

UNIVERSITY OF SOUTHAMPTON
FACULTY OF NATURAL AND ENVIRONMENTAL SCIENCES
SCHOOL OF CHEMISTRY

**Advanced Optimal Control Methods
for Spin Systems**

by

David L. Goodwin

*A thesis submitted for the degree of
Doctor of Philosophy*

OCTOBER 2017

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF NATURAL AND ENVIRONMENTAL SCIENCES

School of Chemistry

ADVANCED OPTIMAL CONTROL METHODS FOR SPIN SYSTEMS

by David L. Goodwin

Work within this thesis advances optimal control algorithms for application to magnetic resonance systems. Specifically, presenting a quadratically convergent version of the *gradient ascent pulse engineering* method. The work is formulated in a superoperator representation of the Liouville-von Neumann equation.

A Newton-GRAPE method is developed using efficient calculation of analytical second directional derivatives. The method is developed to scale with the same complexity as methods that use only first directional derivatives. Algorithms to ensure a well-conditioned and positive definite matrix of second directional derivatives are used so the sufficient conditions of gradient-based numerical optimisation are met.

A number of applications of optimal control in magnetic resonance are investigated: solid-state nuclear magnetic resonance, magnetisation-to-singlet pulses, and electron spin resonance experiments.

Declaration of Authorship

I, David L. Goodwin, declare that the thesis entitled *Advanced Optimal Control Methods for Spin Systems* and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- with the oversight of my main supervisor, editorial advice has been sought. No changes of intellectual content were made as a result of this advice.
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- parts of this work have been published as [107–109, 263]

Signed:

Date: *1st March 2018*

Supervisor: *Dr. Ilya Kuprov*

Co-supervisor: *Dr. Marina Carravetta*

External examiner: *Prof. Steffen Glaser*

Internal examiner: *Dr. Tim Freegarde*

Contents

<i>List of Figures</i>	<i>xi</i>
<i>List of Symbols</i>	<i>xiii</i>
<i>Preface</i>	<i>xxi</i>
1 Introduction	1
1.1 Thesis Outline	5
1.2 Early optimal control	7
1.3 Analytical optimal control	8
1.4 Broadband optimal pulses	9
1.5 Gradient based optimal control	9
1.5.1 Solid-state nuclear magnetic resonance	11
1.5.2 Nuclear magnetic resonance imaging	12
1.5.3 Quantum information	13
1.5.4 Electron spin resonance	15
1.6 Modified methods	15
1.6.1 Optimal tracking	15
1.6.2 Cooperative pulses	16
1.7 Other optimal control methods	16
1.8 Numerical computation	17
1.9 Summary	19
2 Magnetic Resonance Theory	21
2.1 Angular momentum	22
2.1.1 Schrödinger's equation	22
2.1.2 Euclidean principle of relativity	22
2.1.3 Orbital angular momentum	23
2.1.4 The uncertainty principle	24
2.1.5 Spherical polar coordinates	25
2.1.6 Intrinsic angular momentum	27
2.2 Irreducible representations of the rotation group	30

2.3	Liouville-von Neumann equation	32
2.3.1	Superoperator algebra	34
2.3.2	Composite systems	36
2.4	The Spin Hamiltonian	38
2.4.1	Zeeman Interactions	39
2.4.2	Inter-Nuclear Interactions	40
2.4.3	Microwave and Radiofrequency terms	42
3	Optimal Control Theory	45
3.1	Numerical optimisation	46
3.1.1	The gradient ascent method	49
3.1.2	Newton's Method	50
3.1.3	Quasi-Newton Methods	52
3.1.4	Line search subproblem	61
3.1.5	Comparison of Newton-type methods	66
3.2	Gradient Ascent Pulse Engineering	67
3.2.1	Bilinear systems	67
3.2.2	Piecewise constant approximation	69
3.2.3	The Pontryagin maximum principle	70
3.2.4	Fidelity measures	74
3.2.5	Fidelity gradient	75
3.2.6	Algorithm	78
4	Hessian Calculations	81
4.1	Fidelity Hessian	82
4.1.1	Hessian structure	83
4.1.2	Hessian elements	83
4.1.3	Control operator commutativity	85
4.2	Hessian Regularisation	86
4.2.1	Cholesky factorisation	87
4.2.2	Trust Region Method	89
4.2.3	Rational Function Optimisation	89
4.2.4	Hessian Conditioning	90
4.3	Convergence analysis	93
4.3.1	Scalar coupled three-spin state-to-state population transfer .	93
4.3.2	Quadrupolar spin state-to-state population transfer	104
4.3.3	Universal rotation	108
5	Propagator Directional Derivatives	111

5.1	The Matrix Exponential	112
5.1.1	Taylor series approximation	112
5.1.2	Padé approximation	114
5.1.3	Chebyshev rational approximation	115
5.1.4	Scaling and squaring	115
5.2	Directional derivatives	116
5.2.1	Directional derivatives of the matrix exponential	117
5.2.2	Block triangular auxiliary matrix exponential	118
5.3	Directional derivatives of the Propagator	120
5.3.1	First propagator directional derivatives	121
5.3.2	Second directional derivatives	122
5.4	Krylov subspace techniques	122
5.5	Propagator recycling	123
5.5.1	Fidelity and gradient calculation	124
5.5.2	Fidelity, gradient and Hessian calculation	125
5.5.3	Parallelisation	126
6	Constrained Optimal Control	129
6.1	Change of variables	130
6.1.1	Transforms between Cartesian & Polar Coordinates	131
6.2	Power penalties	133
6.2.1	Norm-square penalty	133
6.2.2	Spillout-norm-square penalty	135
6.3	Smoothed Controls	136
6.3.1	Magnetisation-to-singlet transfer	137
6.3.2	Robust magnetisation-to-singlet transfer	138
6.3.3	Smooth, robust magnetisation-to-singlet transfer	140
7	Solid State Nuclear Magnetic Resonance	143
7.1	Floquet Theory	144
7.1.1	Powder average	145
7.1.2	Optimal control of a static powder average	146
7.1.3	Optimal control under magic angle spinning	148
7.2	The Fokker-Planck formalism	150
7.2.1	Optimal Control in HMQC-type experiment	151
8	Feedback Control	157
8.1	Feedback Control Optimisation	159
8.1.1	Hardware and Software	160

8.1.2	Optimisation Method	162
8.2	Experimental Application	166
8.2.1	Feedback Optimised Hahn Echo	166
8.2.2	Feedback Optimised OOP-ESEEM	170
9	Summary	177
A	Irreducible Spherical Tensor Operators	179
B	Arbitrary Waveform Generator	183
B.1	AWG Schematics	183
B.2	Xexpr Python code	184
B.3	Feedback Control Results Tables	188
	Bibliography	193

List of Figures

1.1	Finding a minimum of a function using tangents of its gradient plot	3
1.2	Robot manipulator	4
2.1	Spherical coordinates	26
2.2	Stern-Gerlach experiment	28
2.3	Euler angles	32
3.1	Types of stationary points	48
3.2	Comparison of optimisation methods on the Rosenbrock function	66
3.3	Piecewise-constant approximation	68
3.4	A representation of the adjoint state	72
4.1	Block Hessian diagram	82
4.2	Regularised and conditioned Hessian matrices	94
4.3	Control pulses for HCF state transfer	95
4.4	Trajectory analysis of HCF state transfer	96
4.5	BFGS method updating the Hessian and the inverse Hessian	97
4.6	Newton and BFGS methods with Cholesky factorisation	99
4.7	Newton and BFGS methods with regularisation and conditioning	101
4.8	Hessian conditioning iterations	102
4.9	HCF optimisation step lengths	103
4.10	Trajectory analysis of 14N with Newton-step	105
4.11	Convergence of Newton-GRAPE, without a line search.	106
4.12	Convergence of Newton-GRAPE method for a universal rotation.	108
5.1	Parallelism of gradient and Hessian calculations	127
6.1	Norm square penalty and its derivatives	133
6.2	Spillout norm square penalty and its derivatives	135
6.3	Spillout norm cube penalty and its derivatives	136
6.4	The robustness of the SLIC pulse	138
6.5	Pulse shape from an optimised M2S transfer	139
6.6	Smoothed pulse shape from an optimised M2S transfer	139

6.7	The robustness of the optimised M2S transfer	140
6.8	The robustness of the smoothed, optimised M2S transfer	141
7.1	Orientations and weights of a Lebedev grid	146
7.2	Convergence of Newton-GRAPE method for glycine state transfer .	147
7.3	Control pulses for glycine state transfer	149
7.4	Trajectory analysis of glycine state transfer	150
7.5	Trajectory analysis of HMQC-type experiment with hard pulse . .	151
7.6	Optimal control in an HMQC-type experiment	152
7.7	Performance of four test optimisations: 2, 4, 6, and 8 rotor periods	153
7.8	Phase profile of GRAPE pulses for 2, 4, 6, and 8 rotor periods . .	153
7.9	Trajectory analysis of GRAPE pulses for 2, 4, 6, and 8 rotor periods	154
7.10	Convergence analysis with 2, 4, 6, and 8 rotor periods	155
8.1	Closed-loop feedback control	160
8.2	Software flow	161
8.3	Simplex algorithms	163
8.4	Echo results with 33 points	165
8.5	Echo results with 21 points	168
8.6	Echo results with 11 points	169
8.7	Transformed 11 point shapes	171
8.8	OOP-ESEEM pulse sequence diagram	172
8.9	OOP-ESEEM from feedback-control	173
A.1	Second-rank Wigner functions defined in terms of reduced functions	179
A.2	Rank-1 irreducible spherical tensor operators	180
A.3	Rank-2 irreducible spherical tensor operators	181
B.1	The Bruker SpinJet Schematics.	183
B.2	Feedback control results for OOP-ESEEM experiments	188
B.3	Feedback control results for 2-pulse echo experiments	189
B.4	Convergence of Nelder-Mead and multidirectional search algorithms	190

List of Symbols

Symbol	Description	Notes	Chapter
MAGNETIC RESONANCE:			
$\langle \cdot \cdot \rangle$	inner product		§§1–4, 6, 7
t	time interval		§§2, 3, 5, 7
i	imaginary number	$= \sqrt{-1} = \frac{-1}{\sqrt{-1}}$	§§2–5, 7
$ \Psi(t)\rangle$	state of a system at a time t		§2
$ \Psi_i\rangle$	complete orthonormal basis states	(Hilbert space)	§2
$c_i(t)$	time dependent coefficient of state		§2
$\hat{\mathcal{H}}$	Hamiltonian operator		§2
\hat{R}	rotation operation		§2
\mathbf{A}	(bold typeface) a matrix		§§2–6
\mathbf{A}^T	transpose of a matrix \mathbf{A}	$A_{ij}^T = A_{ji}$	§§2, 4
\mathbf{A}^*	complex conjugate of \mathbf{A}	$= \text{Re}(\mathbf{A}) - \text{Im}(\mathbf{A})$	§§2, 3, 6
\mathbf{A}^\dagger	Hermitian conjugate of \mathbf{A}	$= (\mathbf{A}^*)^T$	§§2, 3
\mathbf{A}^{-1}	matrix inverse of \mathbf{A}	$\mathbf{A} = (\mathbf{A}^{-1})^{-1}$	§§2–5, 7
$[\mathbf{A}, \mathbf{B}]$	commutator of matrices \mathbf{A} and \mathbf{B}	$= \mathbf{AB} - \mathbf{BA}$	§§2, 4, 5
$ \Psi\rangle$	arbitrary state vector		§§2–4
$ \Psi'\rangle$	transformed state vector $ \Psi\rangle$		§2
$\langle \Psi $	Hermitian conjugate of $ \Psi\rangle$	$\langle \Psi = \Psi\rangle^\dagger$	§§2–4
\hat{r}	displacement operator		§2

Symbol	Description	Notes	Chapter
\hat{p}	momentum operator	$= -i\hbar\nabla$	§2
∇	gradient operator	$= \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right)^T$	§§2–7
\hbar	reduced Planck constant ($\text{Js}(\text{rad})^{-1}$)	$\approx 1.05 \times 10^{-34}$	§2
(x, y, z)	Cartesian coordinates		§§2, 6
(r, θ, φ)	spherical coordinates		§§2, 6
r	radial distance	$= \sqrt{x^2 + y^2 + z^2}$	§§2, 6
θ	inclination/polar angle	$= \arccos \frac{z}{r}$	§2
φ	azimuthal angle	$= \text{atan2} \frac{y}{x}$	§§2, 4, 6
\hat{L}	orbital angular momentum operator	$= \hat{r} \times \hat{p}$	§2
\hat{L}_x	x -component of angular momentum operator	$= -i[\hat{L}_y, \hat{L}_z]$	§2
\hat{L}_y	y -component of angular momentum operator	$= -i[\hat{L}_z, \hat{L}_x]$	§2
\hat{L}_z	z -component of angular momentum operator	$= -i[\hat{L}_x, \hat{L}_y]$	§2
\hat{L}^2	Casimir operator	$= \hat{L}_x^2 + \hat{L}_y^2 + \hat{L}_z^2$	§2
\hat{L}_+	raising operator	$= \hat{L}_x + i\hat{L}_y$	§2
\hat{L}_-	lowering operator	$= \hat{L}_x - i\hat{L}_y$	§2
Δx	change in the variable x		§2
$\langle A \rangle$	expected value of A		§2
$Y_\ell^m(\theta, \varphi)$	spherical harmonics		§2
\mathbb{N}	natural numbers	$\dots, -2, -1, 0, 1, 2, \dots$	§2
ℓ	angular momentum quantum number	$\ell \in \mathbb{N}$	§2
m	magnetic quantum number	$m = -\ell, \dots, \ell$	§2
$P_\ell(\xi)$	Legendre polynomials	$= \frac{1}{2^\ell \ell!} \frac{d^\ell}{d\xi^\ell} [(\xi^2 - 1)^\ell]$	§2
$ \Psi(\vec{r}, s)\rangle$	spinor	$= \sum_{n,k} \Psi_n(\vec{r})\rangle \otimes \Psi_k(s)\rangle$	§2
s	spin quantum number	$s = 0, \frac{1}{2}, 1, \frac{3}{2}, 2, \dots$	§2
$\mathbb{1}$	identity matrix		§§2, 3, 5, 6
$\mathbf{0}$	vector or matrix of zeros		§§2, 3

Symbol	Description	Notes	Chapter
\hat{S}	intrinsic angular momentum (spin)		§2
\hat{S}_x	x -component of spin operator	$= -i[\hat{S}_y, \hat{S}_z]$	§2
\hat{S}_y	y -component of spin operator	$= -i[\hat{S}_z, \hat{S}_x]$	§2
\hat{S}_z	z -component of spin operator	$= -i[\hat{S}_x, \hat{S}_y]$	§2
\hat{S}_+	spin raising operator	$= \hat{S}_x + i\hat{S}_y$	§2
\hat{S}_-	spin lowering operator	$= \hat{S}_x - i\hat{S}_y$	§2
$ \alpha\rangle$	α spin state	$= (1, 0)^T$	§2
$ \beta\rangle$	β spin state	$= (0, 1)^T$	§2
T	total time		§§2, 3
$\hat{\mathcal{H}}(t)$	time-dependent Hamiltonian		§§2, 3
$ \psi(t)\rangle$	time dependent state		§2
$\exp_{(o)}$	time-ordered exponential		§§2, 3
$\hat{\rho}(t)$	time-dependent density operator	$= \psi(t)\rangle\langle\psi(t) $	§§2, 3, 6, 6
p_{ij}	probability associated with states ψ_i and ψ_j		§2
$\hat{\mathcal{U}}(T)$	time propagator over the interval $[0, T]$		§§2, 7
$\hat{\rho}_{1,2}$	2 non-interacting, uncorrelated subsystems	$= \hat{\rho}_2 \otimes \hat{\rho}_2$	§2
$\hat{\rho}_{1,2,\dots,k}$	k non-interacting, uncorrelated subsystems	$= \hat{\rho}_2 \otimes \hat{\rho}_2 \otimes \dots \otimes \hat{\rho}_k$	§2
$\hat{S}^{(k)}$	spin operator for the k^{th} spin		§2
$\hat{\mathcal{H}}_{1,2}$	Hamiltonian of 2 non-interacting subsystems	$= \hat{\mathcal{H}}_1 \otimes \mathbb{1}_2 + \mathbb{1}_1 \otimes \hat{\mathcal{H}}_2$	§2
$\hat{T}_{\ell m}^{(12)}$	2-particle tensor operators		§§2, 4
$\mathcal{C}_{m_1 m_2 m}^{\ell_1 \ell_2 \ell}$	Clebsch-Gordan coefficients		§2
$\hat{\mathcal{H}}$	Hamiltonian superoperator	$= \mathbb{1} \otimes \hat{\mathcal{H}} - \hat{\mathcal{H}}^\dagger \otimes \mathbb{1}$	§§2, 3, 7
$\hat{\mathcal{R}}$	relaxation superoperator		§§2, 3
$ \rho(t)\rangle$	time-dependent, vectorised density matrix		§§2, 3, 7

Symbol	Description	Notes	Chapter
B_0	static magnetic field in the z -direction		§§2, 6
B_1	pulsed magnetic field		§§2, 6
\vec{B}	external magnetic field vector		§2
$\hat{\vec{S}}$	vector of Cartesian spin operators	$= (\hat{S}_x, \hat{S}_y, \hat{S}_z)$	§§2, 4
$\hat{\mathcal{H}}_z$	Zeeman interaction Hamiltonian	$= \hat{\vec{S}} \cdot \mathbf{A} \cdot \vec{B}$	§2
\mathbf{A}	Zeeman interaction tensor		§2
$\hat{\mathcal{H}}_{cs}$	chemical shielding Hamiltonian	$= \hat{\vec{S}} \cdot \boldsymbol{\delta} \cdot \vec{B}$	§2
$\boldsymbol{\delta}$	chemical shielding tensor		§2
$\hat{\mathcal{H}}_g$	electron Zeeman Hamiltonian	$= \mu_B \hat{\vec{L}} \cdot \mathbf{g} \cdot \vec{B}$	§2
μ_B	Bohr magneton	$= \frac{e\hbar}{2m_e}$	§2
m_e	electron mass (kg)	$\approx 9.11 \times 10^{-31}$	§2
\mathbf{g}	electron g-tensor		§2
$\hat{\mathcal{H}}_{NN}$	inter-nuclear interaction Hamiltonian		§2
$\hat{\mathcal{H}}_{DD}$	dipole-dipole interaction Hamiltonian		§2
γ	magnetogyric ratio		§2
$\hat{\mathcal{H}}_{SR}$	spin-rotation coupling Hamiltonian	$= - \sum_k \hat{\vec{L}} \cdot \mathbf{A}_k \cdot \hat{\vec{S}}_k$	§2
\mathbf{A}_k	spin-rotation coupling tensors		§2
$\hat{\mathcal{H}}_Q$	quadrupolar coupling Hamiltonian		§§2, 4
$\hat{\mathcal{H}}_J$	J-coupling Hamiltonian	$a(\hat{\vec{L}} \cdot \hat{\vec{S}})$	§§2, 4
$\hat{\mathcal{H}}_{NE}$	electron-nuclear interaction Hamiltonian		§2
$\hat{\mathcal{H}}_{EE}$	inter-electron interaction Hamiltonian		§2
H_n	Hermitian matrices of periodic time-dependent functions		§7
$s(t)$	measured signal as a function of time		§6
\vec{s}	measured signal over time		§6
\vec{r}	reference signal measured from an experiment with hard pulses		§6
Q	characteristic exponents		§7

Symbol	Description	Notes	Chapter
\hat{F}_n	ladder operators in Fourier space		§7
\hat{N}	number operator		§7
NUMERICAL OPTIMISATION:			
$J(\mathbf{c})$	scalar objective function		§§3, 8
\mathbf{c}	objective variable	$= (c_1, c_2, \dots, c_n)^T$	§§3, 8
\mathbb{R}^n	n D space of real numbers		§§3–5
\mathbb{C}^n	n D space of complex numbers		§5
$\min_{\mathbf{c}} \{J(\mathbf{c})\}$	minimum of a function $J(\mathbf{c})$		§§3–5
$\max_{\mathbf{c}} \{J(\mathbf{c})\}$	maximum of a function $J(\mathbf{c})$		§§3, 5, 6
\mathbf{h}	small displacement vector		§3
$\nabla J(\mathbf{c})$	gradient vector of objective		§§3, 4
$\nabla^2 J(\mathbf{c})$	Hessian matrix of objective		§§3, 4
∇	del operator	$= \left(\frac{\partial}{\partial c_1}, \frac{\partial}{\partial c_2}, \dots, \frac{\partial}{\partial c_n} \right)^T$	§§3–6
∇^2	Hessian operator	$= \nabla \rangle \langle \nabla $	§§3–6
∇^r	r^{th} order derivative		§§3, 5
\jmath	optimisation iteration		§§3, 4
\mathbf{c}_\jmath	objective function at an iteration \jmath		§3
\mathbf{c}_*	minimiser/maximiser of objective		§3
	function		
$ \mathbf{c}_\jmath\rangle$	direction at a point of surface, \mathbf{c}_\jmath		§3
$\ \cdot\ _n$	n -norm		§§4–6
$\ \cdot\ _\infty$	infinity norm		§§3, 4
$\ \cdot\ _{\text{F}}$	Frobenius norm		§4
λ	eigenvalues		§§3, 4
\mathbf{d}	line search direction		§3
\mathbf{d}_\jmath	line search direction at iteration \jmath		§3
ε	line search step length		§§3, 4
$\boldsymbol{\delta}$	Newton step		§3

Symbol	Description	Notes	Chapter
$\mathcal{O}(\cdot)$	error term		§3
EPS	double-precision floating point	$\approx 2.22 \times 10^{-16}$	§§3, 4
	accuracy		
Δc	change in objective variable	$= \mathbf{c}_{j+1} - \mathbf{c}_j$	§3
$\Delta \mathbf{g}$	change in gradient	$= \nabla J(\mathbf{c}_{j+1}) - \nabla J(\mathbf{c}_j)$	§3
\mathbf{H}	approximation to the Hessian matrix		§3
\mathbf{I}	approximation to the inverse of the Hessian matrix		§3
$\Delta \mathbf{C}_j$	store of most recent Δc		§3
$\Delta \mathbf{G}_j$	store of most recent $\Delta \mathbf{g}$		§3
\mathbf{L}	lower triangular matrix		§4
$\mathbf{\Lambda}$	diagonal matrix of eigenvalues		§4
\mathbf{Q}	matrix with columns of eigenvectors		§4
\mathbf{S}	scaling matrix		§4
$D_{\mathbf{B}}(f(\mathbf{A}))$	derivative of the function f at \mathbf{A} in the direction \mathbf{B}		§5
$D_{\mathbf{B}}^r(f(\mathbf{A}))$	r^{th} -derivative of the function f at \mathbf{A} in the direction \mathbf{B}		§5
β	simplex expansion factor		§8
γ	simplex contraction factor		§8
\mathbf{c}_j	simplex vertex		§8
\mathbf{c}_0	initial simplex vertex		§8
\mathbf{c}_c	simplex centroid		§8
\mathbf{X}	simplex		§8
\mathbf{R}	reflected simplex		§8
\mathbf{E}	expanded simplex		§8
\mathbf{C}	contracted simplex		§8
$\hat{\Delta}$	Finite difference matrix/differentiation approximation		§6

Symbol	Description	Notes	Chapter
OPTIMAL CONTROL:			
$\hat{\mathcal{H}}_k$	Control operator		§§2, 3
$\hat{\mathcal{H}}_x^{(k)}, \hat{\mathcal{H}}_x^{(k)}$	k^{th} control operator in the x -direction		§§2–4, 6
$\hat{\mathcal{H}}_y^{(k)}, \hat{\mathcal{H}}_y^{(k)}$	k^{th} control operator in the y -direction		§§2–4, 6
$c_k(t)$	time-dependent control amplitudes		§§3, 6
K	number of control operators		§§3–5
$ \rho_0\rangle$	initial state of system		§3
$ \sigma\rangle$	desired state of system		§§3, 6
Δt	small time slice		§§3, 5
N	number of time slices		§§3–6
t_n	n^{th} time interval		§3
\mathbf{c}	2D vector of $K \times N$ control amplitudes		§§3, 4, 6
\mathbf{c}_k	row vector of N control amplitudes		§§3, 6
\mathbf{c}_n	column vector of K control amplitudes		§3
$c_{k,n}$	individual element of \mathbf{c}		§§3–5
$\dot{\rho}(t)$	time derivative of ρ		§3
$\mathcal{L}(c(t))$	Lagrangian as a function of control $c(t)$		§3
$\mathcal{J}(\rho(T))$	fidelity: terminal cost of Lagrangian		§§3–6
\mathcal{K}	penalty: running cost of Lagrangian		§§3, 6
$ \chi(t)\rangle$	costate/adjoint state of system	$ \chi(T)\rangle = \sigma\rangle$	§3
$ \rho_n\rangle$	state at a time interval n		§§3–5
$ \chi_n\rangle$	adjoint state at a time interval n		§§3–5
\mathcal{H}_{P}	pseudo-Hamiltonian		§3
$\hat{\mathcal{P}}_n(\mathbf{c}_n)$	time propagator of controls		§§3, 4
$\tilde{\mathbf{c}}$	normalised control amplitude vector		§§3, 6
p_c	nominal power level of control pulses		§3
$\mathbf{c}^{(j)}$	control vector at an optimisation		§4
	iteration j		

Symbol	Description	Notes	Chapter
$\mathbf{D}_{m,n}$	Hessian matrix of controls at a single time point, $K \times K$		§5
\mathbf{P}_n	time propagator matrix		§5
\mathbf{H}	Hamiltonian matrix		§5

Preface

*Do not think that what is hard for you to master is humanly impossible;
but if a thing is humanly possible, consider it to be within your reach.*

– Marcus Aurelius, MEDITATIONS IV

Resources

The main resource used within this thesis is the *Spinach* software library^{1} [130], developed by Ilya Kuprov, Hannah Hogben, Luke Edwards, Matthew Krzystyniak, Peter J. Hore, Gareth T.P. Charnock, Dmitry Savostyanov, Sergey Dolgov, Liza Sutarina, and Zenawi Welderufael.

