X = randnd(beta,varargin)
% This function is a "decorator" for the Matlab in-build 'randn' function.
% It takes an additional first argument (beta) which shapes the spectral
% characteristic of the data (in all dimensions) as f^beta. The output
     data
% is scaled to zero mean and unit standard deviation.
% Written by Marcin Konowalczyk
% Timmel Group @ Oxford University
%% Parse the input
narginchk(0,Inf); nargoutchk(0,1);
if nargin < 2 || isempty(beta); beta = 0; end % Default to white noise
assert(isnumeric(beta) && isequal(size(beta),[1 1]),..
    '''beta'' must be a number');
assert(-6 <= beta && beta <= 6,...
    '''beta'' out of range'); % Put on reasonable bounds
%% Generate N-dimensional white noise with 'randn'
X = randn(varargin{:});
if isempty(X); return; end;
%% Squeeze prevents an error if X has more than one leading
% singleton dimension. This is a slight deviation from the
% pure functionality of 'randn'
X = squeeze(X);
%% Return if white noise is requested
if beta == 0; return; end;
%% Generate corresponding N-dimensional matrix of multipliers
N = size(X);
% Create matrix of multipliers (M) of X in the frequency domain
M = [];
for j = 1:length(N)
    n = N(j);
    if (rem(n,2)~=0) % if n is odd
        % Nyquist frequency bin does not show up in odd-numbered fft
        k = ifftshift(-(n-1)/2:(n-1)/2);
    else
        k = ifftshift(-n/2:n/2-1);
    end
    % Spectral multipliers
```

```
    m = (k.^2)';
    if isempty(M);
        M = m;
    else
        % Create the permutation vector
        M_perm = circshift(1:length(size(M))+1,[0 1]);
        % Permute a singleton dimension to the beginning of M
        M = permute(M,M_perm);
        % Add m along the first dimension of M
        M = bsxfun(@plus,M,m);
    end
end
%% Reverse M to match X (new dimensions were being added from the left)
M = permute(M,length(size(M)):-1:1);
%% Shape the amplitude multipliers by beta/4 which
% corresponds to shaping the power by beta
M = M.^(beta/4);
%% Set the DC component to zero
M(1,1) = 0;
%% Multiply X by M in frequency domain
Xstd = std(X(:));
Xmean = mean(X(:));
X = real(ifftn(fftn(X).*M));
%% Force zero mean unity standard deviation
X = X - mean(X(:));
X = X./std(X(:));
%% Restore standard deviation and mean from before the spectral shaping
% This ensures the random sample from randn is truly random. After all, if
% the mean was always exactly zero it would not be all that random.
X = X + Xmean;
X = X.*Xstd;
end
```

Listing A.5: Random coloured N-dimentional noise. Published on Mathworks[51]

A.4 Eigenvalue optimisation

```
N = 50; % Size of A
A = rand(N,N); A = A*A'; % Random positive semidefinite matrix
%% Eigenvalues with eig
tic, [V,D] = eig(A); toc
D_eig = diag(D)';
%% Eigenvalues by optimisation
options = optimset('Display','none');%,'MaxIter',1e4,'MaxFunEvals',1e4);
cm = plasma(N);
tic
for j = 1:size(A,1)
    if j > 1, lossfun = @(L) lossfun(L) + log10( (L-d(j-1)).^-2 );
    else, lossfun = @(L) log10( det(A - (L)*eye(size(A,1))).^2 );
    end
    d(j) = fminsearch(lossfun,0,options);
end
toc
D_optim = sort(d);
%% Display
delta_D = abs(D_eig-D_optim);
fprintf('median delta = %.1f eps\n',median(delta_D./eps));
fprintf('max delta = %.1f eps\n',max(delta_D./eps));
```

Listing A.6: Eigenvalues of a large random 50×50 matrix by optimisation.

A.5 Subfunctions for the resonator alignment simulation

```
function M = make_mirror(g,s)
l = 2; N = 1e2; % Fixed cavity length and number of points in the mirror
%% Make the mirror
if g == 1 % Special case
    X = zeros(1,2*N); Y = linspace(-0.5,0.5,2*N);
else
    if g < 1, xs = 1; % Convex mirror
    elseif g > 1, g = 2-g; xs = -1; % Concave mirror
    end
    % x-range to get a small mirror
    r2 = 1/8*((8*l)/(1-g)-sqrt(-16+(64*l^2)/(1-g)^2));
    x = linspace(0,r2,N); y = sqrt(x.*(x-g*x-2*l)./(g-1));
    Y = [fliplr(y) -y]; X = xs*[fliplr(x) x];
    X(N) = []; Y(N) = []; % Remove duplicate middle points
end
M = X+1i*Y;
%% Linearlize
t = [0 cumsum(abs(diff(M)))]; t = t/t(end); % Piecewise arclength
ti = linspace(0,1,N);
M = interp1(t,X,ti,'spline') + 1i*interp1(t,Y,ti,'spline');
%% Shift mirrors to correct positions
if s == 1, M = M-l/2; % Make M1
elseif s == 2, M = -conj(M)+l/2; % Make M2
else, error('Invalid mirror selector');
end
end
```

Listing A.7: Mirror maker for the raytracing simulation

```
function [v,isterminal,direction] = eventfun(t,yc,M1,M2)
isterminal = true; % Stop on event
direction = -1; % Approach from positive v
v = 1.5-abs(yc(1)); % Hit a killplane outside of the resonator
if v < 0, return; end
if real(yc(1))<0 % Bounce of M1
    [v,~] = bouncefun(yc,M1);
else % Bounce of M2
    [v,~] = bouncefun(yc,M2);
end
end
```

Listing A.8: Raytracing event handler

```
function [v,grad] = bouncefun(yc,M)
yc = yc(1); N = numel(M);
D = abs(M-yc); [~,I] = sort(D); I1 = I(1); % Find closest point
%% Deal with special cases of the other point (edges), and then find the
if I1 == 1
    I2 = 2;
elseif I1 == numel(M)
    I2 = numel(M)-1;
elseif D(I(1)+1)<D(I(1)-1)
    I2 = I1+1;
else
    I2 = I1-1;
end
Mp = M([I1 I2]); % Two points in M closest to yc
%% Absolute distance from the line defined by Mp to yc
v = (imag(Mp(2))-imag(Mp(1)))*real(yc)-...
    (real(Mp(2))-real(Mp(1)))*imag(yc)+...
```

```
    real(Mp(2))*imag(Mp(1))-imag(Mp(2))*real(Mp(1));
v = abs(v)./abs(Mp(1)-Mp(2));
%% Check whether the point has passed the mirror.
if abs(yc) > abs(Mp(1)) && abs(yc-Mp(1)) <= 2*abs(Mp(2)-Mp(1))
    v = -v;
end
%% Also calculate the gradient for the impact mapping
grad = (imag(Mp(2))-imag(Mp(1)))/(real(Mp(2))-real(Mp(1)));
end
```

Listing A.9: Ray reflection calculation

```
function y = impactmap(yc,M1,M2)
if real(yc(1))<0 % Impact on M1
    [~,grad] = bouncefun(yc(1),M1);
else % Impact on M2
    [~,grad] = bouncefun(yc(1),M2);
end
%% Angle of the gradient + 1mrad of randomness
theta = atan2(grad,1) + randn*1e-3;
% Rotate such that only the vertical component of the velocity
% changes on bounce. Change it, and rotate back.
y = [yc(1) exp(1i*theta)*conj(exp(1i*-theta)*yc(2))];
end
```

Listing A.10: Ray impact mapping

A.6 Polynomial fitting

```
function [p,V] = polyfit_fast(x,y,n)
%% [p,V] = polyfit_fast(x,y,n)
% Fit n'th degree polynomial to x,y data without any condition checks
% Written by Marcin Konowalczyk
% Timmel Group @ Oxford University
y = y(:);
%% Construct Vandermonde matrix
if isvector(x)
    x = x(:); V = zeros(length(x),n+1,'like',x);
    V(:,n+1) = 1;
    for j = n:-1:1
        V(:,j) = x.*V(:,j+1);
    end
else % Assume x *is* the Vandermode matrix
    V = x;
end
%% Solve least squares problem
p = V\y; p = p.';
end
```

Listing A.11: Fast polynomial fitting. The input is assumed to be well conditioned.

```
function [p,V,W] = polyfit_weighted(x,y,w,n)
%% [p,V,W] = polyfit_weighted(x,y,w,n)
% Fit n'th degree polynomial to x,y data without any condition checks
% Written by Marcin Konowalczyk
% Timmel Group @ Oxford University
y = y(:);
%% Construct Vandermonde matrix
if isvector(x)
    x = x(:); nx = length(x); V = zeros(nx,n+1,'like',x);
```

```
    V(:,n+1) = 1;
    for j = n:-1:1
       V(:,j) = x.*V(:,j+1);
    end
else
    % Assume x *is* a Vandermode matrix
    V = x;
    nx = size(V,1);
end
%% Construct the weights matrix
% (if the input has not already been supplied as one such matrix)
if isvector(w)
    w = w(:);
    w = w/sum(w);
    if nx > 80 || nargin > 1
        W = spdiags(w,0,nx,nx);
    else
        W = diag(w);
    end
else % Assume w *is* a weights matrix
    W = w;
end
%% Solve weighted least squares problem
p = (W*V)\(W*y); p = p.';
end
```

Listing A.12: Fast weighted polynomial fitting. The input is assumed to be well conditioned.

```
function y = polyval_fast(p,x)
%% y = polyval_fast(p,x)
% Evaluate n'th degree polynomial to x,y data without any condition checks
% Written by Marcin Konowalczyk
% Based on matlab's polyfit
% Timmel Group @ Oxford University
%% Parse input
iscolumn_x = iscolumn(x);
if ~iscolumn_x, x = x'; end % Make x a column vector
p = p(:); n = numel(p)-1;
%% Construct Vandermonde matrix
V = zeros(length(x),n+1,'like',x);
V(:,n+1) = 1;
for j = n:-1:1
   V(:,j) = x.*V(:,j+1);
end
%% Evaluate polynomial
y = V*p;
if ~iscolumn_x, y = y'; end % Ensure the output is a column
end
```

Listing A.13: Fast polynomial evaluation. The input is assumed to be well conditioned.

A.7 Model functions

```
function y = sigmoid_step(x,alpha,beta,center,slope)
slope = abs(slope); % Sign of slope determined by 'alpha' and 'beta'
if alpha == beta
    y = ones(size(x)).*alpha;
    return
elseif alpha < beta
```

```
    y = (beta-alpha)./(1+exp(-(x-center)*slope*4)) + alpha;
    else % alpha > beta
    y = -(alpha-beta)./(1+exp(-(x-center)*slope*4)) + alpha;
end
end
```

Listing A.14: Sigmoid from `alpha` to `beta`. The halfway point has a derivative of `(beta-alpha)*slope` at `center`.

```
function y = exponential_step(x,tau,alpha,beta,center)
y = (alpha-beta) .* exp(-(x-center)./abs(tau)) + beta;
y(x<=center) = alpha;
end
```

Listing A.15: Exponential function decaying/growing from `alpha` to `beta` with a time constant `tau`, for `x>center`. This function has a discontinuous derivative at `x=0`.

```
function y = gaussian_lineshape(x,center,width,height)
l42 = 2.772588722239781; % 4.*log(2)
y = exp( -l42.*((x-center)./width).^2 ) .* height;
end
```

Listing A.16: Generalised form of a Gaussian lineshape. `width` corresponds to full width at half maximum (FWHM).

```
function y = lorentzian_lineshape(x,center,width,height)
y = (1./(1+((x-center)/width*2).^2)) .* height;
end
```

Listing A.17: Generalised form of a Lorentzian lineshape. `width` corresponds to full width at half maximum (FWHM). This lineshape is also known as 'Cauchy lineshape' or 'Cauchy distribution'. These names are mostly used in statistics.

```
function y = asymmetric_gaussian_lineshape(x,...
    center,width_left,width_right,height)
l42 = 2.772588722239781; % 4.*log(2)
x = x-center;
y1 = exp( -l42.*(x./width_left).^2 ) .* height;
y2 = exp( -l42.*(x./width_right).^2 ) .* height;
y = y1.*(x<0) + y2.*(x>=0);
end
```

Listing A.18: Asymmetric piecewise Gaussian lineshape. Single lineshape where the left and right halves come from different Gaussians. The second derivative of this lineshape at `x=center` is discontinuous.

```
function y = havercosine_lineshape(x,center,width,height)
region = x>=center-width & x<=center+width;
y = zeros(size(x));
y(region) = ( 1 + cos( (x(region)-center)/width*pi ) ) / 2 * height;
end
```

Listing A.19: Generalised form of a Havercosine lineshape. `width` corresponds to full width at half maximum (FWHM). This lineshape is also known as 'raised cosine' lineshape and it has uses mainly in statistics. The second derivative of the Havercosine lineshape is discontinuous outside of `region`.

```
function y = sech_lineshape(x,center,width,height)
l22s3 = 2.633915793849633; % 2*log(2+sqrt(3))
y = sech((x-center)/width * l22s3) .* (height);
```

```
end
```

Listing A.20: Generalised form of a Hyperbolic secant lineshape. `width` corresponds to full width at half maximum (FWHM).