The majority of numerical simulation in this work was completed with Matlab 2016-2017a/b. All random numbers were seeded with `rng(1)` from within a Matlab session, unless otherwise stated.

ESR spectroscopy measurements were performed in the Centre for Advanced ESR (CAESR) in the Department of Chemistry, University of Oxford, using a Bruker BioSpin EleXSys II E580 spectrometer operating with a SpinJet arbitrary waveform generator (AWG) that is based on a SPDevices SDR14 PCI board, with a 0.625 ns time base. Samples were held at 85 K in a Oxford Instruments CF935O cryostat under a flow of cold N2 gas, controlled by an Oxford Instruments Mercury instrument temperature controller. At X-band, the Bruker Biospin ER4118-MD5-W1 resonator was used, which is a sapphire dielectric resonator of dimensions

^{1} Available at www.spindynamics.org

5 mm ID, 10 mm OD and 13 mm height, that was overcoupled for pulsed measurements to a Q-value of about 200. At Q-band the resonator was a ER5107D2 of typically 80-100 MHz bandwidths.

Photoexcitation of the OOP-ESEEM sample was accomplished with a Continuum Shurlite Nd:YAG emitting a 7 ns LASER pulse at the 3rd harmonic of 1064 nm, 355 nm, at a rate of 10 Hz, attenuated with a $\lambda/2$ plate and finally depolarized to a 1 mJ pulse energy. The beam was found to match the 5 mm cryostat window and was not further focused on the sample. Synchronisation of the LASER and OOP-ESEEM measurement involved using the E580 console PatternJet board user-defined channel as an external trigger to the Stanford Research DG645 delay generator.

The author acknowledges the use of the *iridis* High Performance Computing Facility, and associated support services at the University of Southampton, in the completion of some parts of this work.

This report was typeset by the author using L^AT_EX and the T_EX mathematical typesetting system of Donald Knuth^{1}, and vector graphics were generated with

Acknowledgements

A number of people have influenced this work, particularly my supervisor Prof. Ilya Kuprov; without his advice, I would not have had confidence to undertake this project.

Academics working in the area of optimal control that have had a direct involvement in this work through collaboration and conversation are Mads Sloth Vinding, Thomas Schulte-Herbrüggen, Tony Reinsperger, Robert Zeier, David Tannor, Reuven Eitan, Sean Wright, Jack Saywell. Less direct involvement through discussion, although sometimes more substantial; Thomas Heydenreich, Wolfgang Kallies, Sophie Schirmer, Frank Langbein, Martin Koos, Burkhard Luy, Daniel Wisniewski, Shai Machines, and Tim Freegarde.

Magnetisation-to-singlet state work was an outcome of interesting conversation with Stuart Elliott, Chiristian Bengs, Gabriele Stevanato, Giuseppe Pileio, and Malcolm Levitt.

^{1}D.E.Knuth, T_EX and METAFONT, New Directions in Typesetting, American Mathematical Society and Digital Press, Bedford, Massachusetts (1979)

Work on solid state NMR was a collaboration with James Jarvis, Phil Williamson, Marina Carravetta, and Maria Concistre.

In developing the work using the ESR AWG, I am most grateful to Will Myers for his extensive knowledge of all hardware under his command, but mostly for his patience – I’m sure he saw emptiness behind my eyes while he tried to explain how to configure the spectrometer to perform basic Hahn echo to me. I don’t know if he knows that this was the only physical experiment I have done since my undergraduate days with physics experiments in 2001–2004. In enabling me to do this work I am also grateful to Christiane Timmel, Arzhang Ardavan, Alice Bowen, Thomas Prisner, Anton Tcholakov, Gavin Morley, and Gunnar Jeschke for insightful discussions, and technical support of Bruker through Peter Höfer and Patrick Carl.

Academics to influence me and allow me to believe in myself are Phil Dawson, Tobias Brandes, Mike Godfrey, Clive Saunders, Klaus Gernoth (whose undergraduate course in advanced quantum mechanics and path integrals seduced me into enjoying difficult, and possibly insoluble, problems). Also, Jason Ralph, Neil Oxtoby, and Charles Hill for allowing me to work in self imposed solitude; a mistake I needed to make at that time.

Finally, personal acknowledgement is given to my wife Katarzyna – who helped me in an uncountable number of ways – my father Martin, and my mother Jane. These people have become a bedrock for supporting me and accepting my eccentricities without question. Further acknowledgement is also given to Martin Goodwin and Emmaleen Goodwin, who helped in proofreading this document.

This thesis is dedicated to my son Ruadhán, born three months before submission of this Ph.D. thesis, and who I hope will surpass all of my achievements.

Southampton
October 2017

David L. Goodwin

§ 1

Introduction

In fact, the mere act of opening the box will determine the state of the cat, although in this case there were three determinate states the cat could be in: these being Alive, Dead, and Bloody Furious.

– Terry Pratchett, LORDS AND LADIES

The subject of this thesis is *optimal control* and its aim is advancing existing methods of optimal control for spin systems of magnetic resonance. A novel method with improved convergence properties will be presented with a focus on computational efficiency and avoiding numerical problems. The computationally expensive task of calculating the exponential of a matrix will be reviewed and concluded with particularly useful numerical efficiencies which decrease simulation time. The thesis will be completed by investigating realistic applications of optimal control solutions, including methods to overcome experimental limitations.

Many descriptions of optimisation use the analogy of hiking in a mountainous terrain to highlight the character of numerical optimisation. The goal of the hiking experience could be to get to the highest peak within a range of mountains, or descend to the lowest altitude. The first of these goals is finding the maximum altitude within the landscape, and the second is finding the minimum; here we optimise the altitude as a function of the coordinates. Although this analogy is easy to relate to, it is not a good example of optimisation: In navigating the mountain we have a map and possibly a compass, more importantly we have a clear view of the mountain and can pick our path to a visible peak; we know the easiest path to the maximum on the outset. Furthermore, the mountain range has many peaks – an initial path may only find a low-lying peak which becomes apparent from the summit.

Example of numerical optimisation

A better example is to follow that of Isaac Newton, finding the roots to a simple polynomial function of a single variable, $f(x)$. Assuming the function is well behaved, smooth with no discontinuities, the maximum and minimum should reside in places where the gradient of the function is equal to zero. As a first step in finding these solutions is to use an algorithm to find iteratively better estimates to the roots of a function, the places where $f(x) = 0$: This algorithm is known as the Newton-Raphson method (also known as the Newton method) and is attributed to Isaac Newton [222] and Joseph Raphson [248]:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

where $f(x_n)$ is the function of a real variable x_n at the iteration n . $f'(x_n) = \frac{df}{dx} \Big|_{x_n}$ is the tangent to the function $f(x)$ at x_n . It is instructive to show the algorithm with an example^{I}. For the function $f(x) = \frac{1}{3}x^3 - x + 1$, the tangent to the function can be found analytically by differentiation.

However, the aim is not to find the roots of the function, the aim is to find the roots of the derivative of the function: the place where the gradient is = 0^{II}. Substituting this into the previous equation we find an iterative algorithm to find the extrema of a function $f(x)$:

$$x_{n+1} = x_n - \frac{f'(x_n)}{f''(x_n)}$$

where the second derivative, the derivative of the gradient is $f''(x_n) = \frac{d}{dx} \frac{df}{dx} \Big|_{x_n} = \frac{d^2 f}{dx^2} \Big|_{x_n}$ evaluated at x_n . The Newton-Raphson method can be used to find these roots to the gradient of the function, and the method is shown for the function $f(x) = \frac{1}{3}x^3 - x + 1$ in FIG. 1.1.

The algorithm converges to an extremum at $x = +1$; within an acceptable level of accuracy at 4 iterations and further than the machine precision^{III} after 5 iterations. Notice that starting from a guess $x_0 < 0$ converges to a minimum at -1 , a starting guess of $x_0 = 0$ will be undefined at the first iteration, and a starting

^{I}The function used by Newton in his original example [222] was $f(x) = x^3 - 2x + 5$.

^{II}Pre-dating Isaac Newton's and Gottfried Leibniz' work on calculus, in the middle of the seventeenth century, de Fermat's work in analytical geometry used tangents to show that at the extreme points of various curves, the gradient becomes zero.

^{III}this is floating-point arithmetic, also referred to as *eps*, is about 16 decimal places on a 64-bit computer

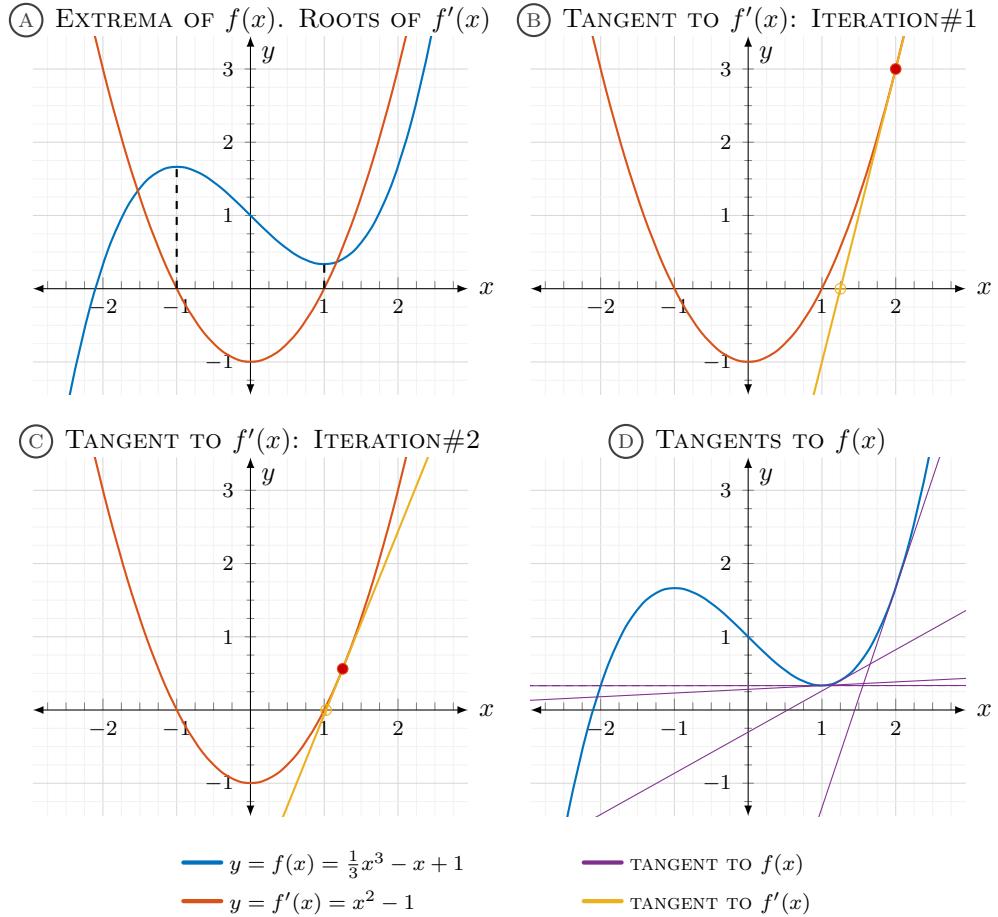


FIGURE 1.1: Finding a minimum of a function using tangents of the gradient from an initial trial point. (A) the function $f(x) = \frac{1}{3}x^3 - x + 1$ and its gradient $x^2 - 1$ with two extremal solutions at $x = \pm 1$. (B) First iteration: An initial guess to the extrema at $x = 2$. The tangent to the gradient shows where to look for a better point, at $x = 1.25$. (C) Second iteration: tangent of the gradient at $x = 1.25$, finding a better point at $x = 1.025$. (D) Iterations shown as the tangents on the functional, converging closer to a tangential solution with zero gradient.

guess $x_0 > 0$ converges to a maximum at $+1$. The reason is clearly seen with a graphical representation of the algorithm in FIG. 1.1.

Example of control theory

Optimal control theory is more subtle. Although optimisation is a part of optimal control theory, the functional of the optimisation is more than navigating an n -dimensional space. Control theory separates the controllable and uncontrollable physics of the system, allowing the controllable physics to become part of

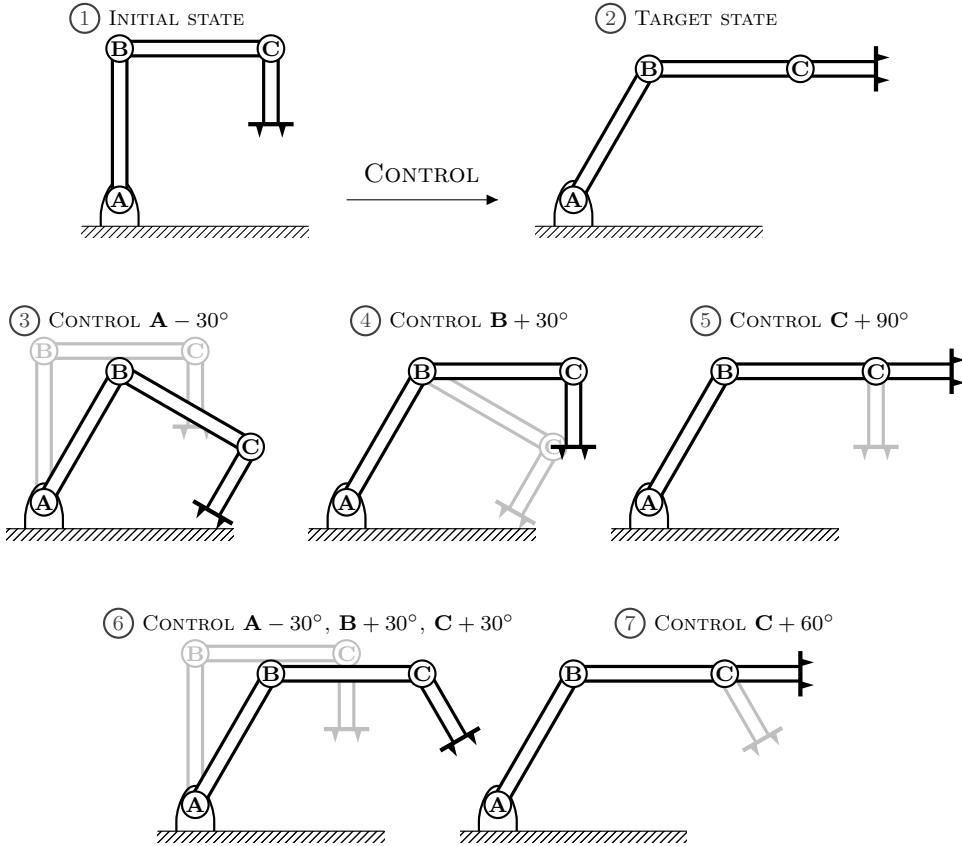


FIGURE 1.2: Diagram of a robot manipulator control task. Upper row is the control task of taking the manipulator from ① an initial state to ② a defined target state. Central row shows a control strategy, moving each control motor in turn, ③, ④, then ⑤. Lower row shows an efficient control strategy, allowing all control motors to be moved simultaneously where possible, ⑥, then a final single control ⑦. Assuming all motors move at the same speed, the control strategy of the lower row takes 60% of the time it takes the central row control task.

the optimisation functional, and the uncontrollable physics of the system become inherent “drift”.

A classical example of a control theory is shown in FIG. 1.2, where the task of taking a robot manipulator from an initial position to a target position can be more efficient if control theory is used. The robot manipulator in FIG. 1.2 has three degrees of freedom, three controllable motors to position the grippers. The easiest way to control the robot manipulator is to move each pivot in turn, controlling each motor to the desired final position in turn. However, if motors are allowed to be controlled simultaneously, the task is performed quicker. This can be viewed as an application of control theory, although simplistic, reducing the amount of time the control task takes.

A proper use of control theory would take into account the weight of each section of the manipulator, the weight of any potential load at the end of the manipulator, the surrounding environment, and ensuring movement of the manipulator is smooth with motors started slowly and finishing slowly. This information, the physics, is easily programmed into a robot manipulator which performs repetitive tasks.

The task of finding the most efficient, the most optimal, controls to move from the initial state to the target state is the task of optimal control.

§ 1.1 Thesis Outline

Chapter 1

An introduction to optimal control, broadly reviewing the literature of the last fifteen years, where optimal control methods have been applied to the area of magnetic resonance. The review focuses on the optimal control method known as GRAPE, but includes a number of other methods also used for optimal control, some well established and some newer.

Chapter 2

This chapter will give an overview of magnetic resonance theory. It will form a particularly theoretical chapter that will be referred to in later chapters, and also serve the purpose of setting out mathematical formalisms that are used in the software toolbox *Spinach* – the software used in all numerical simulation within this thesis.

Chapter 3

A presentation of the GRAPE method in the context of magnetic resonance. The chapter starts from the mathematical area of numerical optimisation, progressing through control theory to optimal control theory. The GRAPE method in this thesis uses a superoperator algebra representation of the Liouville-von Neumann equation and the mathematical derivations of the chapter are all in the context of this formalism.

Chapter 4

Work developed by the author for this thesis and published in [108]: advancing the GRAPE method from the previous best superlinearly convergent method [86] to quadratic convergence. The method presented is shown to scale linearly with a similar computational complexity to previous methods.

The Newton-GRAPE method, unique work for this thesis, will be presented along with well established numerical procedures to avoid predicted singularities in the second-order optimal control method. Convergence analysis will be shown for simulations of magnetic resonance pulse experiments using this optimal control method.

Chapter 5

The matrix exponential is far from a trivial mathematical operator. This chapter reviews a number series methods from the “*nineteen dubious ways to compute the exponential of a matrix*” [211, 212], giving recommendation on which method to use for simulation in magnetic resonance systems – the method used in the software toolbox *Spinach*. The chapter then goes on to present work unique to this thesis, published in [107], of analytical second directional derivative for the GRAPE method. §§4, 5 set out the focus of this thesis – the Newton-GRAPE method of optimal control.

Chapter 6

An overview of popular quadratic penalty methods is presented, optimal control of phase in an amplitude-phase representation of control pulses, and smoothed and shaped pulse solutions with application to finding smooth and robust pulses to generate the singlet state.

Chapter 7

This chapter sets out the application of optimal control to solid-state NMR, simulating the optimisation algorithm over the crystalline orientations in a powder sample, and also for magic angle spinning experiments. The chapter builds to finding optimal control pulses to excite the overtone transition in an HMQC-type experiment.

Chapter 8

Work on optimal control in the area of electron spin resonance is far from trivial – the pulses sent through the spectrometer are not the same shape as those seen by the sample. This is not a problem for shapes that do not vary rapidly – but optimal control pulses tend to be very specific, with high frequency components that are very specific to the outcome of the experiment. To investigate this, a real-time feedback control loop is set up and experiments are optimised in a brute-force way. Experimental results are shown for Hahn echo experiments with shaped pulses produced from this type of real-time feedback loop.

Chapter 9

Summary of main findings of the thesis.

The remainder of this introductory chapter will outline a survey of published literature in the area of optimal control. Although not extensive in its applications, the story of the development of optimal control theory in the context of finding optimal pulses for magnetic resonance systems is fully considered. The reader is directed to a more detailed perspective of modern optimal control theory in the report *Training Schrödinger's cat: Quantum optimal control: Strategic report on current status, visions and goals for research in Europe* [98].

§ 1.2 Early optimal control

Optimal control theory in magnetic resonance started at a similar time to hardware implementation of arbitrarily shaped pulses [283, 303]. Although not optimal control theory in itself, numerical optimisation was used to design highly selective excitation pulses [197] and population inversion pulses [326]. A simple approach was to minimise the square of the difference between an ideal signal and a measured signal,

$$\min_{\text{pulse shape}} \left(\text{IDEAL SIGNAL} - \text{MEASURED SIGNAL} \right)^2$$

using a Nelder-Mead simplex method [220] varying parameters describing a shaped pulse [197]. This approach to optimisation will be revisited in §8.

The development of appropriate hardware, and the early successes of experiments using non-rectangular pulses, led researchers to explore the use of optimal control theory to design “better” pulses for magnetic resonance experiments. Selective excitation problems using numerically designed $\pi/2$ and π pulses used optimal control theory to solve Bloch equations^{I} with a piecewise constant approximation of an arbitrary pulse shape [43]. The optimisation metric used was defined as a *minimum-distance* measure between the current magnetisation vector and a desired magnetisation vector, then solving a Lagrange equation to find an optimum. Later, a modification to the *minimum-distance* measure to include a measure of adiabaticity gave adiabatic and band selective pulse shape solutions [253].