```
function y = exponential_step_wave(x,tau,alpha,beta,dc,period,phi)
%% Set some default parameters
if nargin < 7 || isempty(phi), phi = 0; end
if nargin < 6 || isempty(period), period = 1; end
if nargin < 5 || isempty(dc), dc = 0; end
%% Make y
carrier = sin(2*pi*x/period + phi) > 0; % Carrier square wave
periodic_x = mod(x+phi/pi*period/2,period/2); % Periodic x axis
y = (alpha-beta) .* exp(-periodic_x./abs(tau)) + beta;
y = (carrier.*y) + (~carrier.*(-y)) + dc;
end
```

Listing A.21: Periodic square-wave-like function where each step decays/grows from alpha to beta with a time constant of tau. The derivative of this function is discontinuous (Discontinuities are periodic in `period/4`).

A.8 Additional code for ModMARY

```
%% Pick a Lotentzian
%L = @(x) x; % Unit transfer function
%L = @(x) 1./(1+x.^2); % Unit Lorentzian
L = @(x) 1./(1+x.^2) - 0.02./(0.1+x.^2) - 0.8; % L with small LFE bump
%% Field axes
B0 = linspace(-7,7,1e3+1); % Field axis
b01 = 2; b02 = 0.1; bm = 1/2;
t = linspace(0,1,1e3+1); t(end) = [];
b1 = ones(size(t)).*b01 + bm*sin(t*2*pi*20);
b2 = ones(size(t)).*b02 + bm*sin(t*2*pi*20);
%% Calculate at different offet points
y0 = b1 - mean(b1) + 1e-8*randn(size(t));
y1 = L(b1) - mean(L(b1)) + 1e-8*randn(size(t));
y2 = L(b2) - mean(L(b2)) + 1e-8*randn(size(t));
%% Calculate spectra of the signals
[k,yf0] = fft_simple(t(2)-t(1),y0,'hann');
[~,yf1] = fft_simple(t(2)-t(1),y1,'hann');
[~,yf2] = fft_simple(t(2)-t(1),y2,'hann');
%% Plot
figure(1); clf; hold on;
plot(k,20*log10(yf0/bm*2),'-');
plot(k,20*log10(yf1/bm*2)-200);
plot(k,20*log10(yf2/bm*2)-400);
xlim([0 500]); ylim([-250-400 0]);
grid on; box on;
xlabel('f / Hz'); ylabel('power / dB');
legend('baseband','bo = 2','bo = 0.1');
```

Listing A.22: Harmonic distortion of a single tone by a Lorentzian. This code uses `fft_simple` from listing A.3.

The convolution result (Equations 3.35 and 3.37) can be used to derive an exact analytical expression for the modulated signal. This, however, can be done only for simple cases like, for example, a unit Lorentzian function, given by:

$$M_L(x) = \frac{1}{1+x^2} \tag{A.1}$$

Substituting M_L for M in Equation 3.35 setting $h=1$ and integrating, gives the analytical solution:

$$S(b_o) = \sqrt{2}e^{-i\theta}\frac{\sin(\mathcal{A}) - b_o\cos(\mathcal{A})}{b_m^2\sqrt[4]{4b_o^2+\left(1-b_o^2+bm^2\right)^2}} \tag{A.2}$$

where $\mathcal{A}$ is a function of b_o and b_m, given by:

$$\mathcal{A}(b_o,b_m) = \frac{1}{2}\text{arctan2}\left(1-b_o^2+b_m^2,,2b_o\right) \tag{A.3}$$

The function arctan2 is a 2-argument arctangent which preserves the quadrant information of its input.

The log local error plots in Figure 3.2 have been generated by numerically computing errors in progressive Taylor expansions over a small region of a Lorentzian lineshape $L(x)=\left(1-x^2\right)^{-1}$. n^{th} derivative of L has a form $\mathcal{P}_n\times(1-x^2)^{-(n+1)}$ where $\mathcal{P}_n$ is a polynomial of n^{th} degree. The matrix of coefficients of $\mathcal{P}_n$, for the first 10 derivatives (0^{th} through 9^{th}) is:

$$\mathcal{P} = \begin{Bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 6 & 0 & -2 \\ 0 & 0 & 0 & 0 & 0 & 0 & -24 & 0 & 24 & 0 \\ 0 & 0 & 0 & 0 & 0 & 120 & 0 & -240 & 0 & 24 \\ 0 & 0 & 0 & 0 & -720 & 0 & 2400 & 0 & -720 & 0 \\ 0 & 0 & 0 & 5040 & 0 & -25200 & 0 & 15120 & 0 & -720 \\ 0 & 0 & -40320 & 0 & 282240 & 0 & -282240 & 0 & 40320 & 0 \\ 0 & 362880 & 0 & -3386880 & 0 & 5080320 & 0 & -1451520 & 0 & 40320 \\ -3628800 & 0 & 43545600 & 0 & -91445760 & 0 & 43545600 & 0 & -3628800 & 0 \end{Bmatrix} \tag{A.4}$$

```
%% Lorentzian and its derivatives
P = ⟨10 × 10 matrix⟩ from Equation A.4; % Matrix of polynomial coefficients
for n = 1:size(P,1)
    Ds{n} = @(x) polyval(P(n,:),x)./(1+x.^2).^n;
end
bm = 0.1; % Modulation amplitude
bos = linspace(-10,10,1e3); % Values of bo
N = 1e3; % Number of x points to evaluate the approximant over
LLE = zeros(numel(bos),numel(Ds));
for boi = 1:numel(bos)
    bo = bos(boi); % Current
    x = linspace(bo-bm,bo+bm,N); % Local x axis
    % Exact - just evaluate the Lorentzian
    y = Ds{1}(x);
    % Taylor - Evaluate the Taylor expansion up to nth order
    yt = zeros(size(y));
    for n = 1:numel(Ds)
        yt = yt + Ds{n}(bo).*(x-bo).^(n-1)./factorial(n-1);
        LLE(boi,n) = log10(max(abs(y-yt)));
    end
end
%% Plot
cm = summer(numel(Ds)+1);
figure(1); clf;
h = plot(bos,log10(Ds{1}(bos)*eps),'k--');
hold on;
for n = 1:numel(Ds)
    plot(bos,LLE(:,n),'color',cm(n,:));
end
uistack(h,'top');
grid on; box on;
xlim([min(bos) max(bos)]);
title('Progressive local Taylor expansion error');
xlabel('x'); ylabel('log(max(abs(y-yt))) locally');
```

Listing A.23: Log local error estimate for a Lorentzian lineshape.