In magnetic resonance systems an ensemble of *spins* is manipulated to achieve a desired ensemble state by the application of *unitary operations* [73]. Investigation

^{I}Bloch equations describe magnetisation of isolated spins.

into the *controllability* of quantum systems showed that, given a defined initial state and enough pulsing time, any eigenstate of the system can be populated when $\hat{\mathcal{H}}_x$ and $\hat{\mathcal{H}}_y$ controls are available, and their commutators with the drift span the active space [141]^{11}.

§ 1.3 Analytical optimal control

One of the first successful applications of optimal control theory in NMR used an existing deuterium decoupling pulse sequence [264] as an initial guess to a gradient-based numerical optimisation algorithm [189]. The optimisation metric was defined by the eigenvalues and eigenvectors of the propagator from an average Hamiltonian. The analytical derivatives of the linear operators [2] used for the gradient-based optimisation were derived for a piecewise constant approximation of an arbitrary pulse sequence.

Using a method developed for laser spectroscopy [142, 327], analytical design of optimal unitary operations for use in NMR proved to be a difficult task for complicated, coherence-order selective experiments [313, 314]. However, the maximal unitary bounds on general NMR polarisation transfer experiments were shown to be much larger than the apparent limits of NMR polarisation transfer using state-of-the-art experiments of that time [99, 313]. The search for solutions nearer to these bounds, finding pulsed solutions to give better experimental results, and the search for solutions at these bounds, would become the work of the following decade.

Given that a system is controllable and can be steered from one defined state to another [141], optimal solutions must exist. With the aim of designing optimally short pulses, to minimise the effects of relaxation, unitary transforms are found analytically by separating out a Lie subgroup corresponding to controllable operations, then deducing pulses from the unitary operator [152]. Pulses with the shortest possible time were found for coherence transfer experiments in heteronuclear two-spin systems [152] and linear three-spin systems [153].

^{11}Unitary operations, $\hat{\mathcal{H}}_x$ controls, and $\hat{\mathcal{H}}_y$ controls will be introduced in §2

§ 1.4 Broadband optimal pulses

Optimal control pulses can be designed to be broadband, robust to resonance offsets, by simulating the optimal control algorithm over a range of offsets [287]. These pulses are called Broadband Excitation By Optimised Pulses (BEBOP). Further to this, pulses designed by BEBOP were designed to have a shorted duration and less power [288].

A similar method was used to create Broadband Inversion By Optimised Pulse (BIBOP) [159]. This optimal control problem is simulated over a range of control pulse power levels – creating pulses robust to miscalibration. Allowing only the phase of the pulses to vary, keeping a constant amplitude, allows pulses designed with BEBOP to be calibration-free [286, 289].

§ 1.5 Gradient based optimal control

The original paper setting out the theory for *Gradient Ascent Pulse Engineering*^{I} (GRAPE) followed the work on analytical optimisation [152] - setting out an elegant base for Newton-type optimisation methods applied to quantum systems [154]. The essence of the method involved discretising the system into slices of time, each with a constant Hamiltonian [43], and splitting the system into an uncontrollable drift Hamiltonian, and control operators with associated amplitude constants. The validity of the approximation defines the validity of GRAPE. Along with similar methods, the measure to maximise – to make the highest possible value – is the overlap between two states termed the fidelity of the two states [99]:

$$\max_{\text{pulse shape}} \left\langle \text{TARGET STATE} \mid \text{CURRENT STATE} \right\rangle$$

The angled brackets, $\langle \cdot | \cdot \rangle$, are a formal definition of the overlap between two vectors, denoting the *inner product*^{II} of the desired TARGET STATE vector and the simulated CURRENT STATE of the system.

Following its publication in 2005, GRAPE immediately found applications in creating broadband pulses designed with BEBOP, as any optimal control method

^{I}also termed *Gradient Assisted Pulse Engineering*

^{II}The inner product, $\langle \cdot | \cdot \rangle$, is a generalisation of the scalar product between two vectors. In this context, $|\cdot\rangle$ is a row vector and $\langle\cdot|$ is a column vector, using the Dirac bra-ket notation [55].

should, to systems requiring robust excitation pulses – pulses that are tolerant to miscalibration of the nominal pulse power. The GRAPE method of optimal control can easily be modified to find broadband pulse excitation with tolerance to radio-frequency (B_1) miscalibration/inhomogeneity [158]. Although the system and optimal control task are simple, transferring longitudinal magnetisation of an uncoupled spin to transverse magnetisation, the example shows the possible flexibility of GRAPE, where rectangular pulses alone would fail to be adequately robust.

Transferring a single initial to final magnetisation component, point-to-point state transfer, may not entirely describe an adequate desired final state of the system [159, 160]. Simple *point-to-point*^{I} optimal control solutions can be used within the framework of finding *universal rotation* pulses, required for mixing or refocusing pulses [198]. *Universal rotation* pulses are solutions to problems where each of the three magnetisation vectors of the Bloch equation must have a defined rotation [200, 277]. When these universal rotations are the target state of a GRAPE simulation, the resulting pulse shapes are found to exhibit symmetry around the centre of the pulse [157]. Designing universal rotation pulses to be robust is called Broadband Universal Rotation By Optimised Pulses (BURBOP) [134, 135, 285].

The CPMG sequence (Carr-Purcell-Meiboom-Gill [39, 208]), useful in investigation of relaxation rates^{II}, was redesigned with GRAPE resulting in universal rotations. This application extended the CPMG robustness to a larger range of static B_0 resonance offsets and B_1 pulse power miscalibration [21]. Robust shaped pulses, intended to be direct replacements for hard $\pi/2$ and π pulses, were designed using the GRAPE method [226]. Furthermore, broadband optimal solutions for heteronuclear spin systems can be designed to compensate for coupling evolution over the duration of a pulse shape, with the strategy of using a point-to-point optimisation with its time reversal for finding universal rotation pulses [70].

Spin systems simulated with GRAPE speed up calculation of coherence transfer in larger spin and pseudospin chains [276]. Further to this, a method of using GRAPE with partitioned subsystems used to describe a large spin system proves effective in finding optimal control solutions for very large spin systems [255]. The long-range state transfer can be improved with the temporary occupation of higher order states for larger systems [155].

^{I}also termed *state-to-state* problems; these define an initial state and a target state each as single, or linear combination of, members of the basis set

^{II}Relaxation is called *decoherence* in some literature

Pulses produced by GRAPE may be difficult to implement on real hardware. Hardware constraints may interpolate a shape to a smooth realisation of pulses, rather than the square pulses produced by the GRAPE method. Smooth, shaped pulses can be produced as analytic functions with a modification to the fidelity functional [284]. Alternatively, this hardware constraint can be accounted for in simulations by increasing the discretisation of time evolution above that of the discrete control pulses [216], using experimentally obtained transfer matrices within GRAPE [20], or by employing a feedback control loop after a designed pulse sequence is implemented [69].

§ 1.5.1 Solid-state nuclear magnetic resonance

When NMR is performed on a solid powder, the anisotropic spin interactions result in large line broadening of the spectrum. Spinning the sample at the magic angle^{1} relative to the static B_0 axis averages out many of these interactions, resulting in sharper spectral line shapes [73]. This method is termed MAS (Magic Angle Spinning), and can be useful for investigating dipolar and quadrupolar systems [90].

In the same year as the publication of the GRAPE method, it found application to solid-state MAS NMR, accounting for anisotropic components of spin interactions, sample spinning and specifically addressing problems of B_1 inhomogeneity [322] within simulation. Design of pulses robust to hardware miscalibration of B_1 averages the GRAPE method over a range of B_1 power levels, in effect, performing a GRAPE simulation at each power level, then averaging the fidelities and the gradients [322]. A range of chemical shift offsets can be treated in a similar way to B_1 miscalibration [286]. Chemical shift offsets, B_1 inhomogeneity, and dispersion from powder averaging, can also be simulated with the GRAPE framework [310], in this case using effective Hamiltonians. A combination of optimal control pulses and average Hamiltonian theory can be used to analyse and then remove unwanted second and third order coupling terms produced by optimal control solutions [18].

A method of solid-state NMR can take advantage of the *Overhauser effect* [233] to transfer polarisation from unpaired electrons to nearby nuclei, in a method termed DNP (Dynamic Nuclear Polarisation). The framework to control DNP systems [113] is set out in [150, 207], and simple DNP systems can be handled with GRAPE to give optimal solutions [129] in the presence of relaxation [244]. Optimal control

^{1}the magic angle is $\theta_{\text{magic}} = \arccos \frac{1}{\sqrt{3}} = 54.7^\circ$

proves useful in combining incoherent and coherent transfer schemes, investigated in a system of one electron and two nuclear spins [243].

Again in the context of solid-state NMR, optimal pulses found use in experiments with labelled proteins [119, 146]. Biomolecular samples, where high power control pulses can overheat the sample, dictate the need for low power pulse solutions. These can be designed with GRAPE by penalising high power pulse solutions [147, 223], and can be extended to $^2\text{H} - ^{13}\text{C}$ cross-polarisation MAS experiments [328]. The quadratic power penalty is also used in calculating optimal pulses for conversion of parahydrogen induced longitudinal two-spin order to single spin order [27], and in designing robust pulses for transfer of magnetisation to singlet order [178]. Further radio frequency power and amplitude restriction are investigated for broadband excitation pulses [162].

Methods of simulating solid-state MAS to find optimal control pulses will be given in §7 and an overview of useful penalty methods used in conjunction with GRAPE will be given §6.

§ 1.5.2 Nuclear magnetic resonance imaging

A useful application of NMR in medicine is non-invasive, macroscopic imaging of the human body [73]. Termed MRI (Magnetic Resonance Imaging), the method uses magnetic field gradients to record a projection of proton density such that resonant frequencies are a linear functions of spatial coordinates [179]. Extending GRAPE into an average optimisation over an ensemble of Hamiltonians can compensate for the dispersion in the system dynamics in MRI simulations [156, 192].

When radio-frequency pulse durations are similar to relaxation rates of quadrupolar nuclei, as in the case of MRI applications, previous work on optimal control with MAS [322] fails to be relevant. This was investigated, resulting in a novel method of using GRAPE [183]. The idea is to find optimal solutions with GRAPE for long total pulse duration, then decreasing the total pulse duration but using the same shape. When the total pulse duration was reduced to less than the inverse of the quadrupolar frequency, the resulting pulse shapes were simple, resembling two discrete pulses separated by a delay. Results from that work showed an increase in signal intensity of a spin- $3/2$ central transition while suppressing its satellite transitions. In the context of MRI, this is useful in separating the signal of ^{23}Na in cartilage from that of free ^{23}Na within the image. Although found by optimal

control, the solutions to the problem indicated simple pulse shapes that do not need optimal control [184]. Further investigation proved optimal control theory found its use in designing robust pulses [185], where simple pulse shapes could not. Robust pulses were designed with GRAPE to excite the central transition of arbitrary spin- $3/2$ systems with static powder patterns [230]. DFT (Density Functional Theory)^{I} calculations can be used to predict the necessary interaction parameters for these GRAPE simulations [232], giving a very general approach to numerically finding optimal control pulses.

A practicable realisation of an efficient optimal control algorithm for MRI purposes should be fast, working in a small window of time, to be able to take patient specific distortions into account. The optimisation should be performed as the patient waits in the instrument, and timing is critical – should the patient move during the optimal control simulation, the distortions also change and the optimal control solution becomes ineffective [319]. Furthermore, advanced arrangements of control coils used in modern MRI^{III} need to be taken into account when designing optimal pulses [205]. In comparison to conventional adiabatic pulses, optimal control pulses can achieve similar results with reduced SAR^{III} [186] and within local and global SAR limits [317, 318].

§ 1.5.3 Quantum information

The research area of quantum information processing is based using two-level quantum systems, and can be thought of as extending the language of classical computation to these quantum bits [224, 332]. Just as the language of classical computation is based on circuitry and logic gates, so too is the area of quantum information processing. The difference between classical bits and quantum bits is that quantum bits can exist in a superposition of the classical binary states.

Although optimal control was applied to the area of quantum information before the advent of GRAPE, and there have been many applications of optimal control to this area, this review is intended to show applications of GRAPE to magnetic resonance systems – specifically to ensembles of spins^{IV}. As such, only a modest

^{I}a popular computational method to investigate electronic structure.

^{III}the arrangement is called pTX (parallel transmission), and can have an arrangement of up to 6 control coils, rather than the conventional 2 in MRI.

^{IV}Patient safety guidelines have strict limitations on SAR (specific-absorption-rate) – local hot-spots of temperature, due to the distribution of SAR when using pTX, will be different for every patient and every scan.

^{IV}Magnetic resonance on single spins is an emerging area, but beyond the scope of this work.

number of applications of GRAPE to this area will be mentioned; those the author judges general enough to have use in optimal control of an ensemble system in magnetic resonance.

Compared with the previous best methods for control of superconducting two-level systems, GRAPE can achieve lower information leakage, higher fidelity, and faster gates [294]. A strategy for reducing information leakage entails using an extra *control channel*, proportional to the derivative of the main control, within GRAPE [217]. A chain of 3 coupled superconducting two-level systems can use GRAPE to reduce unwanted “crosstalk” between the two outer two-level systems, so mediating transfer in a serial manner [114]. High fidelity spin entanglement can be achieved with universal rotations designed with GRAPE, demonstrated experimentally for nitrogen-vacancy centres in diamond [56].

The robust nature of GRAPE has been shown to be tolerant to errors in pulse length and off-resonance pulses for nitrogen-vacancy centres in diamond [56, 257], atomic vapour ensembles [271], and design of fault tolerant quantum gates using a modified GRAPE optimisation update rule [225, 267]. Time optimal solutions [152] giving the minimal pulse length to achieve high fidelity transfer of Ising coupled cluster states are verified with GRAPE simulated at different pulse lengths [76], with exponential falloff below a threshold pulse length [155]. Realistic pulse implementation can be accounted for in the form of power and rise-time limits [75]. Optimal GRAPE solutions for these realistic pulses are realised by averaging optimisation over individual detunings of resonance frequencies (with longitudinal magnetisation control), with optimal pulses using only a single x -magnetisation control.

Although the GRAPE method is designed for closed systems, in the case of coupling to an environment the method was advanced to open non-Markovian [249], and open Markovian systems [277]. Control pulses that maximise transfer starting with a high-temperature thermal ensemble are automatically optimal for lower-temperature initial ensembles [324]. Further, the control landscape of an open system is similar to a large closed system [335] and there is evidence that noise can be suppressed with control [6].

§ 1.5.4 Electron spin resonance

The particularly challenging application of optimal control to ESR (Electron Spin Resonance)^{1} includes the problem of the pulse response function of the microwave setup – the pulses seen by the sample differ from those sent to the pulse shaping hardware. The transfer matrix (response function) may be measured either by adding an antenna to the resonator [293], or by using a sample with a narrow ESR line to pick up the intensity of each spectral component [145]. Quasi-linear responses, such as the phase variation across the excitation bandwidth in nutation frequency experiments, can be described with additional transfer matrices [58]. Once measured, the transfer matrix can be used within an optimal control algorithm to transform pulse shapes before trajectory time evolution is calculated, then be used to design broadband ESR excitation pulses [293].

This relatively new area of applying optimal control pulses to ESR systems will be investigated in §8, finding a complimentary approach to those mentioned in the previous paragraph.

§ 1.6 Modified methods

Effective Hamiltonian methods [310] and the GRAPE algorithm [154] have showed great success in finding optimal control solutions. However, more powerful algorithms have the potential to take full advantage of optimal control. Further modifications of the GRAPE method spawned two useful optimal control methods, *optimal tracking* and *cooperative pulses*.

§ 1.6.1 Optimal tracking

Optimal tracking is a modified GRAPE method that takes an input control field that will approximate a desired output trajectory rather than a state [221]. Heteronuclear decoupling is an example of a limitation of the original GRAPE algorithm. The control problem involves taking a system to a desired intermediate state then

^{1}also termed EPR (Electron Paramagnetic Resonance). ESR is used in this thesis to avoid conflict with the quantum mechanics term “EPR-pair” derived from the Einstein-Podolsky-Rosen thought experiment [71]

back to the original state. In this context, defining the cost functional as the overlap between the initial and final states is not enough, since the initial and desired final states are the same – the system may simply do nothing and produce a maximal overlap while not penetrating the intermediate state. As an extension of the GRAPE algorithm, *optimal tracking* [221] propagates the system to an intermediate desired state while simultaneously propagating backward from the final state to this desired intermediate state – so ensuring that the initial and final states are always the same. *Optimal tracking* used for heteronuclear decoupling sequences, with linear offset-dependent scaling factors, can encode chemical shift correlations from an indirect dimension [266, 337].

§ 1.6.2 Cooperative pulses

Cooperative pulses take into account multi-pulse experiments, generalising the experimental method of phase cycling and cooperatively compensating for each pulse imperfection [24]. By generalising the GRAPE method, it can be applied to broadband and band-selective experiments [24], quantum filter type experiments [25], and pseudo-pure state preparation in the context of quantum information processing [329].

§ 1.7 Other optimal control methods

Krotov type methods are algorithms of optimal control [163] that were developed in parallel to the GRAPE method reviewed above. The methods were successfully developed for application to laser spectroscopy [301] and general two-level quantum systems [338]. Krotov methods recursively find pulses necessary to perform a state-transfer problem by solving the Lagrange multiplier problem [240]. The algorithm was recently formulated in the context of magnetic resonance [207] and is an alternative to GRAPE. It was further developed to a hybrid type method, including a quasi-Newton character to improve convergence [72]. The total pulse time of this method can be critical in finding high fidelity solutions to optimal control problems, and a characteristic threshold must be passed to find these solutions using Krotov type methods [36]. Krotov based methods can appear unattractive in that they need a seemingly arbitrary constant in the form of a Lagrange multiplier – and although the constant is the same for similar systems, any new system needs investigation to find its own Lagrange multiplier.

The CRAB (Chopped Random Basis) algorithm shows promise in finding optimal control solutions to many-body problems [35, 59]. Specifically, the method lends itself to *tensor-network* descriptions of the system, also known as time-dependent *density matrix renormalisation group* [269] or the *tensor train formalism* [262] – an almost approximation-free description. An extension of the algorithm avoids local extrema by reformulating pulse bounds [247]. Pulse shapes are described by Fourier harmonics and these become the variables of optimisation. Calculating the gradient elements in this formalism was expected to be prohibitively expensive, therefore the method uses a gradient-free optimisation method such as the Nelder-Mead simplex method [220]. A further modification includes gradient calculations in the GOAT method (Gradient Optimisation of Analytical controls) [199] giving improved convergence.

A method for efficiently calculating gradients will be presented in §5.

§ 1.8 Numerical computation

A number of public software packages include functionality to perform optimal control calculations for magnetic resonance problems. A selection is listed below:

Dynamo [200, 275]

Using Matlab, includes GRAPE, Krotov, and hybrid methods for quantum optimal control simulations of open and closed systems.

QuTiP [138, 139]

Using Python, includes the GRAPE and CRAB algorithms using L-BFGS and Nelder-Mead algorithms, respectively.

Simpson [13]

The release of the magnetic resonance simulation package SIMPSON progressed to include optimal control in the form of the GRAPE method [309, 311]. Includes the ability to simulate liquid- and solid-state NMR, MRI, and quantum computation. Simpson is coded in C with the Tcl scripting interface.

Spinach [130]

In the context of optimal control, *Spinach* solves the Liouville-von Neumann equation using state-space restriction techniques in an irreducible spherical

tensor basis using Matlab. GRAPE-L-BFGS [86] and Krotov methods were included before the author became involved in the project^{I}.

The first systematic look at the algorithmic complexity of the GRAPE algorithm [154] was published in [111]. Efficient implementation would take advantage of inherent parallelisation of the GRAPE algorithm. The bottleneck in computation was identified as performing a matrix exponential – a number of approaches to numerically compute this were compared: scaling and squaring based on the Padé approximation and series expansion with either the Taylor expansion or Chebyshev polynomials [10, 278].

The obvious way to create a gradient for a gradient ascent algorithm is to define it by a central finite-difference approximation – however, for anything but small systems, this gives a computationally expensive calculation^{II} – making optimal control of larger systems impractical. The need for analytic gradients is paramount for any gradient ascent method to become useful in larger systems [169]. Based on an auxiliary matrix formalism, analytic gradient elements can be calculated by taking the exponential of a 2×2 triangular block matrix, with the drift Hamiltonian on the diagonal and the control operator on the off-diagonal [81].

A Quasi-Newton optimisation of GRAPE, using the ℓ -BFGS algorithm, was developed to make full use of the gradient history to give second-order derivative information and improve convergence to superlinear [86]. A Newton-Raphson root finding approach, using the conditioned Jacobian matrix of first-order partial derivatives as a least-squares optimisation problem [84], improved convergence to quadratic for a chain of five coupled two-level systems using two controls. The Jacobian matrix is also used in Krylov-Newton methods for closed spin systems, showing impressive convergence without calculation of a full Hessian [41].

Many examples in the literature above, when simulating high dimensional spin systems, produce optimal control solutions which can be described to appear like “noise”. Trajectory analysis of the “noisy” pulse sequence evolution shows an underlying order to the spin dynamics [167], and a time-frequency representation of the pulse shapes reveals underlying patterns which are not apparent in a conventional Cartesian representation of pulses [161].

^{I}The *Spinach* software package is used and developed by the author of this document.

^{II}A two point finite difference requires two functional evaluations, a four point requires four etc. whereas an analytic gradient would require one functional evaluation by definition.

§ 1.9 Summary

Optimal control has become a useful tool in exploring the limits of, among other technologies, magnetic resonance spectroscopy and imaging. The design of pulsed magnetic resonance experiments with either composite pulses, shaped pulses, or adiabatic pulses has shown improvement when designed with optimal control methods.

The GRAPE method of optimal control has become one of the leaders in its field for finding computationally efficient solutions, able to find state-of-the-art pulse sequences for large and complicated spin systems. Furthermore, pulse sequences produced can be made robust to experimental variability.

In addition to becoming an indispensable tool of chemical spectroscopy, the emerging area of optimal control of MRI shows promise in reducing medical costs at the same time as increasing diagnostic abilities. And it is this last point that becomes the motivation of the work presented in this thesis – finding more accurate optimal control solutions with decreased computational complexity – with the vision of large, highly parallel computational infrastructure installed at medical facilities, real-time, bespoke optimal control solutions will become affordable. With reduced experiment time on the expensive apparatus, and increased throughput: the dream of scientific facilitators working in business and finance will become more real.

§ 2

Magnetic Resonance Theory

A youth who had begun to read geometry with Euclid, when he had learnt the first proposition, inquired, “What do I get by learning these things?” So Euclid called a slave and said “Give him three pence, since he must make a gain out of what he learns.”

– Joannes Stobaeus, PHYSICAL AND MORAL EXTRACTS

Optimal control can be thought of as the study of controlling a system to find an optimally desired solution. A formal definition of optimal control will be given in §3, culminating in the GRAPE method. §1 was a review of published literature on the subject of optimal control applied to magnetic resonance systems. Normally, when studying magnetic resonance systems, an ensemble of particle *spins* are manipulated to give molecular information of how these *spins* interact with one another.