```
function K = MMcore_getScaledK(h,N,bm)
x = linspace(-1,1,N); % Sample unit interval
Kref = sqrt(2)*sin(acos(x))/pi; % h=1 kernel
C = 1/sum(Kref)/sqrt(2)*bm; % Scaling factor
K = C*sqrt(2)*sin(h*acos(x))/h/pi; % Scaled h=... kernel
end
```

Listing A.24: Function used to calculate a correctly scaled version of the modulation kernel `K`. The scaling is needed to make the discrete version of the convolution give correct magnitude of the signal. `h` is the harmonic constant, `N` is the number of points in the desired kernel `K`, and `bm` is the modulation amplitude.

```
function [dydx,dx] = MM_numerical_derivative(x,y)
x = x(:); y = y(:); n = numel(x); dx = mean(diff(x));
dy = zeros(size(y));
% Forward and backward differences at the edges
dy(1) = y(2) - y(1); dy(n) = y(n) - y(n-1);
% Center difference everywhere else
dy(2:n-1) = ( y(3:n)-y(1:n-2) )/2;
dydx = dy/dx;
end
```

Listing A.25: Function used to take a numerical derivative of the transfer function to be then convolved with the modulation kernel. The input is specified as `x` and `y` vectors. `x` is assumed to be monotonic.

```
function [x,yc,bm] = MM_convolve(fun,range,bm,h,N)
%% [x,yc,bm] = MM_convolve(fun,range,bm,h,N)
% Convolve a funciton specified by fun over the range with a modulation
% kernel specified by modulaiton amplitude bm and detection harmonic h.
% Part of the ModMARY code
% Written by Marcin Konowalczyk
% Timmel Group @ Oxford University
%% Parse input
if nargin < 5 || isempty(N)
    N = 1e3+1;
else
    if ~mod(N,2), N = N + 1; end
end
bm = abs(bm); % Modulation amplitude must be positive
%% Get the x axis
range2 = [min(range) max(range)];
x = linspace(range2(1),range2(2),N);
dx = x(2)-x(1); % x spacing
%% Early escape for special case
if bm == 0, yc = zeros(size(x)); return; end
% Pad x on either side to remove the edges where the convolution is invaid
pad_N = ceil(bm/dx);
lower_pad = linspace(range2(1)-pad_N*dx,range2(1)-dx,pad_N);
upper_pad = linspace(range2(2)+dx,range2(2)+pad_N*dx,pad_N);
x2 = [lower_pad x upper_pad];
%% Do the convolution
% Evaluate fun at x
y = fun(x2);
[dy,~] = MM_numerical_derivative(x2,y,dx);
%dy = gradient(y)/dx;
%% Get the convolution kernel. Coersce size to the closest n of points
K_N = 2*pad_N+1;
```

```
bm = pad_N*dx; % Actuall bm used
K = MMcore_getScaledK(h,K_N,bm);
%% Convolve
yc = conv(dy,K,'same');
yc = yc(pad_N+1:end-pad_N);
%% Re-interpolate onto the range if needed
if numel(range) > 2
    F = griddedInterpolant(x(:),yc(:),'linear');
    yc = F(range);
    x = range;
end
```

Listing A.26: Function used to perform the convolution of the transfer function model M specified by function `fun` over the `range`. The modulation kernel is specified by modulation amplitude `bm`. The h^{th} harmonic of `fun` is therefore simulated. The convolution is done numerically with `fun` evaluated over `N` points. The outputs are: `x` – field axis used for the convolution, `yc` convolved signal, and `bm` – actual modulation depth used (coerced to integer multiple of `x` spacing)

A.9 Allan deviation

```
function [a,e,nt,tau] = allan2(Y,tau,pc,np)
%% [a,e,nt,tau] = allan2(Y,tau,pc,np)
% Calculate Allan deviation of the vector of values in 'Y', with integer
% segment lengths of 'tau'. If 'pc' is true, a phase cycling over a range
% of 'np' possible initial phases of the segments is performed.
%
% Written by Marcin Konowalczyk
% Timmel Group @ Oxford University
%% Parse the input
% Pick tau
if nargin < 2 || isempty(tau) % Pick around 100 points
    tau = fix(10.^linspace(log10(1),log10(floor(length(Y)/2)),100));
elseif isequal(size(tau),[1 1]) % Pick 'tau' points
    tau = fix(10.^linspace(0,log10(floor(length(Y)/3)),fix(tau)));
end
tau = sort(unique(tau(:))); % taus shoudl be unique
%% Default parameters
if nargin < 3 || isempty(pc), pc = false; end
if nargin < 4 || isempty(np), np = 10; end
Y = Y(:);
%% Go through all the values of tau
a = zeros(1,numel(tau)); e = zeros(1,numel(tau));
nt = zeros(1,numel(tau));
for j = 1:numel(tau)
    n = tau(j); % Number of points per sample
    %% Pick phase cycling mode
    if isinf(np)
        % Cycle though all the phases in the sample
        phases = unique(fix(1:n));
    elseif pc && np > 1
        % Pick np phases uniformly distributed thoughout the sample
        if n <= 2*np
            pmax = n;
        else
            pmax = n-fix(n/np);
        end
        phases = unique(fix(linspace(1,pmax,np)));
    else % Don't phase cycle
        phases = 1;
```

```
    end
    ns = floor(length(Y)/n); % Number of n-point samples
    if length(phases) > 1
        if ns > 2
            ns = ns-1; % Decrease number of segments for phase cycling
        else
            phases = 1; % Dont phase cycle if only two samples
        end
    end
    %% Go through all the phases
    D = NaN(ns-1,length(phases)); % initialise diffs
    for jj = 1:length(phases) % for each possible phase of the sample
        p = phases(jj); % jj'th phase
        y = Y((1:n*ns) + (p-1)); % trim Y
        y = reshape(y,n,ns); % reshape by samples
        ys = sum(y,1); % take the sum across each sample
        D(:,jj) = diff(ys)'/n;
    end
    D = mean(D,2); % Average of all the phases
    %% Calculate allan deviation and associated error
    a(j) = sqrt(0.5.*(mean(D.^2))); % Allan deviation
    e(j) = a(j)/sqrt(ns+1); % Estimated error in Allan deviation
    nt(j) = ns; % Number of segments
end
end
```

Listing A.27: Allan deviation. Published on MathWorks [154].

A.10 Lorentzian fitting

```
function [pf,dp,yfit,npf] = fit_Lorentzian_Single(x,y,dy,varargin)
%% [pf,dp,yfit,npf] = fit_Lorentzian_Single(x,y,dy,method)
% Fit a lorentzian lineshape to x,y data
% y = pf(1)*pf(2)^2./(x.^2+pf(2)^2)-pf(1);
%
%  pf   - fit parameters
%  dp   - uncertainty in 'pf'
%  yfit - fit function
%  npf  - bootstrapped parameter values
%
% Written by Marcin Konowalczyk
% Timmel Group @ Oxford University
%% Parse input
methods = {'hessian','bootstrap','none'};
spec = {...
    'plot', false, '';
    'confidence', 'hessian', @(y) any(cellfun(@(x) strcmpi(y,x),methods));
    'nbstrap', 1e2,''};
opt = varargin2opt_tiny(varargin,spec);
x = x(:); y = y(:);
assert(numel(x) == numel(y),...
    '''x'' and ''y'' must have the same number of elements');
%% Construct weights based on the residuals
if nargin < 3 || isempty(dy)
    w = ones(size(y)); % Null weights
else
    dy = dy(:);
    assert(numel(y) == numel(dy),...
        '''y'' and ''dy'' must have the same number of elements');
    good = isnan(dy); bad = isinf(dy);
    mdy = max(dy);
```

```
    dy(good | bad) = mdy;
    w = mdy-dy; w = w./max(w);
    w(bad) = 0; w(good) = 1;
end
%% Fit
p0  = [10 100];
f = @(p) fitfun(x,y,w,p,opt.plot);
plb = [-Inf 0];
pub = [Inf Inf];
fc_options = optimoptions('fmincon','Display','none');
warning('off','MATLAB:rankDeficientMatrix');
[pf,~,~,~,~,~,H] = fmincon(f,p0,[],[],[],[],plb,pub,[],fc_options);
warning('on','MATLAB:rankDeficientMatrix');
yfit = @(x) L1(x,pf(1),pf(2));
R = yfit(x)-y;
if strcmpi(opt.confidence,'hessian')
    %% Use the apporx. curvature of the fitfuncton to estimate uncertainty
    s = (1/(numel(x)-length(pf)))*sum(R(:).^2);
    dp = diag(sqrt(abs(s./H)))';
    dp = [dp; dp]; % Same lower and upper prediction abounds
    npf = [];
elseif strcmpi(opt.confidence,'bootstrap')
    %% Use bootstrap resampling to estimate confidence interval
    fit = yfit(x);
    r = y-fit; % Residuals
    npf = zeros(opt.nbstrap,numel(pf)); % Preallocate
    % Bootstrap
    W = warning('off','MATLAB:nearlySingularMatrix');
    for nbi = 1:opt.nbstrap
        % Resample residuals
        %nr = resample_residuals(r,bl);
        [nr,idx] = datasample(r,numel(r),'Replace',true);
        nw = w(idx);
        % Refit best fit data + bootstraped residuals
        f = @(p) fitfun(x,fit+nr,nw,p,false);
        ps = fmincon(f,pf,[],[],[],[],plb,pub,[],fc_options);
        npf(nbi,:) = ps;
    end
    warning(W);
    % Analyse the bootstrapped smaples
    dp = zeros(2,numel(pf));
    for nbi = 1:numel(pf)
        [~,dp(1,nbi),dp(2,nbi)] = mul(npf(:,nbi));
    end
elseif strcmpi(opt.confidence,'none')
    %% No conficence bounds estimation
    dp = []; npf = [];
    end
end

function y = L1(x,a,g)
%% Lorentzian lineshape
y = a*g^2./(x.^2+g^2)-a;
end

function diff = fitfun(x,y,w,p,plt)
%% Fit function
F = L1(x,p(1),p(2));
R = F(:)-y(:);
if plt
    x2 = linspace(1.5*min(x),1.5*max(x),100);
    F2 = L(x2,p(1),p(2));
```

```
    plot(x,y,x2,F2); grid on; drawnow;
end
diff = sum((R.^2).*w(:));
end
```

Listing A.28: Single Lorentzian fit with uncertainty calculation.

```
function options = varargin2opt_tiny(va,specs)
%% options = varargin2opt_tiny(...)
% Tiny implementation of varargin2opt to be appended to other functions
% which may need it.
% SYNTAX
%  spec = {'plot', false, '';
%          'wildcard', [], 'any';
%          'magicfun', [1,2,3], @(x) size(x,2)==3};
%  opt = varargin2opt(varargin,spec);
% Written by Marcin Konowalczyk
% Timmel Group @ Oxford University
II = 'varargin2opt:invalidinput'; % Invalid input message ID
assert(isa(specs,'cell'),II,'Invalid specifications. Must be a cell');
assert(size(specs,2)==3,II,'Invalid specifications. Must be Nx2 or Nx3');
assert(isa(va,'cell'),II,'''varargin'' must be a cell');
assert(~mod(numel(va),2),II,'''varargin'' must have even length');
%% Parse options
p = inputParser; p.CaseSensitive = false;
p.FunctionName = mfilename; p.KeepUnmatched = true;
for Ni = 1:size(specs,1)
    [name, default, valid] = deal(specs{Ni,:});
    assert(ischar(name),II,...
    'Specified key of class ''%s'' as opposed to ''char''',class(name));
    $% Add parameter to the input parser
    if isa(valid,'function_handle') % Validator supplied
        addParameter(p,name,default,valid);
    elseif strcmpi(valid,'any') % Accept any parameter
        addParameter(p,name,default);
    elseif isempty(valid) % Input of the came class as default
        addParameter(p,name,default,@(x) isa(x,class(default)));
    else, error(II,'Specified validator invalid');
    end
end
%% Parse
parse(p,va{:}); options = p.Results;
%% Warn if some unmatched
if ~isempty(fields(p.Unmatched))
warning('varargin2opt:unmatchedOptions',...
    'Some options were not matched to any pattern');
end
options = orderfields(options);
end
```

Listing A.29: Input argument parser.

A.11 Liouville von Neumann simulation

```
function sig = liouville_von_neumann_fun(T,hfc,B0)
%% Options
opt.fixtrace = true; % Fix trace of rho to 1 after each timestep
opt.check = true; % Do sanity checks
%% Setup
% Basic spin matrices (in alpha/beta basis)
```