Magnetic resonance spectroscopy classically uses rectangular pulses of electromagnetic radiation to manipulate spins located within a static magnetic field. Non-rectangular shaped pulses are also used in magnetic resonance with a review given in [89]. The idea of applying optimal control to magnetic resonance is to optimise desired experimental outcomes with respect to the pulse shape.

This chapter will introduce features of magnetic resonance theory that will be used within this thesis. Much of this chapter is taken from standard textbooks on quantum mechanics [7, 175, 209, 300], nuclear magnetic resonance (NMR) [1, 73, 131] and electron spin resonance (ESR) [33, 279].

§ 2.1 Angular momentum

Angular momentum in quantum mechanics can be derived from many starting points. Here, the starting point is the Schrödinger equation and Euclidean geometry.

§ 2.1.1 Schrödinger's equation

In 1926 Erwin Schrödinger formulated the famous Schrödinger equation [273] in an attempt to find solutions to a Hamiltonian equation using Louis de Broglie's wave-particle duality [29]. The Hamiltonian, $\hat{\mathcal{H}}$, is an energy operator which dictates dynamics in the equation of motion of that system in the form of a Hamiltonian equation. If $|\Psi(t)\rangle$ describes the state of a system at a particular time t , then the time-dependent Schrödinger equation is

$$\frac{\partial}{\partial t}|\Psi(t)\rangle = -i\hat{\mathcal{H}}|\Psi(t)\rangle \quad (2.1)$$

It is common to measure energy eigenvalues in angular frequency units within the area of magnetic resonance, and the usual factor of \hbar is therefore dropped from Eq. (2.1). The state $|\Psi(t)\rangle$ can be expanded to a complete orthonormal basis

$$|\Psi(t)\rangle = \sum_{i=1}^n c_i(t)|\Psi_i\rangle, \quad i = 1, 2, \dots, n \quad (2.2)$$

where the time dependence is now described with the coefficients $c_i(t)$ of the complete orthonormal basis states Ψ_i . This basis spans a *Hilbert space* of dimension n .

§ 2.1.2 Euclidean principle of relativity

Quantum mechanics is built on the principle that space is subject to the laws of Euclidean geometry – and that space itself is isotropic and homogeneous. This assumption is called the *Euclidean principle of relativity*. A physical interpretation is – in the absence of external fields and perturbations the effect of moving a physical system, such as experimental apparatus, should not result in measurements that depend on the location or orientation of the experiment. Or more precisely;

a translation or rotation of the coordinate system should not change the Hamiltonian describing the energy of particles within the system. The Hamiltonian is invariant under rotation and translation.

For a rotation operation denoted by \hat{R} the mapping of a state vector Ψ to Ψ' is

$$|\Psi\rangle \mapsto \hat{R}|\Psi\rangle = |\Psi'\rangle \quad \Rightarrow \quad \langle \Psi | \hat{R}^\dagger \hat{\mathcal{H}} \hat{R} | \Psi \rangle = \langle \Psi | \hat{\mathcal{H}} | \Psi \rangle \quad (2.3)$$

This relation must hold for any state vectors, and therefore \hat{R} must commute with the system Hamiltonian $\hat{\mathcal{H}}$

$$\hat{R}^\dagger \hat{\mathcal{H}} \hat{R} = \hat{\mathcal{H}} \quad \Rightarrow \quad \hat{R}^{-1} \hat{\mathcal{H}} \hat{R} = \hat{\mathcal{H}} \quad \Rightarrow \quad [\hat{\mathcal{H}}, \hat{R}] = 0 \quad (2.4)$$

Using the Schrödinger Eq.(2.1), and the measurement of an observable^{I} property of the system being its expected value^{II} – the following relation can be derived

$$-i\langle \Psi | [\hat{\mathcal{H}}, \hat{R}] | \Psi \rangle = 0 \quad (2.5)$$

showing the conservation of the observable property associated with the rotation \hat{R} .

§ 2.1.3 Orbital angular momentum

The property conserved for the above rotation, from the assumption of isotropic space, is *angular momentum*. The quantum mechanical angular momentum operator, \hat{L} , is analogous to its classical vector counterpart [174]

$$\hat{L} = \hat{r} \times \hat{p} = \hat{r} \times \frac{\hbar}{i} \nabla, \quad \nabla \triangleq \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{pmatrix} \quad (2.6)$$

The Cartesian components of the angular momentum operator can be derived by considering infinitesimal rotations giving

$$i\hat{L}_x = \left(y \frac{\partial}{\partial z} - z \frac{\partial}{\partial y} \right), \quad i\hat{L}_y = \left(z \frac{\partial}{\partial x} - x \frac{\partial}{\partial z} \right), \quad i\hat{L}_z = \left(x \frac{\partial}{\partial y} - y \frac{\partial}{\partial x} \right) \quad (2.7)$$

^{I}An Hermitian operator that possesses a complete set of eigenfunctions is an *observable*.

^{II}The expected value of an hermitian operator is $\langle \Psi | \hat{A} | \Psi \rangle$

where the constant unit of \hbar is dropped, as is normal in the subject of magnetic resonance. The solution to a differential equation for the infinitesimal rotation φ is

$$\frac{d}{d\varphi} \hat{R}(\varphi) = -i\hat{L}_z \hat{R} \quad \Rightarrow \quad \hat{R}(\varphi) = e^{-i\hat{L}_z\varphi} \quad (2.8)$$

This can be extended to show all rotation operators are exponentials of the angular momentum operators, and the angular momentum operators are the basis of the Lie algebra corresponding to the Lie group of rotations. By inspection of Eq. (2.7), the following commutation relations can be formed:

$$i\hat{L}_x = [\hat{L}_y, \hat{L}_z], \quad i\hat{L}_y = [\hat{L}_z, \hat{L}_x], \quad i\hat{L}_z = [\hat{L}_x, \hat{L}_y] \quad (2.9)$$

where the commutator of two elements a and b defines a Lie algebra, with $[a, b] = ab - ba$. The square of the angular momentum, the modulus, can be constructed from the angular momentum operator components:

$$\hat{L}^2 = \hat{L}_x^2 + \hat{L}_y^2 + \hat{L}_z^2 \quad (2.10)$$

and this operator, called the Casimir operator, does commute with each of the angular momentum operators

$$[\hat{L}^2, \hat{L}_x] = 0, \quad [\hat{L}^2, \hat{L}_y] = 0, \quad [\hat{L}^2, \hat{L}_z] = 0 \quad (2.11)$$

A convenient set of non-Hermitian operators, named the raising and lowering operators, can be defined in terms of \hat{L}_x and \hat{L}_y

$$\hat{L}_+ = \hat{L}_x + i\hat{L}_y, \quad \hat{L}_- = \hat{L}_x - i\hat{L}_y \quad (2.12)$$

with the following commutation relations involving \hat{L}_z

$$[\hat{L}_+, \hat{L}_-] = 2\hat{L}_z, \quad [\hat{L}_z, \hat{L}_+] = \hat{L}_+, \quad [\hat{L}_z, \hat{L}_-] = -\hat{L}_- \quad (2.13)$$

§ 2.1.4 The uncertainty principle

Given an Hermitian operator \hat{L} , representing an observable property of the system in a state $|\Psi\rangle$; the uncertainty in a measurement of that observable property can

be represented as a standard deviation [330]:

$$\Delta \hat{L} = \sqrt{\langle \hat{L}^2 \rangle - \langle \hat{L} \rangle^2} = \sqrt{\langle \psi | \hat{L}^2 | \psi \rangle - \langle \psi | \hat{L} | \psi \rangle^2} \quad (2.14)$$

Heisenberg's uncertainty principle [122] states that for two hermitian operators:

$$\Delta \hat{A} \Delta \hat{B} \geq \frac{\hbar}{2} \left| \langle [\hat{A}, \hat{B}] \rangle \right| \quad [\hat{A}, \hat{B}] = ic \quad (2.15)$$

where c is a constant. This can be seen with angular momentum operators applied to the Heisenberg uncertainty relation in atomic units, where $\hbar = 1$:

$$\begin{aligned} \Delta \hat{L}_x &= \sqrt{\langle \hat{L}_x^2 \rangle - \langle \hat{L}_x \rangle^2} \\ \Delta \hat{L}_y &= \sqrt{\langle \hat{L}_y^2 \rangle - \langle \hat{L}_y \rangle^2} \\ \Rightarrow \quad \Delta \hat{L}_x \Delta \hat{L}_y &\geq \frac{1}{2} \left| \langle [\hat{L}_x, \hat{L}_y] \rangle \right| = \frac{1}{2} \left| \langle \hat{L}_z \rangle \right| \end{aligned} \quad (2.16)$$

The angular momentum operator components do not commute with each other. This indicates that all components of angular momentum cannot be simultaneously defined, unless all equal zero – which sets it apart from linear momentum^{11}. However, the modulus of the angular momentum, the Casimir operator, can have a definite value at the same time as each component of the angular momentum and is therefore simultaneously observable.

§ 2.1.5 Spherical polar coordinates

The three Cartesian components of the angular momentum operators can be expressed in a spherical polar coordinate system, more natural than a Cartesian coordinate system when considering rotations from §2.1.2. Depicted in FIG. 2.1,

^{11}The result obtained by considering infinitesimal translations and the homogeneity of space, would give relations for the linear momentum operator.

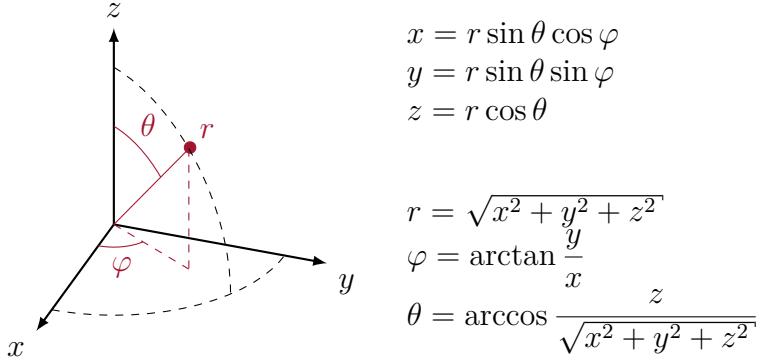


FIGURE 2.1: Diagram of a spherical coordinate system in relation to a Cartesian coordinate system.

their formulation is:

$$\hat{L}_x = -i \left(-\sin \varphi \frac{\partial}{\partial \theta} - \cos \varphi \cot \theta \frac{\partial}{\partial \varphi} \right) \quad (2.17)$$

$$\hat{L}_y = -i \left(\cos \varphi \frac{\partial}{\partial \theta} - \sin \varphi \cot \theta \frac{\partial}{\partial \varphi} \right) \quad (2.18)$$

$$\hat{L}_z = -i \frac{\partial}{\partial \varphi} \quad (2.19)$$

Similarly, the transform of the Casimir operator to spherical polar coordinates gives

$$\hat{L}^2 = -\frac{1}{\sin^2 \theta} \frac{\partial^2}{\partial \varphi^2} - \frac{1}{\sin \theta} \frac{\partial}{\partial \theta} \left(\sin \theta \frac{\partial}{\partial \theta} \right) \quad (2.20)$$

The simultaneous eigenvalues of \hat{L}^2 and \hat{L}_z are

$$\left. \begin{aligned} \hat{L}^2 |Y_\ell^m(\theta, \varphi)\rangle &= \ell(\ell+1) |Y_\ell^m(\theta, \varphi)\rangle \\ \hat{L}_z |Y_\ell^m(\theta, \varphi)\rangle &= m |Y_\ell^m(\theta, \varphi)\rangle \end{aligned} \right\} \quad \begin{aligned} \ell &\in \mathbb{N} \\ m &= -\ell, -\ell+1, \dots, \ell \end{aligned} \quad (2.21)$$

where the normalised, simultaneous eigenfunctions of the Casimir operator, \hat{L}^2 , and the z -projection operator, \hat{L}_z , are the spherical harmonics

$$Y_\ell^m(\theta, \varphi) = (-1)^{\frac{m+|m|}{2}} \sqrt{\frac{2\ell+1}{4\pi} \frac{(\ell-|m|)!}{(\ell+|m|)!}} P_\ell^m(\cos \theta) e^{im\varphi} \quad (2.22)$$

$$P_\ell^m(\xi) = (1-\xi^2)^{\frac{|m|}{2}} \frac{d^{|m|}}{d\xi^{|m|}} P_\ell(\xi) \quad (2.23)$$

where $P_\ell(\xi)$ are the Legendre polynomials which can be defined by the formula

$$P_\ell(\xi) = \frac{1}{2^\ell \ell!} \frac{d^\ell}{d\xi^\ell} \left[(\xi^2 - 1)^\ell \right] \quad (2.24)$$

ℓ is the total momentum quantum number, m is the projection quantum number, $\ell \in \mathbb{N}$, $m = -\ell, -\ell+1, \dots, \ell$.

Following Eqs. (2.17), (2.18) the transform of raising and lowering operators to spherical polar coordinates are

$$\begin{aligned} \hat{L}_+ &= e^{i\varphi} \left(\frac{\partial}{\partial\theta} + i \cot\theta \frac{\partial}{\partial\varphi} \right) \\ \Rightarrow \hat{L}_+ |Y_\ell^m\rangle &= \sqrt{\ell(\ell+1) - m(m+1)} |Y_\ell^{m+1}\rangle \end{aligned} \quad (2.25)$$

$$\begin{aligned} \hat{L}_- &= -e^{i\varphi} \left(\frac{\partial}{\partial\theta} - i \cot\theta \frac{\partial}{\partial\varphi} \right) \\ \Rightarrow \hat{L}_- |Y_\ell^m\rangle &= \sqrt{\ell(\ell+1) - m(m-1)} |Y_\ell^{m-1}\rangle \end{aligned} \quad (2.26)$$

The raising and lowering operators perform operations that increment and decrement the magnetic quantum number m .

§ 2.1.6 Intrinsic angular momentum

In dealing with the subject of magnetic resonance the idea of *intrinsic angular momentum* (as opposed to *orbital angular momentum* discussed above), termed *spin*, must first be introduced [236, 312]. The most direct evidence for this *spin* is the Stern-Gerlach experiment, proposed by Otto Stern [295]: Silver atoms are sent in a focused beam through a non-uniform magnetic field, B_0 [93]. Classically, the atoms should give a continuous trace, between $+\mu$ and $-\mu$, when they emerge from the magnetic field. However, observation of the emerging atoms shows a number of distinct traces, rather than the expected continuous trace, spaced equally between $+\mu$ and $-\mu$. Conventionally, μ is considered to be the magnetic moment of the particle. A diagram of the experiment is shown in FIG. 2.2.

This is attributed to the two spin states of atoms within the incident beam – *spin up* and *spin down*. The “up” and “down” could also be termed $+1$ and -1 , or α and β . The nomenclature is only to indicate that there is a quantised property of the silver atoms, having two states which become physically separated in the presence of an external magnetic field.

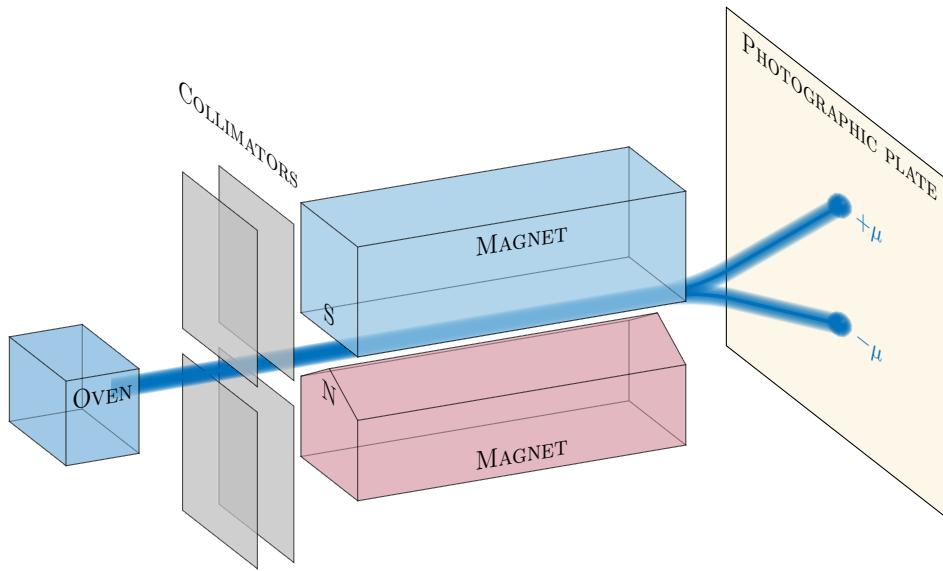


FIGURE 2.2: Schematic diagram of the Stern-Gerlach experiment showing a beam of silver atoms split by an inhomogeneous magnetic field. The classical picture of this experiment would show a continuous, rather than discrete, trace on the photographic plate.

This spatial splitting of the wave functions describing the silver atoms must be included in addition to, and independent of (commuting with), the spatial coordinate description of the wave function. The discrete spin variable describing the splitting in the Stern-Gerlach can take two values^{11}, +1 or -1.

Spin degrees of freedom can be split from the spatial degrees of freedom, forming a *spinor*:

$$|\Psi(\vec{r}, s)\rangle = \sum_{n,k} |\Psi_n(\vec{r})\rangle \otimes |\Psi_k(s)\rangle \quad (2.27)$$

The spin part of this construct can be described by a pair of orthogonal vectors projecting onto the two discrete values of the spin,

$$\chi = \begin{pmatrix} c_{+1/2} \\ c_{-1/2} \end{pmatrix} = c_{+1/2} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + c_{-1/2} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = c_{+1/2} |\alpha\rangle + c_{-1/2} |\beta\rangle \quad (2.28)$$

which must be normalised, requiring $\chi^\dagger \chi = |c_{+1/2}|^2 + |c_{-1/2}|^2 = 1$. In the case of a spin- $1/2$ particle, this basis can be used for describing the two spin eigenstates of the Stern-Gerlach experiment. Without derivation, we make the assertion that the spin quantum number can have $2s + 1$ values, and for a spin- $1/2$ particle these values are $s = \pm 1/2$. This imposes the condition that $s = 0, \frac{1}{2}, 1, \frac{3}{2}, 2, \dots$

^{11}for a spin- $1/2$ particle, which is the case for silver atoms

In analogy to orbital angular momentum, for intrinsic angular momentum a matrix representation can be formulated with raising and lowering operators to move between the $|\alpha\rangle$ and $|\beta\rangle$ states

$$\hat{S}_+|\alpha\rangle = |\mathbf{0}\rangle, \quad \hat{S}_+|\beta\rangle = |\alpha\rangle, \quad \Rightarrow \quad \hat{S}_+ = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \quad (2.29)$$

$$\hat{S}_-|\alpha\rangle = |\beta\rangle, \quad \hat{S}_-|\beta\rangle = |\mathbf{0}\rangle, \quad \Rightarrow \quad \hat{S}_- = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \quad (2.30)$$

and with analogy to orbital angular momentum, the raising and lowering operators can be formulated in terms of \hat{S}_x and \hat{S}_y :

$$\hat{S}_+ = \hat{S}_x + i\hat{S}_y, \quad \hat{S}_- = \hat{S}_x - i\hat{S}_y \quad (2.31)$$

The Cartesian operators can be constructed from the raising and lowering operators, and by requiring the eigenvalues of the z -component to be the allowed values of spin $\pm 1/2$, giving

$$\begin{aligned} \hat{S}_x &= \frac{\hat{S}_+ + \hat{S}_-}{2}, & \hat{S}_y &= \frac{\hat{S}_+ - \hat{S}_-}{2i}, & \hat{S}_z|\alpha\rangle &= +\frac{1}{2}|\alpha\rangle \\ \hat{S}_x &= \frac{1}{2} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, & \hat{S}_y &= \frac{1}{2} \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, & \hat{S}_z &= \frac{1}{2} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \end{aligned} \quad (2.32)$$

These are called the Pauli matrices of a spin- $1/2$ particle. It is easy to see that the square of each of these operators is the same and proportional to the unit matrix, $\mathbb{1}$. Unlike orbital angular momentum, the square of spin is also proportional to the identity

$$\hat{S}^2 = \hat{S}_x^2 + \hat{S}_y^2 + \hat{S}_z^2 = \frac{1}{4}\mathbb{1} + \frac{1}{4}\mathbb{1} + \frac{1}{4}\mathbb{1} = \frac{3}{4}\mathbb{1} \quad (2.33)$$

For spin greater than $1/2$, the matrix dimension becomes larger, and similar operators can be constructed e.g. for spin-1 particles:

$$\hat{S}_x = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad \hat{S}_y = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & -i & 0 \\ i & 0 & -i \\ 0 & i & 0 \end{bmatrix}, \quad \hat{S}_z = \frac{1}{2} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (2.34)$$

The commutation relations follow by inspection:

$$[\hat{S}_x, \hat{S}_y] = i\hat{S}_z \quad [\hat{S}_z, \hat{S}_x] = i\hat{S}_y \quad [\hat{S}_y, \hat{S}_z] = i\hat{S}_x \quad [\hat{S}^2, \hat{S}_{\{x,y,z\}}] = 0 \quad (2.35)$$

§ 2.2 Irreducible representations of the rotation group

Returning to the *Euclidean principle of relativity*, space is required to be isotropic and homogeneous. The consequence is that when a state vector $|\Psi\rangle$ undergoes a rotation operation in space, \hat{R} , all probabilities must be invariant. Mathematically, this means that all inner products of two rotated states remain invariant

$$|\Psi\rangle \mapsto \hat{R}|\Psi\rangle = |\Psi'\rangle \quad (2.36)$$

$$|\langle \Psi' | \Phi' \rangle|^2 = |\langle \Psi | \Phi \rangle|^2 \quad (2.37)$$

for every pair of state vectors, Ψ and Φ . This mapping is called *isometry* and must be reversible. A second rotation results only in a phase change [209]:

$$|\Psi''\rangle = e^{i\varphi} |\Psi'\rangle \quad (2.38)$$

The transform of a sum of two vectors is equal to the sum of the transforms of two vectors. It follows that [209]

$$\langle \Psi''_a | \Psi''_b \rangle + \left(\langle \Psi''_a | \Psi''_b \rangle \right)^* = \langle \Psi_a | \Psi_b \rangle + \left(\langle \Psi_a | \Psi_b \rangle \right)^* \quad (2.39)$$

$$\text{Re} \langle \Psi''_a | \Psi''_b \rangle = \text{Re} \langle \Psi_a | \Psi_b \rangle \quad (2.40)$$

$$\text{Im} \langle \Psi''_a | \Psi''_b \rangle = \pm \text{Im} \langle \Psi_a | \Psi_b \rangle \quad (2.40)$$

$$\Rightarrow \left(\lambda | \Psi \rangle \right)'' = \lambda | \Psi'' \rangle \quad (2.41)$$

$$\left(\lambda | \Psi \rangle \right)'' = \lambda^* | \Psi'' \rangle \quad (2.42)$$

where $*$ has been used here to denote the complex conjugate. Eqs. (2.39), (2.41) are the properties of a linear operator, and Eq. (2.41) implies the transform is unitary. A unitary transformation operator \hat{U} satisfies $\hat{U}^\dagger \hat{U} = \hat{U} \hat{U}^\dagger = \mathbb{1}$ and gives a symmetry operation^{11}, an important property of a rotation. Eq. (2.42) characterises an antilinear operator, not considered in generating rotation operations, which is important in describing a system under *time reversal*.