```
Ix = 0.5*[0  1; 1  0]; Iy = 0.5*[0 -1i;1i  0]; Iz = 0.5*[1  0; 0  -1];
Ie = eye(2); % Identiy matrix
%% Successive kronecker product of four elements
qk = @(x,y,z,q) kron(kron(kron(x,y),z),q);
tol = 10*eps; % Tolerance for asserts
%% Similarity transform to Singlet-Triplet basis
s2 = 1/sqrt(2); Pe = [1 0 0 0; 0 s2 s2 0; 0 -s2 s2 0; 0 0 0 1];
P = kron(kron(Pe,Ie),Ie); % Nuclei stay in alpha/beta basis
ST = @(A) P*A*P'; % Similarity transform to the ST basis
%% Check that P is unitary
if opt.check
    condition = ST(eye(16)) - eye(16) < tol;
    assert(all(condition(:)),'P is not unitary');
end
%% Spin matrices
% Components of SA,B,C,D vector operators
SAx=ST(qk(Ix,Ie,Ie,Ie)); SAy=ST(qk(Iy,Ie,Ie,Ie)); SAz=ST(qk(Iz,Ie,Ie,Ie));
SBx=ST(qk(Ie,Ix,Ie,Ie)); SBy=ST(qk(Ie,Iy,Ie,Ie)); SBz=ST(qk(Ie,Iz,Ie,Ie));
SCx=ST(qk(Ie,Ie,Ix,Ie)); SCy=ST(qk(Ie,Ie,Iy,Ie)); SCz=ST(qk(Ie,Ie,Iz,Ie));
SDx=ST(qk(Ie,Ie,Ie,Ix)); SDy=ST(qk(Ie,Ie,Ie,Iy)); SDz=ST(qk(Ie,Ie,Ie,Iz));
SASC = SAx*SCx + SAy*SCy + SAz*SCz; % A-C product operator
SASD = SAx*SDx + SAy*SDy + SAz*SDz; % A-D product operator
%% Make the Hamiltonian
gamma_n = 42.577e2; % Hydrogen nucleus gyromagnetic ratio in rad/s/G
gamma_e = 28024.951e2; % Electron gyromagnetic ratio in MHz/G
% Hyperfine frequencies of spin A
a1 = gamma_e * hfc(1); a2 = gamma_e * hfc(2);
omega0 = gamma_e*B0; omega0n = gamma_n*B0; % Electron and nuclear Larmor
% Electron and nuclear Zeeman
H_mag = omega0.*(SAz + SBz) + omega0n.*SCz + omega0n.*SDz;
H_hyperfine = a1.*SASC + a2.*SASD;
H = H_mag + H_hyperfine;
%% Define the projection operators, propagators and the observable
% Singlet projection operator in the aba basis
PS = 0.25*eye(16) - qk(Ix,Ix,Ie,Ie) - qk(Iy,Iy,Ie,Ie) - qk(Iz,Iz,Ie,Ie);
PS = ST(PS); % Project into ST basis
rh0 = PS./sum(trace(PS)); % Starts as singlet with no nuclear polarisation
PS = sparse(PS);
%% Sanity checks
if opt.check
    assert(abs(trace(rh0)-1) < tol,'Invalid starting point trace')
    condition = diag(rh0) >= -tol;
    assert(all(condition(:)),'Invalid sign of the starting point')
end
%% Calcualte the propagator matrices
%dt = mean(diff(T));
dt = min(1./[a1 a2 omega0 omega0n])/4; % Pick dt to satisfy Nyquist
Um = expm(-1i*H*dt); Up = expm(1i*H*dt);
rhp = rh0; % Spin density at previous timestep
%% Define the function to calculate the observable on the density matrix
%observable = @(rh) real(trace(PS*rh)); % Singlet character
observable = @(rh) real(sum(diag(PS*rh))); % Singlet character
%% Iterate through all the time points
Tsig = (0:ceil(max(T)./dt))*dt; % Timesteps at which the 'sig' is
     simulated
sig = NaN(1,length(Tsig)); % Prealocate the matrix for the signal
sig(1) = observable(rh0); % Observable of the initial density matrix
for i = 2:length(Tsig)
    % Propagate the density matrix by dt
    rht = Um*rhp*Up;
    if opt.fixtrace && opt.check; rht = rht/trace(rht); end % Fix roundof
    rhp = rht;
```