^{11}A transformation that leaves the physics unaltered is called a *symmetry operation*, and a symmetry operation applied to all eigenstates of a basis must leave the Schrödinger equation invariant. This condition is satisfied if the system Hamiltonian, \hat{H} , commutes with the transform: $[\hat{H}, \hat{U}_R] = 0$

Spin does not rotate when a particle is rotated, but anisotropic interactions (see §2.4) do rotate. To represent these rotations, the following complex Hermitian matrices will become generators of the rotations

$$\hat{J}_1 \triangleq \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & i \\ 0 & -i & 0 \end{bmatrix}, \quad \hat{J}_2 \triangleq \begin{bmatrix} 0 & 0 & i \\ 0 & 0 & 0 \\ -i & 0 & 0 \end{bmatrix}, \quad \hat{J}_3 \triangleq \begin{bmatrix} 0 & i & 0 \\ -i & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (2.43)$$

A rotation from a state $|\Psi\rangle$ to $|\Psi'\rangle$ can be expressed as a unitary operator, $|\Psi\rangle = \hat{U}_R|\Psi'\rangle$. From the generators in Eq. (2.43) and the rotation in Eq. (2.38), the unitary rotation operators representing rotations through angles $\{\alpha, \beta, \gamma\}$ are

$$\begin{aligned} e^{i\hat{J}_1\alpha} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix} \\ e^{i\hat{J}_2\beta} &= \begin{bmatrix} \cos \beta & 0 & -\sin \beta \\ 0 & 1 & 0 \\ \sin \beta & 0 & \cos \beta \end{bmatrix} \\ e^{i\hat{J}_3\gamma} &= \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (2.44)$$

\hat{J} must satisfy the general commutation relations to give a well-defined transform:

$$[\hat{J}_i, \hat{J}_j] = i\epsilon_{ijk}\hat{J}_k, \quad \epsilon_{ijk} = \begin{cases} 1 & ijk \in \{123, 231, 312\} \\ -1 & ijk \in \{321, 213, 132\} \\ 0 & \text{all else} \end{cases} \quad (2.45)$$

where \hat{J} is a rotation operator and i, j, k take the values 1, 2, 3 and represent the Cartesian components x, y, z . Further to the unitary operator representing a rotation in Eq. (2.44), we also require the same rotation at further 2π angles

$$\hat{U}_R = \exp \left[-i\hat{J}(\varphi + 2\pi k) \right] = \exp \left[-i\hat{J}\varphi \right] \exp \left[-i\hat{J}2\pi k \right] \quad (2.46)$$

where k is an arbitrary integer. The additional factor in Eq. (2.46) has the effect of multiplying an eigenstate Y_ℓ^m by $(-1)^{2km}$, where m is the magnetic quantum number and the eigenvalue of an operator \hat{J}_z .

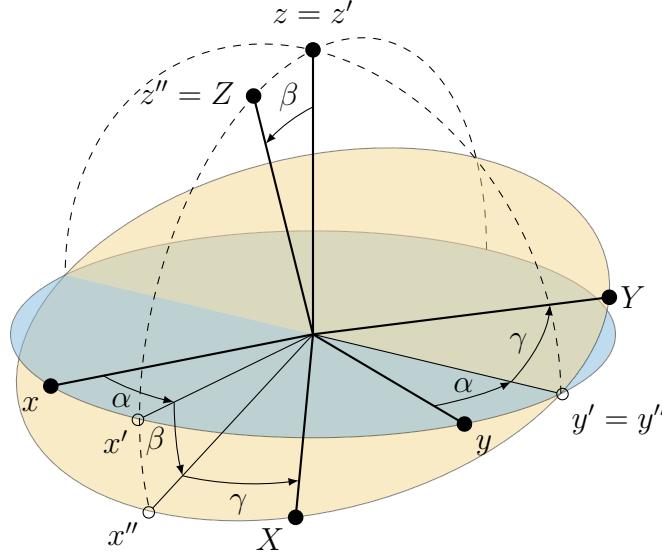


FIGURE 2.3: Euler angles: This rotation convention corresponds to ① rotation of angle α around the z -axis, ② rotation of angle β around the y' -axis, ③ rotation of angle γ around the z'' -axis.

Any rotation \hat{R} operation in 3D can be described by three *Euler angles*^{1}, shown in FIG. 2.3, which represent three consecutive rotations by angles α, β, γ [7],

$$\hat{R}(\alpha, \beta, \gamma) = \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \beta & 0 & -\sin \beta \\ 0 & 1 & 0 \\ \sin \beta & 0 & \cos \beta \end{bmatrix} \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.47)$$

§ 2.3 Liouville-von Neumann equation

Magnetic resonance is concerned with ensemble averages of spin particles. Simply describing the basis in a Hilbert space of §2.1.1 is not enough, and a more descriptive, larger space must be constructed.

Given the initial state of the system, $|\psi(0)\rangle$, the solution to this Eq. (2.1) at a future time $T > 0$ is

$$|\psi(T)\rangle = \exp_{(o)} \left[-i \int_0^T \hat{\mathcal{H}}(t) dt \right] |\psi(0)\rangle \quad (2.48)$$

^{1}Other rotation conventions exist, such as quaternions or directional cosine matrices, but Euler angles are shown here for convenience.

where $\exp_{(o)}[\dots]$ indicates a time ordered exponential [62, 63, 73, 300].

The density operator is a description of the state of a system ensemble

$$\hat{\rho}(t) = \sum_{i,j} p_{ij} |\psi_i(t)\rangle\langle\psi_j(t)| \quad (2.49)$$

$$= |\psi(t)\rangle\langle\psi(t)| \quad (2.50)$$

where p_{ii} are the associated probabilities for each state $\psi_i(t)$, normalised so that $\sum_i p_{ii} = 1$. The diagonal elements of $\hat{\rho}(t)$ are the probabilities of being in an eigenstate of the Hamiltonian, and the off-diagonal elements of $\hat{\rho}(t)$ indicate a superposition of states [73].

The equation of motion for the density operator can be found from the time-dependent Schrödinger equation:

$$\begin{aligned} \frac{\partial}{\partial t} \hat{\rho}(t) &= \frac{\partial}{\partial t} \left(|\psi(t)\rangle\langle\psi(t)| \right) = \left(\frac{\partial}{\partial t} |\psi(t)\rangle \right) |\psi(t)\rangle\langle\psi(t)| \left(\frac{\partial}{\partial t} \langle\psi(t)| \right) \\ &= -i\hat{\mathcal{H}}(t) |\psi(t)\rangle\langle\psi(t)| + i|\psi(t)\rangle\langle\psi(t)|\hat{\mathcal{H}}(t) \\ &= -i \left(\hat{\mathcal{H}}(t)\hat{\rho}(t) - \hat{\rho}(t)\hat{\mathcal{H}}(t) \right) \end{aligned}$$

Identifying the commutator, the equation of motion can be written as

$$\frac{\partial}{\partial t} \hat{\rho}(t) = -i [\hat{\mathcal{H}}(t), \hat{\rho}(t)] \quad (2.51)$$

This is known as the Liouville-von Neumann equation. In a similar manner to finding the solutions of EQ. (2.1) with EQ. (2.48), the solutions to EQ. (2.51) are

$$\hat{\rho}(T) = \hat{\mathcal{U}}(T)\hat{\rho}(0)\hat{\mathcal{U}}^{-1}(T) \quad (2.52)$$

where $\hat{\mathcal{U}}(T)$ is the time propagator over a time interval $[0, T]$ defined by

$$\hat{\mathcal{U}}(T) = \exp_{(o)} \left[-i \int_0^T \hat{\mathcal{H}}(t) dt \right] \quad (2.53)$$

§ 2.3.1 Superoperator algebra

The transform in EQ. (2.52) can be represented as the multiplication of three $n \times n$ matrices in the following schematic:

$$\begin{bmatrix} \rho_{11} & \rho_{12} & \cdots & \rho_{1n} \\ \rho_{21} & \rho_{22} & \cdots & \rho_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{n1} & \rho_{n2} & \cdots & \rho_{nn} \end{bmatrix} = \begin{bmatrix} & & & \\ & \hat{\mathcal{U}} & & \\ & \text{SIZE: } (n \times n) & & \end{bmatrix} \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} \end{bmatrix} \begin{bmatrix} & & & \\ & \hat{\mathcal{U}}^{-1} & & \\ & \text{SIZE: } (n \times n) & & \end{bmatrix} \quad (2.54)$$

The same transform in EQ. (2.52) can also be written as a matrix-vector multiplication, numerically more appealing than the product of three matrices, by using the linear transform of density matrix vectorisation:

$$\begin{pmatrix} \rho_{11} \\ \rho_{21} \\ \vdots \\ \rho_{n1} \\ \rho_{12} \\ \rho_{22} \\ \vdots \\ \rho_{n2} \\ \vdots \\ \rho_{1n} \\ \rho_{2n} \\ \vdots \\ \rho_{nn} \end{pmatrix} = \begin{bmatrix} & & & \\ & \hat{\mathcal{U}} & & \\ & \text{SIZE: } (n^2 \times n^2) & & \end{bmatrix} \begin{pmatrix} \sigma_{11} \\ \sigma_{21} \\ \vdots \\ \sigma_{n1} \\ \sigma_{12} \\ \sigma_{22} \\ \vdots \\ \sigma_{n2} \\ \vdots \\ \sigma_{1n} \\ \sigma_{2n} \\ \vdots \\ \sigma_{nn} \end{pmatrix} \quad (2.55)$$

The commutation relation in EQ. (2.51) can be represented as a matrix in itself, called a superoperator, and transforms the Liouville-von Neumann equation into a matrix-vector multiplication

$$\frac{\partial}{\partial t} \hat{\rho} = -i \hat{\mathcal{H}} \hat{\rho} \quad (2.56)$$

where the “double-hat” superoperator notation has been introduced, with the Liouville superoperator acting on another operator defined as a commutation $\hat{\mathcal{H}} \hat{\rho} = [\hat{\mathcal{H}}, \hat{\rho}]$, and is represented by a matrix of size $n^2 \times n^2$. Using this superoperator

representation EQ. (2.52) becomes

$$|\rho(T)\rangle = \exp_{(o)} \left[-i \int_0^T \hat{\mathcal{H}}(t) dt \right] |\rho(0)\rangle \quad \hat{\mathcal{H}} = \mathbb{1} \otimes \hat{\mathcal{H}} - \hat{\mathcal{H}}^\dagger \otimes \mathbb{1} + i \hat{\mathcal{R}} \quad (2.57)$$

where $\hat{\mathcal{R}}$ is the relaxation superoperator, describing damping and dissipations effects, and $\mathbb{1}$ is the identity matrix^{11}. The Liouville superoperator defined in this way, as a direct product, allows interacting composite systems described in a numerically efficient way.

Continuing from the spherical harmonics of §2.1.5, the vector space of Eqs. (2.21), (2.25), (2.26) can be decomposed to $(2\ell + 1)$ -dimensional subspaces, which are invariant under rotation. Under a rotation, a state in the spherical polar coordinate basis transforms as

$$\hat{R} |Y_\ell^m\rangle = \sum_{m'=-\ell}^{\ell} |Y_\ell^{m'}\rangle \mathfrak{D}_{m'm}^{(\ell)}(\alpha, \beta, \gamma) \quad (2.58)$$

where the matrix $\mathfrak{D}_{m'm}^{(\ell)}(\alpha, \beta, \gamma)$ is the *Wigner rotation matrix* (FIG. A.1) with the elements described by

$$\mathfrak{D}_{m'm}^{(\ell)}(\alpha, \beta, \gamma) = \langle Y_\ell^{m'} | \hat{R} | Y_\ell^m \rangle \quad (2.59)$$

shown explicitly in §A. A rotation in one of these subspaces can be expressed by an *irreducible spherical tensor operator* $\hat{T}_{\ell m}$ in a matrix representation, which is defined as [259]

$$\hat{R} \hat{T}_{\ell m} \hat{R}^{-1} = \hat{\mathcal{R}} \hat{T}_{\ell m} = \sum_{m'=-\ell}^{\ell} \hat{T}_{\ell m'} \mathfrak{D}_{m'm}^{(\ell)}(\alpha, \beta, \gamma) \quad (2.60)$$

where α, β, γ are Euler angles in EQ. (2.47). For the infinitesimal rotations used in §§2.1.2, 2.1.3, \hat{J} , EQ. (2.60) simplifies to

$$[\hat{J}, \hat{T}_{\ell m}] = \sum_{m'=-\ell}^{\ell} \hat{T}_{\ell m'} \langle Y_\ell^{m'} | \hat{J} | Y_\ell^m \rangle \quad (2.61)$$

^{11}The identity matrix, $\mathbb{1}$, is a matrix with ones on every diagonal element and zeros everywhere else.

giving an alternate form of Eqs. (2.21), (2.25), (2.26) as commutation relations for the spherical tensor operator:

$$[\hat{J}_z, \hat{T}_{\ell m}] = m \hat{T}_{\ell m} \quad (2.62)$$

$$[\hat{J}_+, \hat{T}_{\ell m}] = \sqrt{\ell(\ell+1) - m(m+1)} \hat{T}_{\ell m+1} \quad (2.63)$$

$$[\hat{J}_-, \hat{T}_{\ell m}] = \sqrt{\ell(\ell+1) - m(m-1)} \hat{T}_{\ell m-1} \quad (2.64)$$

The single particle spherical tensor operators of rank-1 are

$$\hat{T}_{00} = \mathbb{1} \quad (2.65)$$

$$\hat{T}_{10} = \hat{L}_z \quad (2.66)$$

$$\hat{T}_{1\pm 1} = \mp \frac{1}{\sqrt{2}} \hat{L}_{\pm} \quad (2.67)$$

The full list of irreducible spherical tensor operators is given in §A FIG. A.2. Within the software toolbox *Spinach* [130], all rotations are active, meaning the system is rotated and not the coordinates.

§ 2.3.2 Composite systems

The total wavefunction of two non-interacting uncorrelated subsystems is a *direct product* of the subsystems^{I}:

$$|\psi_{1,2}\rangle = |\psi_1\rangle \otimes |\psi_2\rangle \quad (2.68)$$

and the density matrices are

$$\hat{\rho}_{1,2} = |\psi_{1,2}\rangle\langle\psi_{1,2}| = (|\psi_1\rangle\langle\psi_1|) \otimes (|\psi_2\rangle\langle\psi_2|) = \hat{\rho}_2 \otimes \hat{\rho}_2 \quad (2.69)$$

For a system of k non-interacting uncorrelated subsystems, the system is described by a chain of k direct products:

$$\hat{\rho}_{1,2,\dots,k} = \hat{\rho}_2 \otimes \hat{\rho}_2 \otimes \dots \otimes \hat{\rho}_k \quad (2.70)$$

^{I}Also called the Kronecker product.

Similarly, when the angular momentum of a composite system is represented by a direct product space, so to are the normalised simultaneous eigenvectors:

$$|Y_{\ell_1, \ell_2}^{m_1, m_2}\rangle = |Y_{\ell_1}^{m_1}\rangle \otimes |Y_{\ell_2}^{m_2}\rangle \quad (2.71)$$

and the state for n particles can be expressed with n direct products

$$|Y_{\ell_1, \ell_2, \dots, \ell_n}^{m_1, m_2, \dots, m_n}\rangle = |Y_{\ell_1}^{m_1}\rangle \otimes |Y_{\ell_2}^{m_2}\rangle \otimes \dots \otimes |Y_{\ell_n}^{m_n}\rangle \quad (2.72)$$

which form the basis of this direct product space.

The operators for a multi-particle system are direct products, where the operator of the k^{th} subsystem occurs at the k^{th} position in the direct product chain e.g.

$$\begin{aligned} \hat{S}_x^{(k)} &= \mathbb{1} \otimes \mathbb{1} \otimes \dots \otimes \hat{S}_x \otimes \dots \otimes \mathbb{1} \\ \hat{S}_y^{(k)} &= \mathbb{1} \otimes \mathbb{1} \otimes \dots \otimes \hat{S}_y \otimes \dots \otimes \mathbb{1} \\ \hat{S}_z^{(k)} &= \mathbb{1} \otimes \mathbb{1} \otimes \dots \otimes \hat{S}_z \otimes \dots \otimes \mathbb{1} \end{aligned} \quad (2.73)$$

where $\hat{S}_{\{x, y, z\}}^{(k)}$ is the operator representing an x -, y -, or z - projection of spin on the k^{th} particle in the system of particles.

The Hamiltonians of two non-interacting uncorrelated subsystems must not allow the subsystems to affect each other, giving a composite Hamiltonian of two subsystems as

$$\hat{\mathcal{H}}_{1,2} = \hat{\mathcal{H}}_1 \otimes \mathbb{1}_2 + \mathbb{1}_1 \otimes \hat{\mathcal{H}}_2 \quad (2.74)$$

where $\mathbb{1}$ is the identity operator.

In extending this description of spherical tensor operators to coupled particles, *Clebsch-Gordan coefficients* can be used to couple two tensors of a given rank to another rank [7]. The *Clebsch-Gordan coefficients* can be interpreted as a unitary transformation from a linear combination of products of single particle operators to two particle operators [73]. Two particle tensor operators, $\hat{T}_{\ell m}^{(12)}$, from the single particle operators $\hat{T}_{\ell_1 m_1}^{(1)}$ and $\hat{T}_{\ell_2 m_2}^{(2)}$ defined in this way are

$$\hat{T}_{\ell m}^{(12)} = \sum_{m_1} \mathcal{C}_{m_1 m_2 m}^{\ell_1 \ell_2 \ell} \hat{T}_{\ell_1 m_1}^{(1)} \hat{T}_{\ell_2 m_2}^{(2)} \quad (2.75)$$

where $C_{m_1 m_2 m}^{\ell_1 \ell_2 \ell}$ are *Clebsch-Gordan coefficients*. The two particle spherical tensor operators of rank-2 are

$$\hat{T}_{2\pm 2}^{(2)} = \frac{1}{2} \hat{L}_{\pm}^2 \quad (2.76)$$

$$\hat{T}_{2\pm 1}^{(2)} = \mp \frac{1}{2} (\hat{L}_z \hat{L}_{\pm} + \hat{L}_{\pm} \hat{L}_z) \quad (2.77)$$

$$\hat{T}_{20}^{(2)} = \sqrt{\frac{2}{3}} (\hat{L}_z^2 - \frac{1}{4} (\hat{L}_+ \hat{L}_- + \hat{L}_- \hat{L}_+)) \quad (2.78)$$

The full list of irreducible spherical tensor operators is given in §A FIG. A.3.

§ 2.4 The Spin Hamiltonian

Spin Hamiltonians are subject to three types of interactions, and can only have three generic algebraic forms [73]:

Linear in the spin operators

→ Interactions with the static magnetic field, B_0 , and the irradiated radio/microwave frequency $B_1(t)$ field.

Bilinear in the spin operators

→ Interactions between two spins.

Quadratic in the spin operators

→ Indirect interactions e.g. with field gradient, interpreted as the interaction of a spin and itself.

In generating the total Hamiltonian of a arbitrary spin system, it is useful to partition the system into general parts corresponding to different physical processes

$$\hat{\mathcal{H}} = \hat{\mathcal{H}}_Z + \hat{\mathcal{H}}_{NN} + \hat{\mathcal{H}}_{NE} + \hat{\mathcal{H}}_{EE} + \hat{\mathcal{H}}_k \quad (2.79)$$

Each of the split Hamiltonian parts represents the following interactions

$\hat{\mathcal{H}}_Z$ = Zeeman interactions

→ Chemical shielding tensors and g-tensors.

$\hat{\mathcal{H}}_{NN}$ = Inter-Nuclear interactions

→ J-couplings, quadrupolar interactions, dipolar interactions.

$\hat{\mathcal{H}}_{\mathcal{N}\mathcal{E}}$ = **Electron-Nuclear interactions**

→ Isotropic and anisotropic hyperfine couplings.

$\hat{\mathcal{H}}_{\mathcal{E}\mathcal{E}}$ = **Inter-Electron interactions**

→ Exchange interaction, zero field splitting, dipolar interactions.

$\hat{\mathcal{H}}_k$ = **Microwave/Radiofrequency irradiation**

→ Irradiated pulses with amplitude and phase, or control pulses with associated amplitude coefficients.

In describing these interactions it is useful to introduce the notation for a vector of spin operators, $\hat{\vec{S}}$, which contains the three Cartesian operators as elements, ordered $(\hat{S}_x, \hat{S}_y, \hat{S}_z)$.

§ 2.4.1 Zeeman Interactions

An applied external magnetic field, \vec{B} induces an energy difference ΔE between the quantised magnetic moment values, with the energy is given by

$$E = -\vec{\mu} \cdot \vec{B} \quad (2.80)$$

where \vec{B} is the external magnetic field. For nucleons, this is the nuclear Zeeman interaction, and for electrons, this is called the electron Zeeman interaction. Its Hamiltonian is

$$\hat{\mathcal{H}}_Z = \hat{\vec{S}} \cdot \mathbf{A} \cdot \vec{B} = (\hat{S}_x \quad \hat{S}_y \quad \hat{S}_z) \begin{pmatrix} a_{xx} & a_{xy} & a_{xz} \\ a_{yx} & a_{yy} & a_{yz} \\ a_{zx} & a_{zy} & a_{zz} \end{pmatrix} \begin{pmatrix} B_x \\ B_y \\ B_z \end{pmatrix} \quad (2.81)$$

where the tensor \mathbf{A} is the Zeeman interaction tensor. This is the coupling of the spin dipole moment to the external magnetic field. Without any further interaction, the Zeeman tensor reduces to the constant $\frac{g\mu}{\hbar}$, where g is the g -factor and μ is the nuclear magneton for nuclei and the Bohr magneton for electrons.

Chemical shielding tensor

The magnetic field also perturbs the electronic structure of the nucleus, with the induced field $\delta\vec{B}_0$. Using perturbation theory [209] gives the spin interaction from

the contribution of $\delta\vec{B}_0$ as

$$\hat{\mathcal{H}}_{CS} = \hat{\vec{S}} \cdot \boldsymbol{\delta} \cdot \vec{B} = \begin{pmatrix} \hat{S}_x & \hat{S}_y & \hat{S}_z \end{pmatrix} \begin{pmatrix} \delta_{xx} & \delta_{xy} & \delta_{xz} \\ \delta_{yx} & \delta_{yy} & \delta_{yz} \\ \delta_{zx} & \delta_{zy} & \delta_{zz} \end{pmatrix} \begin{pmatrix} B_x \\ B_y \\ B_z \end{pmatrix} \quad (2.82)$$

where $\boldsymbol{\delta}$ is known as the chemical shielding tensor.

Electron g-tensor

A similar interaction to the perturbation from chemical shielding can be derived for electrons. The energy operator for the electron dipole in an external magnetic field is

$$\hat{\mathcal{H}}_G = \mu_B \hat{\vec{L}} \cdot \boldsymbol{g} \cdot \vec{B} = \begin{pmatrix} \hat{L}_x & \hat{L}_y & \hat{L}_z \end{pmatrix} \begin{pmatrix} g_{xx} & g_{xy} & g_{xz} \\ g_{yx} & g_{yy} & g_{yz} \\ g_{zx} & g_{zy} & g_{zz} \end{pmatrix} \begin{pmatrix} B_x \\ B_y \\ B_z \end{pmatrix} \quad (2.83)$$

where \boldsymbol{g} is known as the g -tensor and the constant μ_B is the Bohr magneton.

§ 2.4.2 Inter-Nuclear Interactions

Inter-nuclear dipolar interactions

Treating nuclear spins as point magnetic dipoles, $\vec{\mu}$, the energy of the interaction between two nuclear dipoles, $\vec{\mu}_1$ and $\vec{\mu}_2$ is

$$E = -\vec{\mu} \cdot \vec{B} = -\frac{\mu_0}{4\pi} \frac{3(\vec{\mu}_1 \cdot \vec{r})(\vec{r} \cdot \vec{\mu}_2) - r^3(\vec{\mu}_1 \cdot \vec{\mu}_2)}{r^5} \quad (2.84)$$

where \vec{r} is the vector that connects the dipoles, and the constant μ_0 is the permeability of free space. Hamiltonian is for this interaction between two spins, $\hat{\vec{S}}^{(1)}$ and $\hat{\vec{S}}^{(2)}$

$$\hat{\mathcal{H}}_{DD} = -\frac{\mu_0 \gamma_1 \gamma_2 \hbar}{4\pi r^5} \left(3(\hat{\vec{S}}^{(1)} \cdot \vec{r})(\vec{r} \cdot \hat{\vec{S}}^{(2)}) - r^2(\hat{\vec{S}}^{(1)} \cdot \hat{\vec{S}}^{(2)}) \right) \quad (2.85)$$

where γ_1 and γ_2 are the magnetogyric ratios of the two spins.

Spin-rotation coupling

Collections of particles have angular momentum, just as single particles do. A magnetic induction is generated by a collection charges moving at velocity. Using EQ. (2.81) and considering the motion of the n^{th} charge with mass m_n and angular momentum operator $\hat{\vec{L}}_n$ – the spin rotation coupling interaction is

$$\hat{\mathcal{H}}_{SR} = \sum_k \gamma_k \vec{B} \cdot \hat{\vec{S}}_k = -\frac{\mu_0}{4\pi} \sum_{n,k} \frac{\gamma_k q_n}{m_n r_n^3} \left(\hat{\vec{L}}_n \cdot \hat{\vec{S}}_k \right) \quad (2.86)$$

where γ_k is the magnetogyric ratio of the k^{th} nucleus and $\hat{\vec{S}}_k$ are their spin operators. Evaluating the sum over the n moving charges, the spin-rotation Hamiltonian is

$$\hat{\mathcal{H}}_{SR} = - \sum_k \hat{\vec{L}} \cdot \mathbf{A}_k \cdot \hat{\vec{S}}_k \quad (2.87)$$

where \mathbf{A}_k are spin-rotation coupling tensors.

Quadrupolar coupling

Nuclei with spin $> \text{spin} - 1/2$ have non-spherical charge distribution, and the direction of this *nuclear quadrupole moment* is fixed, it interacts with the spin itself.

$$\hat{\mathcal{H}}_Q = \frac{-eQ}{2s(2s-1)} \hat{\vec{L}} \cdot \mathbf{V} \cdot \hat{\vec{L}} = \frac{-eQ}{2s(2s-1)} \begin{pmatrix} \hat{L}_x & \hat{L}_y & \hat{L}_z \end{pmatrix} \begin{pmatrix} v_{xx} & v_{xy} & v_{xz} \\ v_{yx} & v_{yy} & v_{yz} \\ v_{zx} & v_{zy} & v_{zz} \end{pmatrix} \begin{pmatrix} \hat{L}_x \\ \hat{L}_y \\ \hat{L}_z \end{pmatrix} \quad (2.88)$$

where eQ is the quadrupole moment of the nucleus, s is the spin of the nucleus, $-\frac{eQ}{2s(2s-1)} \mathbf{V}$ is the quadrupolar coupling tensor. Quadrupolar coupling only exists for spin – 1 nuclei and higher.

J-coupling

J-coupling is the scalar spin-spin coupling between two spins, after ignoring anisotropy, its Hamiltonian is

$$\begin{aligned}\hat{\mathcal{H}}_{\mathcal{J}} &= (\hat{L}_x \quad \hat{L}_y \quad \hat{L}_z) \begin{pmatrix} a & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & a \end{pmatrix} \begin{pmatrix} \hat{S}_x \\ \hat{S}_y \\ \hat{S}_z \end{pmatrix} \\ &= a(\hat{L}_x \hat{S}_x + \hat{L}_y \hat{S}_y + \hat{L}_z \hat{S}_z) = a(\hat{\vec{L}} \cdot \hat{\vec{S}})\end{aligned}\quad (2.89)$$

where a is the scalar coupling strength.

§ 2.4.3 Microwave & Radiofrequency terms

These interactions are the most important to the area of optimal control – they are the part of the Hamiltonian that can be directly controlled – all else is considered uncontrollable and is summed to a single *drift Hamiltonian*. Commonly referred to as the time dependent $B_1(t)$ field, areas of magnetic resonance manipulating nuclear spins use radiofrequency fields, and areas of magnetic resonance manipulating electrons use microwave fields.

Optimal control, set out in §3, uses commonly control operators $\hat{\mathcal{H}}_x$ and $\hat{\mathcal{H}}_y$ with their amplitude coefficients becoming the variables of an optimisation algorithm. These amplitude coefficients relate directly to the pulse power in Hz, obtained from pulse calibration on the instrument.

This interaction has the same form as the Zeeman interaction in EQ. (2.81)

$$\hat{\mathcal{H}} = -\hat{\vec{L}}_k \cdot \gamma_k \cdot \vec{B}_{rf}(t) \quad (2.90)$$

It is normal to polarise the radiofrequency field, of frequency ω , with a phase φ [73]:

$$\vec{B}_{rf} = 2B_1 \cos(\omega t(\vec{e}_x \cos \varphi + \vec{e}_y \sin \varphi)) \quad (2.91)$$

where \vec{e}_x and \vec{e}_y are the orthogonal unit vectors in the x and y directions. The above equation can be decomposed into two counter-rotating components (one of

which can be neglected with a high B_0 field), giving the Hamiltonian as [73]^{1}

$$\hat{\mathcal{H}}_k = -B_1 \sum_{k=1}^{\frac{1}{2}K} \gamma_k \left(\hat{L}_x^{(k)} \cos(\omega t + \varphi) + \hat{L}_y^{(k)} \sin(\omega t + \varphi) \right) \quad (2.92)$$

and in the time-independent rotating frame, applying a rotation transform, gives:

$$\hat{\mathcal{H}}_x^{(k)} = -B_1 \gamma_k \hat{L}_x^{(k)} \cos \varphi \quad (2.93)$$

$$\hat{\mathcal{H}}_y^{(k)} = -B_1 \gamma_k \hat{L}_y^{(k)} \sin \varphi \quad (2.94)$$

^{1}The factor in the sum of $\frac{1}{2}K$ will become apparent when discussing optimal control in detail, but it is common to sum to the total number of control operators, usually $2 \times$ the number of spins – it is kept here for consistency.

§ 3

Optimal Control Theory

“You know what charm is: a way of getting the answer yes without having asked any clear question.”

– Albert Camus, THE FALL

The focus of this thesis is advancing optimal control methods, specifically the GRAPE method [154]. Before this thesis progresses to present a novel optimal control method in §4 and advances in efficiency of calculation within this method in §5, the current chapter will present the theory of optimal control in the context of the GRAPE method.

The state of a magnetic resonance system can be controlled using a set of electromagnetic pulses, discrete or continuous in time. Design of these pulses may prove difficult for control of complicated systems; numerical optimisation methods can be used to find a maximum overlap between the desired state and the state produced by the set of pulses, with the pulse schedule being the parameter of the objective function in the optimisation problem.

The art of running an experiment to a given accuracy with minimal expenditure of time and resources is known as *optimal control theory*. A simple interpretation of optimal control theory is the application of numerical optimisation [17, 23, 79, 96, 229] to control theory [45, 92, 191, 246, 298]. This chapter will be divided into two sections, firstly outlining gradient based numerical optimisation with Newton-type methods, and secondly building a derivation of the optimal control method GRAPE.

§ 3.1 Numerical optimisation

Continuing from §1 and the example of Newton's root finding method adapted to find the roots of an analytical gradient in FIG. 1.1, this section will formally extend that idea to build a family of generalised optimisation algorithms called Newton-type methods. An optimisation problem can be written as

$$\min_{\mathbf{c}} \left\{ J(\mathbf{c}) \right\}, \quad \mathbf{c} \in \mathbb{R}^n \quad (3.1)$$

finding a local minimum, and a transformation of the above problem can find a local maximum:

$$-\min_{\mathbf{c}} \left\{ -J(\mathbf{c}) \right\} = \max_{\mathbf{c}} \left\{ J(\mathbf{c}) \right\}, \quad \mathbf{c} \in \mathbb{R}^n \quad (3.2)$$

The function $J(\mathbf{c})$ is a scalar function of n variables called the *objective function*. The maximisation problem of EQ. (3.2) was recast in the form of a minimisation problem with a simple transform, and it seems sensible to consider only minimisation problems for the remainder of this section.

In this work an objective function should be a smooth function – one which is continuously differentiable – that can be represented by a convergent power series of terms calculated from the derivatives in the vicinity of a point i.e. a local approximation. For a function of one variable, this is

$$J(\mathbf{c} + \mathbf{h}) = J(\mathbf{c}) + \frac{\mathbf{h}}{1!} \nabla J(\mathbf{c}) + \frac{\mathbf{h}^2}{2!} \nabla^2 J(\mathbf{c}) + \dots = \sum_{r=0}^{\infty} \frac{\mathbf{h}^r}{r!} \nabla^r J(\mathbf{c}) \quad (3.3)$$

where \mathbf{h} is a small displacement to the vector \mathbf{c} , and $\nabla^r J(\mathbf{c})$ is the r^{th} order derivative of the function $J(\mathbf{c})$. This series is known as the many variable *Taylor series* [201, 302].

The *gradient vector*, $\nabla J(\mathbf{c})$, and the *Hessian matrix*, $\nabla^2 J(\mathbf{c})$, are introduced here. These two definitions will be used extensively throughout the remainder of this thesis and further explanation of their notation is required: ∇ is the gradient

operator defined by

$$\nabla \triangleq |\nabla\rangle \triangleq \begin{pmatrix} \frac{\partial}{\partial c_1} \\ \frac{\partial}{\partial c_2} \\ \vdots \\ \frac{\partial}{\partial c_n} \end{pmatrix} \quad \mathbf{c} \in \mathbb{R}^n \quad (3.4)$$

which can be used to construct a Hessian operator ∇^2 :

$$\nabla^2 \triangleq |\nabla\rangle\langle\nabla| = \begin{pmatrix} \frac{\partial}{\partial c_1} \frac{\partial}{\partial c_1} & \frac{\partial}{\partial c_1} \frac{\partial}{\partial c_2} & \cdots & \frac{\partial}{\partial c_1} \frac{\partial}{\partial c_n} \\ \frac{\partial}{\partial c_2} \frac{\partial}{\partial c_1} & \frac{\partial}{\partial c_2} \frac{\partial}{\partial c_2} & \cdots & \frac{\partial}{\partial c_2} \frac{\partial}{\partial c_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial}{\partial c_n} \frac{\partial}{\partial c_1} & \frac{\partial}{\partial c_n} \frac{\partial}{\partial c_2} & \cdots & \frac{\partial}{\partial c_n} \frac{\partial}{\partial c_n} \end{pmatrix} = \begin{pmatrix} \frac{\partial^2}{\partial c_1^2} & \frac{\partial^2}{\partial c_1 \partial c_2} & \cdots & \frac{\partial^2}{\partial c_1 \partial c_n} \\ \frac{\partial^2}{\partial c_2 \partial c_1} & \frac{\partial^2}{\partial c_2^2} & \cdots & \frac{\partial^2}{\partial c_2 \partial c_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2}{\partial c_n \partial c_1} & \frac{\partial^2}{\partial c_n \partial c_2} & \cdots & \frac{\partial^2}{\partial c_n^2} \end{pmatrix} \quad (3.5)$$

A point in a vector space is denoted by a bold symbol, representing the coordinates of that point e.g. \mathbf{c}_j , where the superscript in parentheses denotes the specific iteration, j , of the optimisation. A direction is a line in the vector space \mathbb{R}^n , denoted with Dirac bra-ket symbols e.g. $|\mathbf{c}_j\rangle$. A point that minimises or maximises a function will be denoted with an asterisk, \mathbf{c}_* , and is called a *minimiser* or *maximiser* respectively^{I}.

In describing optimisation problems, it will be useful to introduce two descriptions of the objective function as a specified point: the *slope* and *curvature*, which are defined at a point, \mathbf{c}_j , as

$$\begin{aligned} \text{slope} &\triangleq \langle \nabla J(\mathbf{c}) | \mathbf{c}_j \rangle \\ \text{curvature} &\triangleq \langle \mathbf{c}_j | \nabla^2 J(\mathbf{c}) | \mathbf{c}_j \rangle \end{aligned}$$

At the minimiser of a smooth function, two observations hold^{II}:

- ① at the minimiser, the slope is zero
- ② at the minimiser, the curvature is non-negative in all directions

^{I}It is important to point out that it is only practicable to find a *local minimiser*, \mathbf{c}_* , which may not be a *global minimiser*.

^{II}for example, see FIG. 1.1 ④

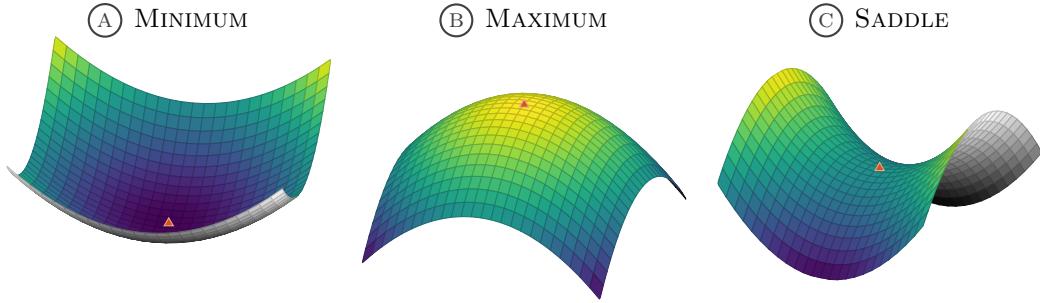


FIGURE 3.1: Types of stationary points, where the gradient is equal to zeros. (A) shows the minimiser of the function $x^2 + y^2$, (B) shows the maximiser of the function $-x^2 - y^2$, and (C) shows the saddle point of the function $x^2 - y^2$. Each plot has a single point where the gradient is equal to zero.

The two observations of a point being a minimiser lead to the *necessary conditions* for a local solution to a minimisation problem [79]. The first, that the slope is zero at a minimiser is equivalent to

$$\begin{aligned} \langle \nabla J(\mathbf{c}_*) | \mathbf{c} \rangle &= 0 \\ \implies \lim_{\mathbf{c} \rightarrow \mathbf{c}_*} \|\nabla J(\mathbf{c}_*)\|_\infty &= 0 \end{aligned} \quad (3.7)$$

and is called the *first-order necessary condition* for a minimiser, and can be interpreted as requiring the gradient to be zero at a local minimiser. FIG. 3.1 illustrates the three different types of *stationary points* satisfy the *first-order necessary condition*. Notice, only FIG. 3.1 (A) shows a minimiser, and FIG. 3.1 (C) has a point of zero gradient that is not a minimiser or a maximiser. FIG. 3.1 shows by example that EQ. (3.7) alone is not enough to identify a local minimiser.

The observation concerning curvature of the objective function at the minimiser is required to identify a minimum. This observation is equivalent to

$$\lim_{\mathbf{c} \rightarrow \mathbf{c}_*} \langle \mathbf{c} | \nabla^2 J(\mathbf{c}_*) | \mathbf{c} \rangle \geq 0, \quad \forall \mathbf{c} \quad (3.8)$$

and is called the *second-order necessary condition* for a minimiser, and can be interpreted as requiring a positive semi-definite^{1} Hessian matrix in the vicinity of a local minimiser. However, a semi-definite matrix will become a problem when an inverse of the Hessian is required. Numerically, it is useful to reformulate this

^{1}A positive semi-definite matrix has all eigenvalues $\lambda \geq 0$

last condition as

$$\lim_{\mathbf{c} \rightarrow \mathbf{c}_*} \langle \mathbf{c} | \nabla^2 J(\mathbf{c}_*) | \mathbf{c} \rangle > 0, \quad \forall \mathbf{c} \neq \mathbf{0} \quad (3.9)$$

which requires the Hessian matrix to be positive definite^{I} in the vicinity of a local minimiser. Eq. (3.7) together with Eq. (3.9) set out the *sufficient conditions* for a local minimiser, and can be checked numerically. Returning again to FIG. 3.1, it is clear that FIG. 3.1 (A) has positive curvature in all direction around the vicinity of the minimiser, FIG. 3.1 (B) has negative curvature around the maximiser, and FIG. 3.1 (C) as a mixture of positive and negative curvature around the saddle point. These sufficient conditions for an optimal solution will be used extensively in §4, and will be elaborated there.

§ 3.1.1 The gradient ascent method

In searching for an optimum, the desired minimum or maximum of an objective function, a general strategy must be developed to find new points giving better objective function values. Two popular strategies are the *restricted step methods*^{II} and the *line search method*. This chapter will be concerned with the *line search method*, and §4 will be concerned with *restricted step methods*.

An example of a strategy to perform a line search at an iteration j is

- ① find a search direction \mathbf{d}_j
- ② find a search length, ε , to minimise or maximise $J(\mathbf{c}_j + \varepsilon \mathbf{d}_j)$ with respect to ε .
- ③ set the new iteration, $\mathbf{c}_{j+1} = \mathbf{c}_j + \varepsilon \mathbf{d}_j$.

Step ② is an idealised solution called an *exact line search*, and is conceptually useful but not practicable in a finite number of operations. The subproblem of finding ε will be outlined in §3.1.4, for now it is sufficient to use the concept of an exact line search.

^{I}A positive semi-definite matrix has all eigenvalues $\lambda > 0$

^{II}also known as the *trust region methods*.

Line search methods introduce the search direction \mathbf{d}_j , at an iteration j . One choice of search direction is that of the gradient. For minimisation, the search direction should be antiparallel to the gradient, and for maximisation the search direction should be parallel to the gradient:

$$\text{minimisation,} \quad \mathbf{d}_j = -\nabla J(\mathbf{c}_j) \quad (3.10a)$$

$$\text{maximisation,} \quad \mathbf{d}_j = \nabla J(\mathbf{c}_j) \quad (3.10b)$$

Together with an adequate line search method, these two methods are called the *gradient descent method* for minimisation and the *gradient ascent method* for maximisation. The update to the objective variable using a scaled step length found from a line search is

$$\text{minimisation :} \quad \mathbf{c}_{j+1} = \mathbf{c}_j - \varepsilon \nabla J(\mathbf{c}_j) \quad (3.11a)$$

$$\text{maximisation :} \quad \mathbf{c}_{j+1} = \mathbf{c}_j + \varepsilon \nabla J(\mathbf{c}_j) \quad (3.11b)$$

These methods can also be used when no derivative information is available, although less effective, by using an estimate of the derivatives:

$$\nabla J(\mathbf{c}) = \frac{J(\mathbf{c}(1+h)) - J(\mathbf{c})}{h} + \mathcal{O}(h) \quad (3.12a)$$

$$\nabla J(\mathbf{c}) = \frac{J(\mathbf{c}(1+h)) - J(\mathbf{c}(1-h))}{2h} + \mathcal{O}(h^2) \quad (3.12b)$$

where $h = \sqrt{\text{EPS}}$ is a small perturbation to \mathbf{c} . These two approximations of the gradient are called finite difference approximations, Eq. (3.12a) is the forward differences approximation, and Eq. (3.12b) is the central differences approximation. In a similar way, the second derivatives can be approximated with finite differences of the first derivatives. In addition to their use when no derivative information is available, these approximations are also useful in checking derivative calculations when derivative information is available.

§ 3.1.2 Newton's Method

Considering that the linear approximation of a function, such as gradient descent/ascent methods, will fail to differentiate between an extremum and a stationary point, as in FIG. 3.1, more is required in the form of the sufficient conditions

already set out in EQ. (3.9). A useful model in numerical optimisation is the *quadratic model* of the function to be optimised. Although the optimisation methods should work on many functions, not just quadratic functions, in deriving the methods for optimisation the *quadratic model* has proved fruitful. Desirable properties of this model are [79]:

- ① Quadratic functions are smooth, with a well defined minimum.
- ② A continuously differentiable function is locally well approximated by a quadratic function near a minimum.
- ③ More accurate than a linear approximation of a function, as in gradient descent/ascent methods, and higher derivative orders will likely be small.
- ④ Can be made invariant under linear transformation of variables^{I}.

The aim of using a *quadratic model* of a function is to find its extremum in a single iteration by analytically solving the model of the function to find its single stationary point shown in FIG.3.1. Given an initial starting point of \mathbf{c}_0 , a quadratic model of the objective function at an extremum is a truncated Taylor series:

$$J(\mathbf{c}_*) \triangleq J(\mathbf{c}_0 + \boldsymbol{\delta}) \approx J(\mathbf{c}_0) + \langle \nabla J(\mathbf{c}_0) | \boldsymbol{\delta} \rangle + \frac{1}{2} \langle \boldsymbol{\delta} | \nabla^2 J(\mathbf{c}_0) | \boldsymbol{\delta} \rangle \quad (3.13)$$

This equation is solved to find the vector $\boldsymbol{\delta}$ required to step from the initial point \mathbf{c}_0 to the extremum \mathbf{c}_* . The solution is found when the derivative of EQ. (3.13), with respect to the step $\boldsymbol{\delta}$, is equal to zero^{II}:

$$\begin{aligned} \frac{d}{d\boldsymbol{\delta}} J(\mathbf{c}_0 + \boldsymbol{\delta}) &= \langle \nabla J(\mathbf{c}_0) | + \langle \boldsymbol{\delta} | \nabla^2 J(\mathbf{c}_0) = \langle \mathbf{0} | \\ &\implies \langle \nabla J(\mathbf{c}_0) | + \nabla^2 J(\mathbf{c}_0) | \boldsymbol{\delta} \rangle = | \mathbf{0} \rangle \end{aligned} \quad (3.14)$$

This gives two solutions: either $|\boldsymbol{\delta}\rangle = |\mathbf{0}\rangle$ which means the starting point is an extremum, $\mathbf{c}_0 = \mathbf{c}_*$; or the non-trivial solution

$$|\boldsymbol{\delta}\rangle = -[\nabla^2 J(\mathbf{c}_0)]^{-1} |\nabla J(\mathbf{c}_0)\rangle \quad (3.15)$$

^{I}An inductive proof of this property is given for Newton-like methods is given in [79].