```
    if opt.check % Sanity checks
        % The trace of the density matrix must be equal to one
        % Error accummulates therefore multiply tol by i
        condition = abs(trace(rht) - 1) < i*tol;
        assert(condition,'Invalid trace %d', abs(trace(rht) - 1)/tol)
        % Diagonal elements of the density matrix must be positive
        condition = diag(rht) >= -tol;
        assert(all(condition(:)),'Invalid sign')
    end
    sig(i) = observable(rht); % Observable at time t
end
%% Interpolate from the simulation points to the desired time points
sig = interp1(Tsig,sig,T,'pchip');
end
```

Listing A.30: Liouville von Neumann simulation.

Appendix B

Additional figures

Any additional figures referenced throughout the text are listed in this appendix.

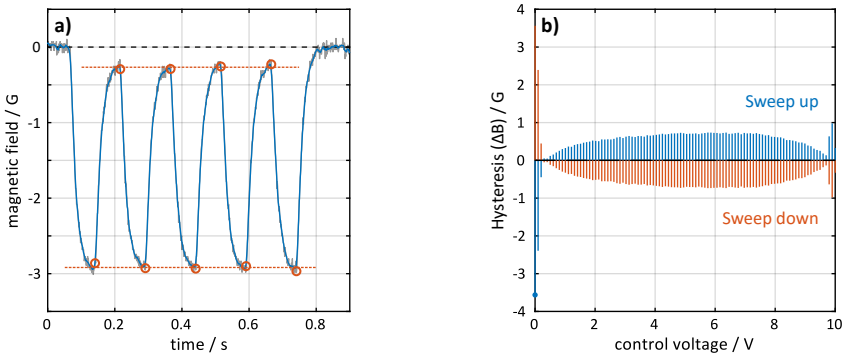


Figure B.1: a) Measurement of a typical field burst performed during the calibration procedure. 75 ms field step length. The upper and lower levels of the burst, as estimated by the calibration software, are shown by the dotted lines. In this case, the lower level does not quite reach 0 G and, hence, a little undercurrent will need to be applied. The value of the undercurrent will be found using the bisection algorithm. This procedure will be repeated for each burst magnitude calibrated. **b)** Magnetic field hysteresis measured by slowly sweeping the field up (blue) and down (red) over the entire range of the coils, and recording the value at each point. The mean of the two is then subtracted from the data akin to the centerline. The differences between this centerline and the data is the magnitude of the field 'lag' – hysteresis. Note that, at the time of this measurement, the coils were controlled with a 0–10 V control voltage. Since then this has been changed to –5/ + 5 V range, such that 0 V corresponds to 0 G (after constant offset calibration).

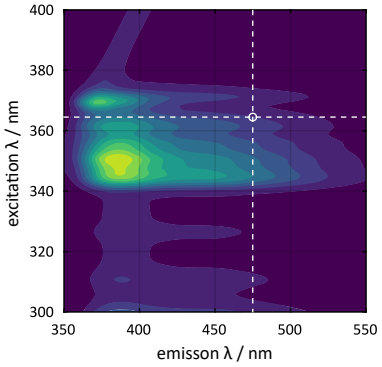


Figure B.2: 2D excitation/emission spectrum of Py /DCB. The peak at ca. (370 nm, 350 nm) corresponds to the pyrene monomer fluorescence. Its red-shifted shoulder is the fluorescence of the Py-DCB exciplex. The points indicated are the excitation (365 nm) and emission (475 nm) wavelengths used in the ModMARY experiment.

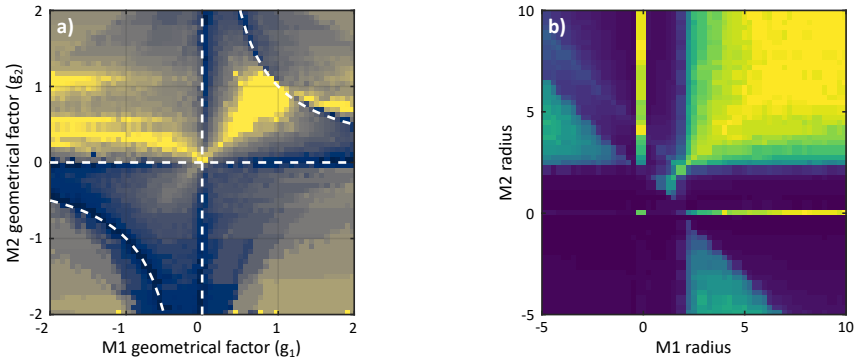


Figure B.3: Extension of the simulation shown in Listing 5.2 and Figure 5.3 **a)** Mode width as a function of the geometrical factor of M1 and M2. The mode width has been calculated by fitting the average number of ray bounces in the cavity across the range of starting points. The upper-right stable quadrant can be seen to have much wider cavity modes than the bottom left one. **b)** Surface from Figure 5.3a, but plotted as a function the M1/M2 *radius*, as opposed their geometrical factor.