^{II}EQ. (3.14) requires the Hessian matrix to be symmetric, $\nabla^2 = (\nabla^2)^\dagger$

Specifically, this model is true for the function in FIG. 3.1 (A), which has one minimum and the function must be minimised. To be able to find the maximum in FIG. 3.1 (B), a small modification must be made to EQ. (3.13). There are now two conditions, one is a minimisation condition that $J(\mathbf{c}_*) \leq J(\mathbf{c}_0)$ and the other is the maximisation condition that $J(\mathbf{c}_*) \geq J(\mathbf{c}_0)$:

$$\text{minimisation :} \quad J(\mathbf{c}_*) - J(\mathbf{c}_0) \leq 0 \quad (3.16a)$$

$$\text{maximisation :} \quad J(\mathbf{c}_*) - J(\mathbf{c}_0) \geq 0 \quad (3.16b)$$

Using the Newton step found in EQ. (3.15), these conditions lead to the search directions

$$|\mathbf{d}_{\text{NEWTON}}\rangle = -[\nabla^2 J(\mathbf{c})]^{-1} |\nabla J(\mathbf{c})\rangle \quad (3.17)$$

where the direction of ascent/descent is accounted for by a positive/negative definite Hessian matrix. If a *linear model* of the objective function is used, the Taylor series truncates after the gradient term in EQ. (3.13) and the update formulae would be described by EQS. (3.10a), (3.10b), the gradient descent/ascent methods using an exact line search. Following the modification of the exact line search, a Newton method using a line search method would modify the update rule to

$$|\mathbf{c}_{j+1}\rangle = |\mathbf{c}_j\rangle - \varepsilon [\nabla^2 J(\mathbf{c}_j)]^{-1} |\nabla J(\mathbf{c}_j)\rangle \quad (3.18)$$

where ε is calculated with an appropriate line search method in §3.1.4. The final case of using a Newton method to find (or avoid) the stationary point in FIG. 3.1 (C) is an important one, but left to §4.

§ 3.1.3 Quasi-Newton Methods

Although Newton's method has attractive quadratic convergence properties, it has the obvious disadvantage of needing a quadratic number of Hessian element calculations. Moreover, the formulae for the Hessian elements must be supplied. The disadvantages associated with a Hessian calculation can be avoided by using a popular class of *quasi-Newton methods*. These methods are similar to Newton's method, except the Hessian is approximated by a symmetric, positive definite matrix, \mathbf{H} which is updated at every iteration with information from gradient

calculations.

$$\text{minimisation : } |\mathbf{d}_{\text{QN}}\rangle = -[\mathbf{H}]^{-1}|\nabla J(\mathbf{c})\rangle \quad (3.19\text{a})$$

$$\text{maximisation : } |\mathbf{d}_{\text{QN}}\rangle = +[\mathbf{H}]^{-1}|\nabla J(\mathbf{c})\rangle \quad (3.19\text{b})$$

A procedure to update the Hessian approximation follows

- ① set the current search direction $|\mathbf{d}_j\rangle = \pm [\mathbf{H}_j]^{-1}|\nabla J(\mathbf{c}_j)\rangle$
- ② perform a line search to find a step length, ε , giving $|\mathbf{c}_{j+1}\rangle = |\mathbf{c}_j\rangle + \varepsilon|\mathbf{d}_j\rangle$
- ③ update the Hessian approximation, \mathbf{H}_j , with an appropriate method to give a new approximation \mathbf{H}_{j+1}

Maximisation and minimisation are accounted for with \pm : maximisation with positive direction; minimisation with negative direction, allowing a strictly positive definite Hessian matrix.

A useful property of this procedure is that the Hessian approximation can use any gradient information obtained in the line search. Without better information, it is normal to set the initial Hessian approximation equal to the identity matrix $\mathbf{H}_0 = \mathbb{1}$. An advantage of *quasi-Newton methods* over other methods, including the *conjugate gradient method* [124, 177], is that the quasi-Newton iteration does not need to be restarted periodically [79].

After repeated updating, the matrix \mathbf{H} should approximate the Hessian matrix $\nabla^2 J(\mathbf{c})$, and one method to achieve this is to use the difference of gradients from iteration to iteration. For brevity, the change in objective variable and the change in gradient will be denoted by

$$\Delta \mathbf{c} \triangleq (\mathbf{c}_{j+1} - \mathbf{c}_j) \quad (3.20\text{a})$$

$$\Delta \mathbf{g} \triangleq (\nabla J(\mathbf{c}_{j+1}) - \nabla J(\mathbf{c}_j)) \quad (3.20\text{b})$$

Using a truncated Taylor series of the gradient, with a quadratic model of the objective, an equation for the change in gradient can be derived:

$$\begin{aligned} |\nabla J(\mathbf{c}_j + \varepsilon \mathbf{d}_j)\rangle &= |\nabla J(\mathbf{c}_j)\rangle + \nabla^2 J(\mathbf{c}_j) \varepsilon \mathbf{d}_j + \dots \\ \implies |\Delta \mathbf{g}\rangle &= \nabla^2 J(\mathbf{c}_j) |\Delta \mathbf{c}\rangle \end{aligned} \quad (3.21)$$

Since the Hessian approximation is calculated after a line search as the final stage of an iteration, \mathbf{H}_{j+1} correctly relates the gradient change in EQ. (3.21), giving

$$|\Delta \mathbf{g}\rangle = \mathbf{H}_{j+1} |\Delta \mathbf{c}\rangle \quad (3.22)$$

which is known as the *secant equation* [229]. It maps the symmetric, positive-definite Hessian to the gradient, which is only possible if

$$[x_{k+1} - x_k]^T [\nabla f_{k+1} - \nabla f_k] > 0 \quad (3.23)$$

This is known as the *curvature condition* and is satisfied for convex function [229].

Alternatively, the matrix approximating the inverse to the Hessian can be updated, reducing the computational load^{11}.

$$\text{minimisation :} \quad \mathbf{d}_{\text{QN}} = -[\mathbf{I}] |\nabla J(\mathbf{c})\rangle \quad (3.24a)$$

$$\text{maximisation :} \quad \mathbf{d}_{\text{QN}} = +[\mathbf{I}] |\nabla J(\mathbf{c})\rangle \quad (3.24b)$$

A similar procedure as above updates the approximation to the inverse-Hessian, \mathbf{I}_j , except the search direction does not involve an inverse of a matrix: $|\mathbf{d}_j\rangle = \pm [\mathbf{H}_j]^{-1} |\nabla J(\mathbf{c}_j)\rangle$. Similarly, the change in gradient is used to approximate the inverse-Hessian with

$$|\Delta \mathbf{c}\rangle = [\mathbf{I}_{j+1}] |\Delta \mathbf{g}\rangle \quad (3.25)$$

which is known as the *quasi-Newton condition* [79].

Rank-1 Hessian update

Initial attempts to find a method of updating the Hessian approximation that ensures its symmetry and satisfies EQ. (3.22), resulted in a method called the *symmetric rank-1* or SR1 method [30, 47]. A rank-1 symmetric matrix can be constructed from the outer product of a real vector with itself, $|\mathbf{u}\rangle\langle\mathbf{u}|$, giving the symmetric rank-1 update to the Hessian approximation

$$\mathbf{H}_{j+1}^{\text{SR1}} = \mathbf{H}_j^{\text{SR1}} + a |\mathbf{u}\rangle\langle\mathbf{u}| \quad (3.26)$$

^{11}The inverse of a matrix is regarded as an expensive matrix operation that can be ill-defined [106]

where a is an arbitrary constant and $|\mathbf{u}\rangle$ is a vector to be determined. Substituting the last equation into the secant equation, Eq. (3.22), gives

$$|\Delta\mathbf{g}\rangle = \mathbf{H}_j^{\text{SR1}}|\Delta\mathbf{c}\rangle + a\left[\langle\mathbf{u}|\Delta\mathbf{c}\rangle\right]|\mathbf{u}\rangle \quad (3.27)$$

The vector $|\mathbf{u}\rangle$ must be proportional to $|\Delta\mathbf{g}\rangle - \mathbf{H}_j^{\text{SR1}}|\Delta\mathbf{c}\rangle$, with proportionality absorbed into the arbitrary constant a , giving

$$|\Delta\mathbf{g}\rangle - \mathbf{H}_j^{\text{SR1}}|\Delta\mathbf{c}\rangle = a\left[\langle\Delta\mathbf{c}|\left(|\Delta\mathbf{g}\rangle - \mathbf{H}_j^{\text{SR1}}|\Delta\mathbf{c}\rangle\right)\right]\left(|\Delta\mathbf{g}\rangle - \mathbf{H}_j^{\text{SR1}}|\Delta\mathbf{c}\rangle\right) \quad (3.28)$$

This is satisfied for a if [229]

$$a = \frac{\text{sign}\left[\langle\Delta\mathbf{c}|\left(|\Delta\mathbf{g}\rangle - \mathbf{H}_j^{\text{SR1}}|\Delta\mathbf{c}\rangle\right)\right]}{\left[\langle\Delta\mathbf{c}|\left(|\Delta\mathbf{g}\rangle - \mathbf{H}_j^{\text{SR1}}|\Delta\mathbf{c}\rangle\right)\right]} \quad (3.29)$$

and gives the only symmetric rank-1 update, satisfying Eq. (3.22), to be

$$\mathbf{H}_{j+1}^{\text{SR1}} = \mathbf{H}_j^{\text{SR1}} + \frac{|\Delta\mathbf{g} - \mathbf{H}_j^{\text{SR1}}\Delta\mathbf{c}\rangle\langle\Delta\mathbf{g} - \mathbf{H}_j^{\text{SR1}}\Delta\mathbf{c}|}{\langle\Delta\mathbf{g}|\Delta\mathbf{c}\rangle - \langle\Delta\mathbf{c}|\mathbf{H}_j^{\text{SR1}}|\Delta\mathbf{c}\rangle} \quad (3.30)$$

A similar derivation can be made to find a symmetric rank-1 update that satisfies Eq. (3.25), giving an update rule for the inverse-Hessian approximation. Alternatively, a useful formula to use Eq. (3.30) to find a corresponding update rule for the inverse-Hessian approximation is the *Sherman-Morrison-Woodbury formula* [106]^{I}. Using the formula for a rank-1 update amounts to the interchanges $\mathbf{H} \longleftrightarrow \mathbf{I}$ and $\Delta\mathbf{g} \longleftrightarrow \Delta\mathbf{c}$, giving^{II}

$$\mathbf{I}_{j+1}^{\text{SR1}} = \mathbf{I}_j^{\text{SR1}} + \frac{|\Delta\mathbf{c} - \mathbf{I}_j^{\text{SR1}}\Delta\mathbf{g}\rangle\langle\Delta\mathbf{c} - \mathbf{I}_j^{\text{SR1}}\Delta\mathbf{g}|}{\langle\Delta\mathbf{c}|\Delta\mathbf{g}\rangle - \langle\Delta\mathbf{g}|\mathbf{I}_j^{\text{SR1}}|\Delta\mathbf{g}\rangle} \quad (3.31)$$

Eqs. (3.30), (3.31) both ensure any update is a symmetric matrix, however, the update is not guaranteed to be positive definite [79, 229]. This drawback may

^{I} Sherman-Morrison-Woodbury formula: [106]

$$(\mathbf{A} + \mathbf{U}\mathbf{V}^T)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}(\mathbf{I} + \mathbf{V}^T\mathbf{A}^{-1}\mathbf{U})^{-1}\mathbf{V}^T\mathbf{A}^{-1}$$

with $\mathbf{A} \in \mathbb{R}^{n \times n}$ and both $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{n \times k}$ for $1 \leq k \leq n$. This shows a rank- k update to a matrix gives a rank- k update to its inverse.

^{II} This is a property of a *self-dual formula*.

be alleviated by forcing a positive definite matrix with a method described in §4. However, there is a second problem with Eqs. (3.30), (3.31), that the denominator may vanish.

Rank-2 Hessian update

In an attempt to avoid the problems of using Eqs. (3.30), (3.31), a more flexible Hessian update formula can be constructed by replacing the rank-1 update of Eq. (3.26) with a rank-2 update

$$\mathbf{H}_{j+1}^{\text{DFP}} = \mathbf{H}_j^{\text{DFP}} + a|\mathbf{u}\rangle\langle\mathbf{u}| + b|\mathbf{v}\rangle\langle\mathbf{v}| \quad (3.32)$$

Using the secant equation, Eq. (3.22), and substituting into this rank-2 update gives

$$|\Delta\mathbf{g}\rangle = \mathbf{H}_j^{\text{DFP}}|\Delta\mathbf{c}\rangle + a\left[\langle\mathbf{u}|\Delta\mathbf{c}\rangle\right]|\mathbf{u}\rangle + b\left[\langle\mathbf{v}|\Delta\mathbf{c}\rangle\right]|\mathbf{v}\rangle \quad (3.33)$$

There was only one symmetric rank-1 solution to Eq. (3.22), there are a family of symmetric rank-2 update formulae which will be summarised at the end of this section in a single equation of one variable, Eqs. (3.39a), (3.39b), for a Hessian approximation and an inverse-Hessian approximation.

To start a derivation of the general rank-2 formulae Eqs. (3.39a), (3.39b) a set of non-unique solutions to Eq. (3.33) are trialled, while ensuring the solutions satisfy the curvature condition of Eq. (3.23): $|\mathbf{u}\rangle = \mathbf{H}_j^{\text{DFP}}|\Delta\mathbf{c}\rangle$ and $|\mathbf{v}\rangle = |\Delta\mathbf{g}\rangle$, giving the arbitrary constants [79]

$$a = \frac{1}{\langle\Delta\mathbf{c}|\mathbf{H}_j^{\text{DFP}}|\Delta\mathbf{c}\rangle} \quad b = \frac{-1}{\langle\Delta\mathbf{g}|\Delta\mathbf{c}\rangle} \quad (3.34)$$

This gives one of two special cases to a general formula to compute the approximation to the Hessian matrix is [229], which in itself can be used as a rank-2 update approximating a Hessian:

$$\begin{aligned} \mathbf{H}_{j+1}^{\text{DFP}} = \mathbf{H}_j^{\text{DFP}} &+ \left[\mathbb{1} + \frac{\langle\Delta\mathbf{c}|\mathbf{H}_j^{\text{DFP}}|\Delta\mathbf{c}\rangle}{\langle\Delta\mathbf{g}|\Delta\mathbf{c}\rangle} \right] \frac{|\Delta\mathbf{g}\rangle\langle\Delta\mathbf{g}|}{\langle\Delta\mathbf{g}|\Delta\mathbf{c}\rangle} \\ &- \left[\frac{|\Delta\mathbf{g}\rangle\langle\Delta\mathbf{c}|\mathbf{H}_j^{\text{DFP}} + \mathbf{H}_j^{\text{DFP}}|\Delta\mathbf{c}\rangle\langle\Delta\mathbf{g}|}{\langle\Delta\mathbf{g}|\Delta\mathbf{c}\rangle} \right] \end{aligned} \quad (3.35)$$

The formula is called the DFP updating formula^{I} and was originally published in a technical paper [46, 48] and later presented using Eq. (3.32) by Fletcher and Powell [80]. Again, using the *Sherman-Morrison-Woodbury formula* [106], the update in Eq. (3.35) can be transformed to update an inverse-Hessian approximation:

$$\mathbf{I}_{j+1}^{\text{DFP}} = \mathbf{I}_j^{\text{DFP}} - \left[\frac{\mathbf{I}_j^{\text{DFP}} |\Delta \mathbf{g} \rangle \langle \Delta \mathbf{g}| \mathbf{I}_j^{\text{DFP}}}{\langle \Delta \mathbf{g} | \mathbf{I}_j^{\text{DFP}} | \Delta \mathbf{g} \rangle} \right] + \frac{|\Delta \mathbf{c} \rangle \langle \Delta \mathbf{c}|}{\langle \Delta \mathbf{g} | \Delta \mathbf{c} \rangle} \quad (3.36)$$

In comparison to the SR1 updates in Eqs. (3.30), (3.31), the DFP updates in Eqs. (3.35), (3.36) ensure a positive definite update, which can be proved with a Cholesky factorisation^{II} of $\mathbf{I}_j^{\text{DFP}}$. There still exists a drawback with these update methods, that they need an accurate line search [79], which can become computationally expensive when gradient calculations are needed for a line search (see §3.1.4).

Although the SR1 formulae in Eqs. (3.30), (3.31) are *self-dual*, it is obvious that the DFP formulae in Eqs. (3.35), (3.36) are not. An important formula, called the BFGS formula [31, 32, 78, 100, 280]^{III}, can be constructed by observing $\mathbf{H}_{j+1}^{\text{BFGS}} \mathbf{I}_{j+1}^{\text{BFGS}} = \mathbf{1}$ and finding the *dual formula* of Eq. (3.36) by interchanging $\mathbf{H} \longleftrightarrow \mathbf{I}$ and $\Delta \mathbf{g} \longleftrightarrow \Delta \mathbf{c}$:

$$\mathbf{H}_{j+1}^{\text{BFGS}} = \mathbf{H}_j^{\text{BFGS}} - \left[\frac{\mathbf{H}_j^{\text{BFGS}} |\Delta \mathbf{c} \rangle \langle \Delta \mathbf{c}| \mathbf{H}_j^{\text{BFGS}}}{\langle \Delta \mathbf{c} | \mathbf{H}_j^{\text{BFGS}} | \Delta \mathbf{c} \rangle} \right] + \frac{|\Delta \mathbf{g} \rangle \langle \Delta \mathbf{g}|}{\langle \Delta \mathbf{c} | \Delta \mathbf{g} \rangle} \quad (3.37)$$

A BFGS update for an approximation to the inverse-Hessian can be found by transforming this last formula with the *Sherman-Morrison-Woodbury formula* [106]:

$$\begin{aligned} \mathbf{I}_{j+1}^{\text{BFGS}} = \mathbf{I}_j^{\text{BFGS}} + & \left[\mathbf{1} + \frac{\langle \Delta \mathbf{g} | \mathbf{I}_j^{\text{BFGS}} | \Delta \mathbf{g} \rangle}{\langle \Delta \mathbf{c} | \Delta \mathbf{g} \rangle} \right] \frac{|\Delta \mathbf{c} \rangle \langle \Delta \mathbf{c}|}{\langle \Delta \mathbf{c} | \Delta \mathbf{g} \rangle} \\ & - \left[\frac{|\Delta \mathbf{c} \rangle \langle \Delta \mathbf{g}| \mathbf{I}_j^{\text{BFGS}} + \mathbf{I}_j^{\text{BFGS}} |\Delta \mathbf{g} \rangle \langle \Delta \mathbf{c}|}{\langle \Delta \mathbf{c} | \Delta \mathbf{g} \rangle} \right] \end{aligned} \quad (3.38)$$

The BFGS update formulae are widely accepted to perform better than the DFP formulae [79, 96, 229], requiring a line search with comparatively fewer inner iterations.

^{I}The method is commonly named after the authors Davidon, Fletcher, and Powell. It is also known as the *variable metric algorithm*.

^{II}if a matrix \mathbf{A} is positive definite, there exists a factorisation $\mathbf{A} = \mathbf{L}\mathbf{L}^\dagger$ where \mathbf{L} is a lower triangular matrix with real, positive diagonal elements. This is called a Cholesky factorisation [106]

^{III}named after the authors Broyden, Fletcher, Goldfarb, and Shanno

The DFP and BFGS are both special cases for the general *Broyden family* rank-2 formulae [30]:

$$\mathbf{H}_{j+1}^\phi = (1 - \phi)\mathbf{H}_{j+1}^{\text{DFP}} + \phi\mathbf{H}_{j+1}^{\text{BFGS}} \quad (3.39\text{a})$$

$$\mathbf{I}_{j+1}^\phi = (1 - \phi)\mathbf{I}_{j+1}^{\text{DFP}} + \phi\mathbf{I}_{j+1}^{\text{BFGS}} \quad (3.39\text{b})$$

with $\phi \in [0, 1]$.

In considering the benefit of calculating the inverse-Hessian approximation of Eqs. (3.36), (3.38), that there is no need to find the inverse of a matrix explicitly, there is a drawback – as will be pointed out in §4, the inverse-Hessian approximations do not allow the implementation of *trust region methods* and other conditioning methods.

Limited memory quasi-Newton methods

Large optimisation problems may require Hessian approximations which are not sparse and cannot be computed at a reasonable cost. Rank-1 and rank-2 updates only add further vector information to the Hessian approximation, and the end use of the Hessian information is to calculate a vector search direction. *Limited memory quasi-Newton methods* [228] make use of a store of the m most recent vectors used to describe the Hessian approximation – saving storage space if $m < n$, where n is the number of objective function variables. The following review of this class of methods is from [229]. As the ℓ -BFGS (limited-memory BFGS) method [34, 194] is a popular limited-memory quasi-Newton method in optimal control [86] (and more compactly derived), the following derivation will follow this update to the inverse Hessian approximation. Before proceeding, it will be useful to present Eq. (3.38) in a product form [228]:

$$\mathbf{I}_{j+1}^{\text{BFGS}} = \mathbf{V}_j^\dagger \mathbf{I}_j^{\text{BFGS}} \mathbf{V}_j + \frac{|\Delta \mathbf{c}_j \rangle \langle \Delta \mathbf{c}_j|}{\langle \Delta \mathbf{g}_j | \Delta \mathbf{c}_j \rangle}, \quad \mathbf{V}_j = \left[\mathbf{1} - \frac{|\Delta \mathbf{g}_j \rangle \langle \Delta \mathbf{c}_j|}{\langle \Delta \mathbf{g}_j | \Delta \mathbf{c}_j \rangle} \right] \quad (3.40)$$

Considering that quasi-Newton methods are updates to the Hessian, or inverse-Hessian approximation from vectors describing the change in objective function, $\Delta \mathbf{c}$, and the change in its gradient $\Delta \mathbf{g}$ – a Hessian approximation at an iteration j can be recursively calculated from a store of these vectors and the initial Hessian

approximation at $j = j - m$ i.e. m iterations before the current iteration j .

$$\begin{aligned} \mathbf{I}_j^{\text{BFGS}} = & \left[\prod_{k=j-1}^{j-m} \mathbf{V}_k^\dagger \right] \mathbf{I}_{j-m}^{\text{BFGS}} \left[\prod_{k=j-m}^{j-1} \mathbf{V}_k \right] \\ & + \frac{1}{\langle \Delta \mathbf{g}_{j-m} | \Delta \mathbf{c}_{j-m} \rangle} \left[\prod_{k=j-1}^{j-m+1} \mathbf{V}_k^\dagger \right] \langle \Delta \mathbf{c}_{j-m} | \Delta \mathbf{c}_{j-m} \rangle \left[\prod_{k=j-m+1}^{j-1} \mathbf{V}_k \right] \\ & + \frac{1}{\langle \Delta \mathbf{g}_{j-m+1} | \Delta \mathbf{c}_{j-m+1} \rangle} \left[\prod_{k=j-1}^{j-m+2} \mathbf{V}_k^\dagger \right] \langle \Delta \mathbf{c}_{j-m+1} | \Delta \mathbf{c}_{j-m+1} \rangle \left[\prod_{k=j-m+2}^{j-1} \mathbf{V}_k \right] \\ & + \cdots + \frac{1}{\langle \Delta \mathbf{g}_{j-1} | \Delta \mathbf{c}_{j-1} \rangle} \langle \Delta \mathbf{c}_{j-1} | \Delta \mathbf{c}_{j-1} \rangle \quad (3.41) \end{aligned}$$

where the ordered products of \mathbf{V}_k and its transpose \mathbf{V}_k^\dagger form ordered products. This formula lends itself to a recursive algorithm to calculate the inverse-Hessian approximation given a store of objective and gradient vectors. However, the inverse-Hessian approximation $\mathbf{I}_{j-m}^{\text{BFGS}}$ must still be stored, defeating the point of the algorithm. The proposition of the limited-memory methods is to use a scaling matrix \mathbf{I}_0 , computed at each update [228]:

$$\mathbf{I}_j^0 = \mathbb{1} \frac{\langle \Delta \mathbf{c}_{j-1} | \Delta \mathbf{g}_{j-1} \rangle}{\langle \Delta \mathbf{g}_{j-1} | \Delta \mathbf{g}_{j-1} \rangle} \quad (3.42)$$

Further to a recursive algorithm to update the inverse-Hessian approximation, a short-cut can be taken to update the search directional of Eqs. (3.19a), (3.19b) directly. A popular ℓ -BFGS method factors in the gradient to a two-loop recursive algorithm [194, 228] to give the search direction^{11}. It should be noted that the ℓ -BFGS method is expected to perform badly when applied to ill-conditioned problems, where the Hessian proper has a wide range of eigenvalues [229] and the scaling matrix fails to be effective. This will be investigated in the context of optimal control in §4.