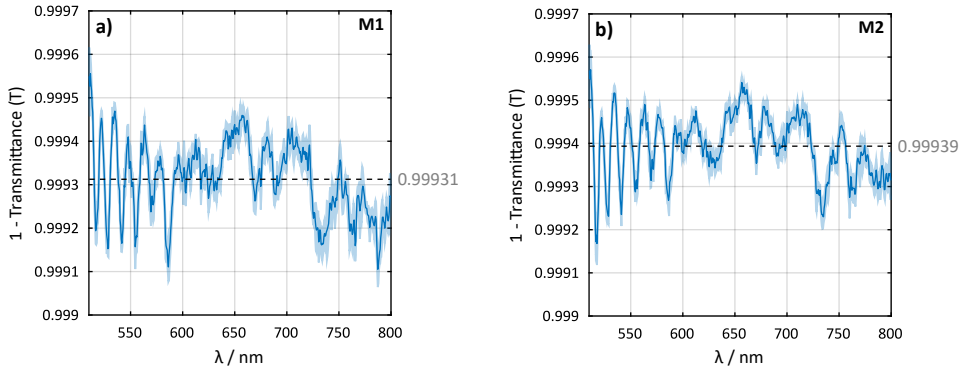


Figure B.4: Spectral transmittance, T, of the mirrors used on the BBCEAS experiment described in Chapter 5. If one assumes negligible scattering and mirror absorption contributions, the mirror reflectivity R is $R = 1-T$. **a)** The injection mirror M1 and **b)** the probe mirror M2 are shown as labelled in Figure 5.5. Compare this Figure with the spectral shape of the light intensity transmitted through the cavity in Figure 5.9 and typical CEF in Figure 5.4.

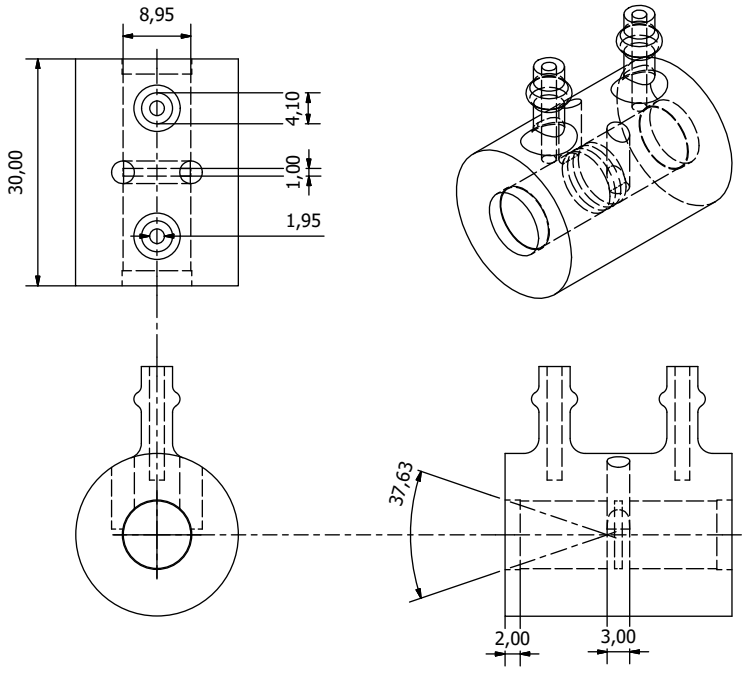


Figure B.5: Model of the Hellma QS 165-1-40 sample cell used. Model based on measurements of the actual cell. Only the approximate inside of the cell has been modelled. All the dimensions are given in millimetres. The drawing is in scale 1:1 on A4. The opening angle of the cell has been indicated.

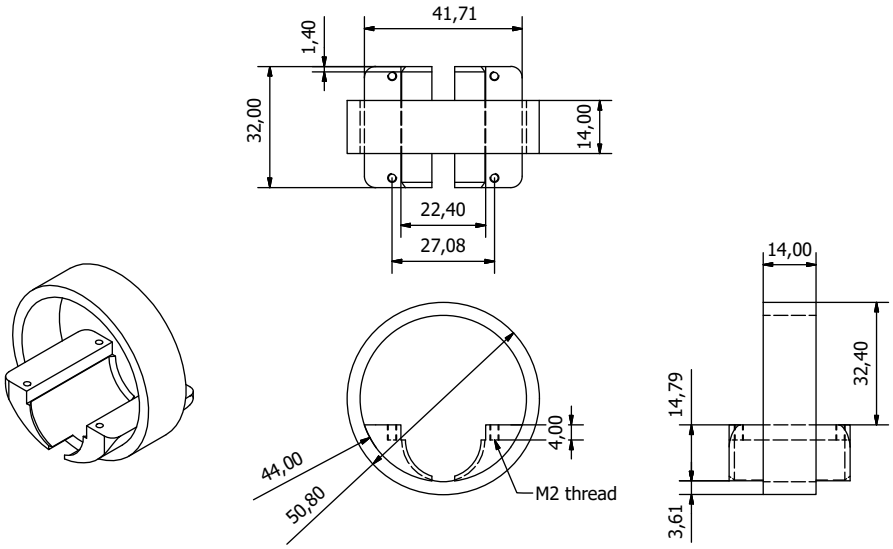


Figure B.6: Model of the cell mount designed to fit the cell shown in Figure B.5 inside of the Thorlabs GM200/M gimbal mount. All the dimensions are given in millimetres. The drawing is in scale 1:2 on A4. Printed at 0.07 mm layer height and 100 % infill in polylactic acid (PLA). Thread holes have been tapped after printing.

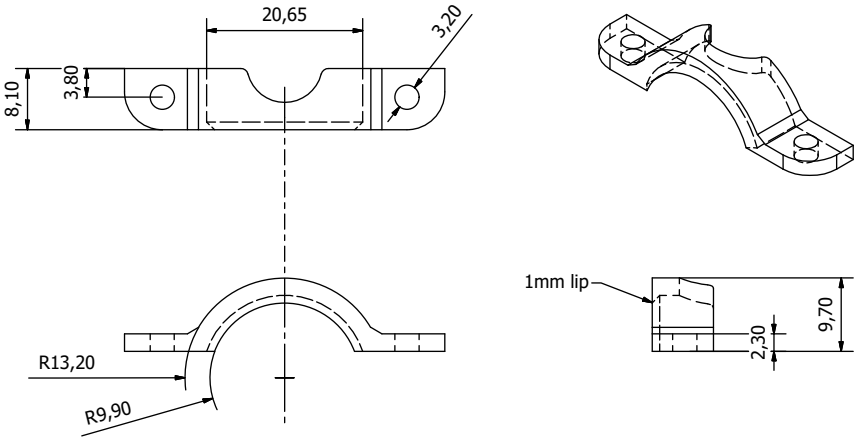


Figure B.7: Model of the cell clamp to be used with the cell mount shown in Figure B.6. All the dimensions are given in millimetres. The drawing is in scale 2:1 on A4. Printed at 0.07 mm layer height and 100 % infill in polylactic acid (PLA).

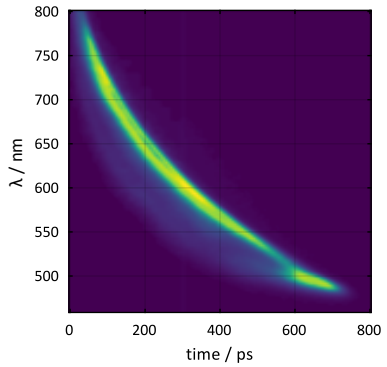


Figure B.8: Streak camera image of a single SC source pulse in the visible region. The specific SC source model is *not* the same as used in this work. Data reproduced from personal communication with NKT photonics technical consultant (Paul Fitzsimons).

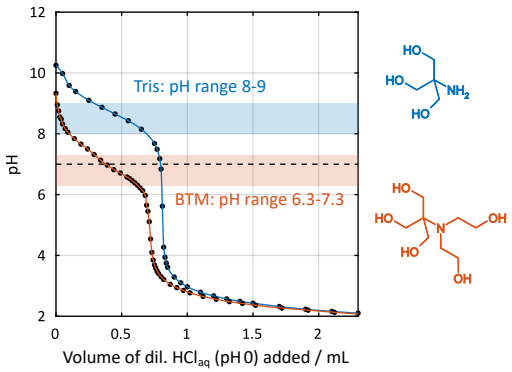


Figure B.9: Acid titration curves of Tris (tris(hydroxymethyl)aminomethane) and BTM (Bis-(2-hydroxy-ethyl)-amino-tris(hydroxymethyl)-methane) buffer solutions. 10 mM buffer solution is made up to a volume of 200 mL in 20 % v/v glycerol. 150 mM NaCl was added to mimic the buffer solution used in Chapter 6. The solution was set on a magnetic stirrer, with a calibrated pH probe inserted into the solution, and basified with conc. NaOH. Dilute HCl$_{aq}$ (pH 0) was added to the solution with a titration column. After each addition the solution was left to equilibrate and the pH was recorded. The structures of Tris and BTM are shown on the side of the figure.

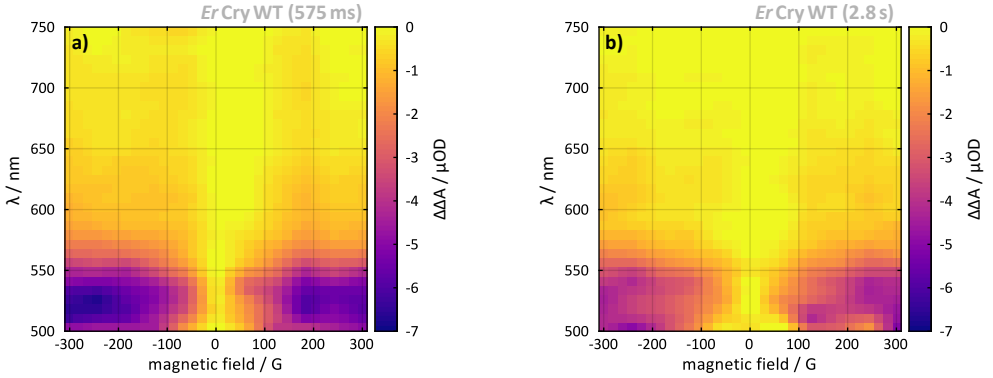


Figure B.10: ΔΔA surfaces of *Er*Cry WT at **a)** early (~575 ms) and **b)** late (~2.8 s) times.

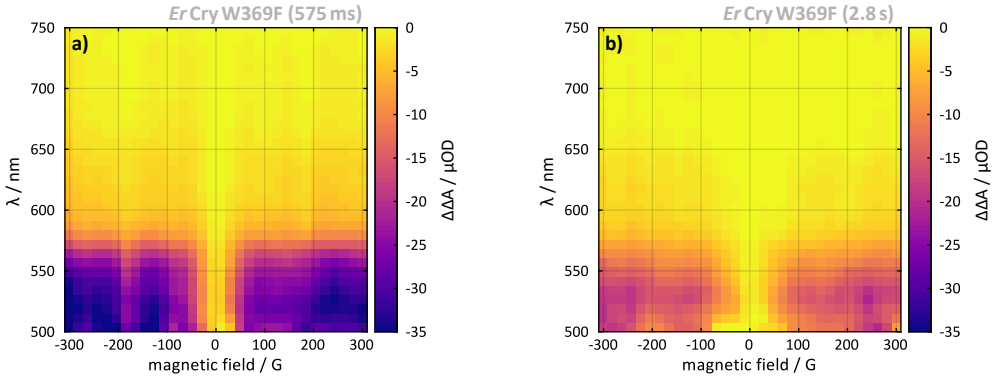


Figure B.11: ΔΔA surfaces of *Er*Cry W369F at **a)** early (~575 ms) and **b)** late (~2.8 s) times.