To derive similar recursive update algorithms for other quasi-Newton methods is not so neat because a similar product form of Eq. (3.40) has not been derived for other Broyden class methods [229]. However, an “outer product” representation of quasi-Newton methods is useful in deriving other limited-memory algorithms and shows a compact form for general quasi-Newton methods. For the ℓ -BFGS Hessian approximation of Eq. (3.37), and using the diagonal scaling matrix $\mathbf{H}_0 = [\mathbf{I}_0]^{-1}$,

^{11}The ℓ -BFGS two-loop recursive algorithm is the implementation in *Spinach*, and the one used in all ℓ -BFGS optimisations of this thesis

this compact notation is [34]:

$$\mathbf{H}_j^{\text{BFGS}} = \mathbf{H}_0 - \begin{bmatrix} \mathbf{H}_0 \Delta \mathbf{C}_j & \Delta \mathbf{G}_j \end{bmatrix} \begin{bmatrix} \mathbf{H}_0 \langle \Delta \mathbf{C}_j | \Delta \mathbf{C}_j \rangle & \mathbf{L}_j \\ \mathbf{L}_j^\dagger & -\mathbf{K}_j \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{H}_0 \Delta \mathbf{C}_j^\dagger \\ \Delta \mathbf{G}_j^\dagger \end{bmatrix} \quad (3.43)$$

where the stores of $\Delta \mathbf{c}$ and $\Delta \mathbf{g}$ are $n \times m$ matrices defined by

$$\Delta \mathbf{C}_j \triangleq \begin{bmatrix} \Delta \mathbf{c}_{j-m} & \Delta \mathbf{c}_{j-m+1} & \cdots & \Delta \mathbf{c}_{j-1} \end{bmatrix} \quad (3.44a)$$

$$\Delta \mathbf{G}_j \triangleq \begin{bmatrix} \Delta \mathbf{g}_{j-m} & \Delta \mathbf{g}_{j-m+1} & \cdots & \Delta \mathbf{g}_{j-1} \end{bmatrix} \quad (3.44b)$$

\mathbf{L}_j is an $m \times m$ matrix defined by

$$(\mathbf{L}_j)_{p,q} \triangleq \begin{cases} \langle \Delta \mathbf{c}_{j-m-1+p} | \Delta \mathbf{g}_{j-m-1+q} \rangle & p > q \\ 0 & p \leq q \end{cases} \quad (3.45)$$

and \mathbf{K}_j is a diagonal $m \times m$ matrix defined by

$$\mathbf{K}_j \triangleq \begin{bmatrix} \langle \Delta \mathbf{c}_{j-m} | \Delta \mathbf{g}_{j-m} \rangle & 0 & \cdots & 0 \\ 0 & \langle \Delta \mathbf{c}_{j-m+1} | \Delta \mathbf{g}_{j-m+1} \rangle & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \langle \Delta \mathbf{c}_{j-1} | \Delta \mathbf{g}_{j-1} \rangle \end{bmatrix} \quad (3.46)$$

The central matrix in EQ. (3.43) is of small size, $2m \times 2m$, when compared to the number of optimisation variables, n : their factorisation requires negligible amount of computation [229]. This ℓ -BFGS method is an update to the Hessian approximation, and so allows the implementation of Hessian conditioning methods.

Using the definitions above, an ℓ -SR1 algorithm can be derived as [34]:

$$\mathbf{H}_j^{\text{SR1}} = \mathbf{H}_0 + \left[\Delta \mathbf{G}_j - \mathbf{H}_0 \Delta \mathbf{C}_j \right] \left[\mathbf{K}_j + \mathbf{L}_j + \mathbf{L}_j^\dagger - \Delta \mathbf{C}_j^\dagger \mathbf{H}_0 \Delta \mathbf{C}_j \right]^{-1} \left[\Delta \mathbf{G}_j - \mathbf{H}_0 \Delta \mathbf{C}_j \right]^\dagger \quad (3.47)$$

Further compact notations can be derived for all the Broyden class of quasi-Newton methods, for inverse-Hessian and Hessian approximations, allowing design of their limited-memory algorithms [34].

§ 3.1.4 Line search subproblem

This section will present criteria for an acceptable point using the scalar quantity called the *step length*, mentioned in Eqs. (3.11a), (3.11b), (3.18), (3.21), deciding how far to move along the search direction.

The *linear approximation* to a function in §3.1.1, and the *quadratic approximation* to a function in §§3.1.2, 3.1.3, usually need more than their derived search direction to find an acceptable point. Although the gradient based methods will step in the correct direction, without a step length, the methods could wildly misjudge a predicted extremum. Algorithms to find an appropriate step length are called *line search methods* or *inexact line search*. The update to an objective variable at the next iteration is

$$\mathbf{c}_{j+1} = \mathbf{c}_j + \varepsilon \mathbf{d}_j \quad (3.48)$$

where the search direction is calculated with a method chosen from Eqs. (3.10a), (3.10b), (3.17), (3.19a), (3.19b), (3.24a), (3.24b). The line search subproblem can be defined as a minimisation^{I} problem in itself:

$$\min_{\varepsilon > 0} J(\mathbf{c}_j + \varepsilon \mathbf{d}_j) \quad (3.49)$$

The exact minimisation defined by Eq. (3.49) is not required, and may not find this solution in a finite number of iterations, only an approximation of ε is required to progress the optimisation procedure.

In deciding if the approximation is acceptable, a number of conditions should be tested. The first of these conditions is trivial, $J(\mathbf{c}_{j+1}) < J(\mathbf{c}_j)$, and should be accounted for with calculation of a search direction giving a descent direction:

$$\langle \mathbf{d}_j | \nabla J(\mathbf{c}_j) \rangle < 0 \quad (3.50)$$

However, this alone does not guarantee convergence. There must be *sufficient decrease* in the slope, $\langle \nabla J(\mathbf{c}) | \mathbf{c} \rangle$, of the objective along the search direction. Assuming that an iterative algorithm produces calculated step lengths, ε , the acceptable

^{I}A maximisation problem follows a similar definition

$$\max_{\varepsilon > 0} J(\mathbf{c}_j + \varepsilon \mathbf{d}_j)$$

The rest of this section will follow a minimisation problem, but a maximisation problem is easily accounted for.

step length should satisfy

$$\begin{aligned} & \text{Armijo} \\ & \text{condition: } J(\mathbf{c}_j + \varepsilon \mathbf{d}_j) \leq J(\mathbf{c}_j) + \kappa_1 \varepsilon \langle \nabla J(\mathbf{c}_j) | \mathbf{d}_j \rangle, \quad \kappa_1 \in (0, 1) \end{aligned} \quad (3.51)$$

for a pre-defined constant, κ_1 . This condition is known as the *sufficient decrease condition* or the *Armijo condition* [8], and the constant is chosen to be small, usually $\kappa_1 \sim 10^{-4}$.

The condition of EQ. (3.51) will be satisfied for small ε , and the line search algorithm may make insufficient progress by underestimating an acceptable step length. In addition to the condition in EQ. (3.51), a condition known as the *Goldstein condition* [104], ensures that the step length is not underestimated:

$$\begin{aligned} & \text{Goldstein} \\ & \text{condition: } J(\mathbf{c}_j + \varepsilon \mathbf{d}_j) \geq J(\mathbf{c}_j) + (1 - \kappa_1) \varepsilon \langle \nabla J(\mathbf{c}_j) | \mathbf{d}_j \rangle, \quad \kappa_1 \in (0, 1/2) \end{aligned} \quad (3.52)$$

where the bounds of κ_1 are now more narrow than in EQ. (3.51).

An alternative to EQ. (3.52) in avoiding underestimation of the step length is to account for curvature. This requires the gradient to be calculated at each step length calculation but should give a much better estimation of EQ. (3.49). The *curvature condition* [333, 334] requires ε to satisfy

$$\begin{aligned} & \text{Curvature} \\ & \text{condition: } \langle \nabla J(\mathbf{c}_j + \varepsilon \mathbf{d}_j) | \mathbf{d}_j \rangle \geq \kappa_2 \langle \nabla J(\mathbf{c}_j) | \mathbf{d}_j \rangle, \quad \kappa_2 \in (\kappa_1, 1) \end{aligned} \quad (3.53)$$

This ensures that the slope found during step length iteration, ε_j , is greater than the slope at $\varepsilon = 1$. The constant κ_2 is usually chosen to be relatively large for quasi-Newton and Newton methods [229], $\kappa_2 \simeq 0.9$, but should be reduced for gradient descent methods, $\kappa_2 \simeq 0.1$. Eqs. (3.51), (3.53) are known as the *Wolfe conditions* [333, 334].

The *sufficient decrease condition* stipulates that the decrease in $J(\mathbf{c})$ must be proportional to the step length and directional derivative. We can extend the *weak curvature condition* of EQ. (3.53) to ensure that the step length is in the region of a local minimum, ensuring the derivative in EQ. (3.53) is not too positive, by using a *strong curvature condition* [333, 334]:

$$\begin{aligned} & \text{Strong Wolfe} \\ & \text{condition: } \left| \langle \nabla J(\mathbf{c}_j + \varepsilon \mathbf{d}_j) | \mathbf{d}_j \rangle \right| \geq \kappa_2 \left| \langle \nabla J(\mathbf{c}_j) | \mathbf{d}_j \rangle \right|, \quad \kappa_2 \in (\kappa_1, 1) \end{aligned} \quad (3.54)$$

This equation, together with Eq. (3.51), are known collectively as the *strong Wolfe conditions*. These conditions are particularly effective when used in a quasi-Newton method from §3.1.3.

Backtracking line search

Perhaps the most simple line search method is to iteratively reduce the step length by a predefined factor until Eq. (3.51) is satisfied. This algorithm is called the *backtracking line search*, and is useful when gradient calculations are accurate but expensive. The iterative reduction of ε follows

$$\varepsilon_{j+1} = \beta \varepsilon_j \quad \text{while} \quad J(\varepsilon_j \mathbf{d}_j) \geq \kappa_1 \langle \nabla J(\mathbf{c}_j) | \mathbf{d}_j \rangle \quad (3.55)$$

where $\beta \in (0, 1)$, continuing until Eq. (3.51) is satisfied, when $\varepsilon \rightarrow \varepsilon_j$. The initial step length is chosen as the Newton step, $\varepsilon_0 = 1$, for quasi-Newton and Newton methods. The initial step length can be chosen to be larger for gradient methods and sometimes quasi-Newton methods to avoid small steps at the initial iterations of an optimisation, far from an extremum^{I}.

Bracketing & sectioning line search

A bracketing and sectioning line search starts with *bracketing phase* which contains an interval of acceptable points, $[a, b]$. Once the bracket is identified, a *sectioning phase* follows dividing the bracket to give a sequence of brackets, $[a_i, b_i]$, where their length tends to zero [79].

Assuming gradients are available and they are cheap to calculate^{II}, the bracketing and sectioning line search can use the Wolfe conditions in Eqs. (3.51), (3.53), (3.54).

The initial step in the bracketing phase would be choosing an initial step length, usually this is the Newton-step $\varepsilon_0 = 1$ as in the backtracking line search. Assuming a lower bound on an acceptable point, J_{\min} , and a line search is restricted to the

^{I}A suggestion in [229] is to set $\varepsilon_0 = \frac{1}{\|\nabla J(\mathbf{c}_j)\|_\infty}$

^{II}It will be outlined later in this chapter, in §3.2.5, that gradients needed in optimal control used in this thesis are particularly cheap to calculate, following at little extra cost to calculating the required objective function value.

interval $(0, \mu]$ where

$$\mu = \frac{J_{\min} - J(\mathbf{c})}{\kappa_1 \nabla J(\mathbf{c})} \quad (3.56)$$

a general procedure starting from $J(\mathbf{c} + \varepsilon_0 \mathbf{d})$ at an iteration (the optimisation iteration j does not change during a line search) j , set out in [79] is

(1) Set $j = 1$.

(2) Calculate $J(\mathbf{c} + \varepsilon_j \mathbf{d})$, where $0 < \varepsilon_j \leq \mu$.

(3) If $J(\mathbf{c} + \varepsilon_j \mathbf{d}) \leq J_{\min}$,

terminate with exit message *Boundary J_{\min} has been reached*.

(4) If $J(\mathbf{c} + \varepsilon_j \mathbf{d}) > J(\mathbf{c} + \mathbf{d}) + \varepsilon_j \nabla J(\mathbf{c})$ or $J(\mathbf{c} + \varepsilon_j \mathbf{d}) \geq J(\mathbf{c} + \varepsilon_{j-1} \mathbf{d})$ go to STEP (4).I, else go to STEP (5).

- I. Then $a_i \rightarrow \varepsilon_{j-1}$ and $b_i \rightarrow \varepsilon_j$,
Bracket located – case 1 (Wolfe conditions).
Continue to **Sectioning phase**.

(5) Calculate $\nabla J(\mathbf{c} + \varepsilon_j \mathbf{d})$.

(6) If $|\nabla J(\mathbf{c} + \varepsilon_j \mathbf{d})| \leq -\kappa_2 \nabla J(\mathbf{c})$, **terminate**.

(7) If $|\nabla J(\mathbf{c} + \varepsilon_j \mathbf{d})| \geq 0$, go to STEP (7).I, else go to STEP (8).

- I. Then $a_i \rightarrow \varepsilon_j$ and $b_i \rightarrow \varepsilon_{j-1}$,
Bracket located – case 2 (Wolfe conditions).
Continue to **Sectioning phase**.

(8) If $\mu \leq 2\varepsilon_j - \varepsilon_{j-1}$, go to STEP (8).I, else go to STEP (8).II.

- I. Then $\varepsilon_{j+1} \rightarrow \mu$.
- II. Else choose $\varepsilon_{j+1} \in \{2\varepsilon_j - \varepsilon_{j-1}, \min(\mu, \varepsilon_j + \tau_1(\varepsilon_j - \varepsilon_{j-1}))\}$.
Continue to **Interpolation phase**

(9) $j = j + 1$, return to STEP (2).

Once a bracket has been identified the sectioning phase can begin:

- ① Set $k = j$.
- ② Choose $\varepsilon_k \in \{a_k + \tau_2(b_k - a_k), b_k - \tau_3(b_k - a_k)\}$.
Continue to **Interpolation phase**
- ③ Calculate $J(\mathbf{c} + \varepsilon_k \mathbf{d})$.
- ④ If $J(\mathbf{c} + \varepsilon_k \mathbf{d}) > J(\mathbf{c} + \mathbf{d}) + \kappa_1 \varepsilon_k \nabla J(\mathbf{c})$ or $J(\mathbf{c} + \varepsilon_k \mathbf{d}) \geq J(\mathbf{c} + a_k \mathbf{d})$, go to STEP ④.I, else go to STEP ⑤.
 - I. $a_{k+1} \rightarrow a_k$ and $b_{k+1} \rightarrow \varepsilon_k$, go to STEP ⑨.
- ⑤ Calculate $\nabla J(\mathbf{c} + \varepsilon_k \mathbf{d})$.
- ⑥ If $|\nabla J(\mathbf{c} + \varepsilon_k \mathbf{d})| \leq -\kappa_2 \nabla J(\mathbf{c})$,
terminate with an acceptable step length.
- ⑦ $a_{k+1} \rightarrow \varepsilon_k$, go to STEP ⑨.
- ⑧ If $b_k - a_k) \nabla J(\mathbf{c} + \varepsilon_k \mathbf{d}) \geq 0$, go to STEP ⑧.I, else go to STEP ⑧.II.
 - I. $b_{k+1} \rightarrow a_k$.
 - II. $b_{k+1} \rightarrow b_k$.
- ⑨ $k = k + 1$, return to STEP ②.

The constants $\tau_1 > 0$ and $0 < \tau_2 < \tau_3 \leq 1/2$ are set as the following in *Spinach*: Bracketing expansion factor $\tau_1 = 3$; Left bracket contraction factor $\tau_2 = 0.1$; Right bracket contraction factor $\tau_3 = 0.5$.

It is common to include an *interpolation phase*, STEPS ⑧.II, ②, particularly within the sectioning phase, to find a point close to a local minimiser. Here, a quadratic or cubic polynomial if fitted to data from the acceptable bracket^{11}.

^{11}The work in this thesis uses a cubic polynomial interpolation.

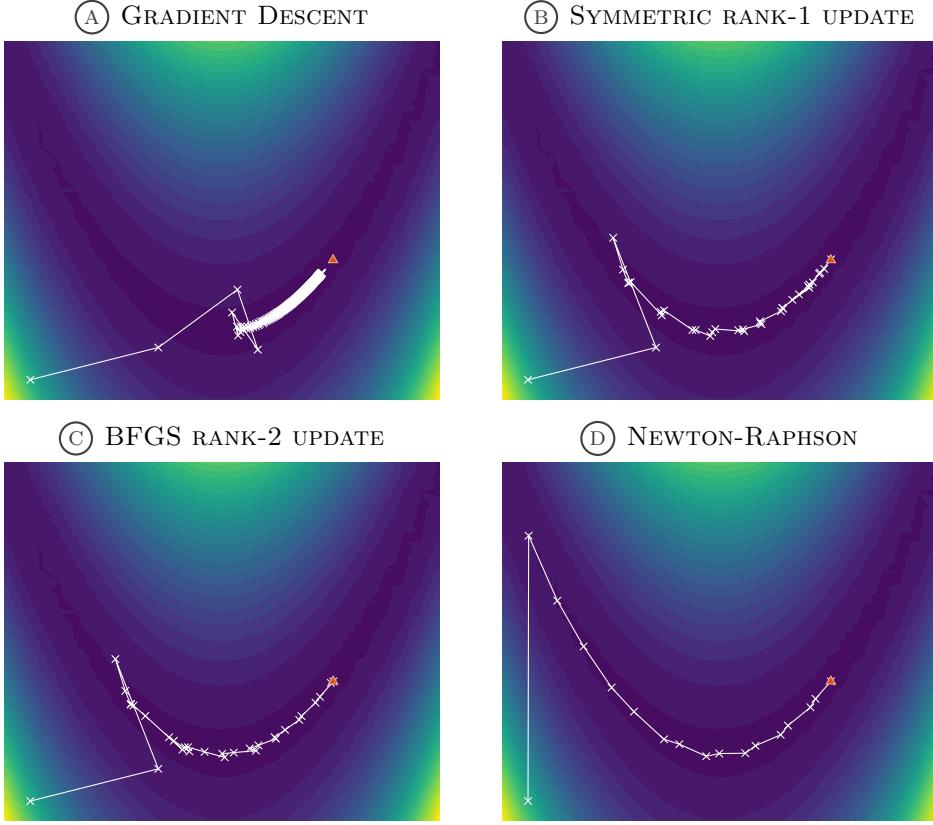


FIGURE 3.2: Comparison of gradient based optimisation methods on the Rosenbrock function [252], with a backtracking line-search method. The optimisation algorithms terminate when $f(x) < 5 \times 10^{-2}$. (A) – gradient descent method, taking 376 iterations with 5101 functional evaluations. (B) – symmetric rank-1 update, taking 33 iterations with 185 functional evaluations. (C) – BFGS rank-2 update, taking 29 iterations with 78 functional evaluations. (D) – Newton-Raphson method, taking 16 iterations with 43 functional evaluations.

§ 3.1.5 Comparison of Newton-type methods

A common test for optimisation methods is on the Rosenbrock function $(1-x)^2 + 100(y-x^2)^2$ [252]. The function has one minimum at $(1, 1)$, but the difficulty in its minimisation is that the valley is very flat. Although it is easy to find the valley, once there, optimisation usually converges slowly to the point $(1, 1)$.

FIG. 3.2 compares the convergence of the gradient descent method of Eq. (3.11a), the Newton-Raphson method of Eq. (3.18), and two quasi-Newton method of symmetric rank-1 update, Eq. (3.31), and the BFGS rank-2 update in Eq. (3.38).

It is clear that the gradient descent method performs badly, even when using an analytical gradient. The convergence trajectory is seen zig-zagging in small

steps on either side of the valley. It is clear that the gradient used in this way is quite ineffective. The Newton-Raphson method performs the best, with an analytic Hessian to use local curvature information in finding the best route down the valley. The quasi-Newton methods also perform quite well, starting from a gradient descent iteration, then being able to approximate the curvature within the valley to find the best route, although taking smaller steps compared to the Newton-Raphson method. Comparing the two quasi-Newton methods, the rank-1 and rank-2 updates both have a similar trajectory iteration count, however, the rank-1 update puts a greater load on the line search algorithm – needing more than twice the amount of gradient evaluations to get similar curvature information compared to the rank-2 update.

§ 3.2 Gradient Ascent Pulse Engineering

The task of taking a quantum system from one state to another to a specified accuracy with minimal expenditure of time and energy, with the emphasis on the word *minimal*, is increasingly important in physics and engineering [154, 159, 227, 311].

Optimal control can be thought of as an algorithm; there is a start and a stop. in the language of physics; a dynamic system with a initial state, $|\rho_0\rangle$, and a desired destination state, $|\sigma\rangle$. A solution should be found with a minimum of effort, an optimal amount of effort. In forming a problem for use in a numerical optimisation algorithm, the first step is to create a system model. Within this model the controllable parameters are exposed then fed into an optimisation algorithm.

§ 3.2.1 Bilinear systems

The Hamiltonian of a practical NMR quantum system can be split into two parts [152]: that which is beyond the control of the instrument and a *control* part that measurement instrumentation can vary within certain limits.

$$\hat{\mathcal{H}}(t) = \hat{\mathcal{H}}_0 + i\hat{\mathcal{R}} + \sum_{k=1}^K c_k(t)\hat{\mathcal{H}}_k \quad (3.57)$$

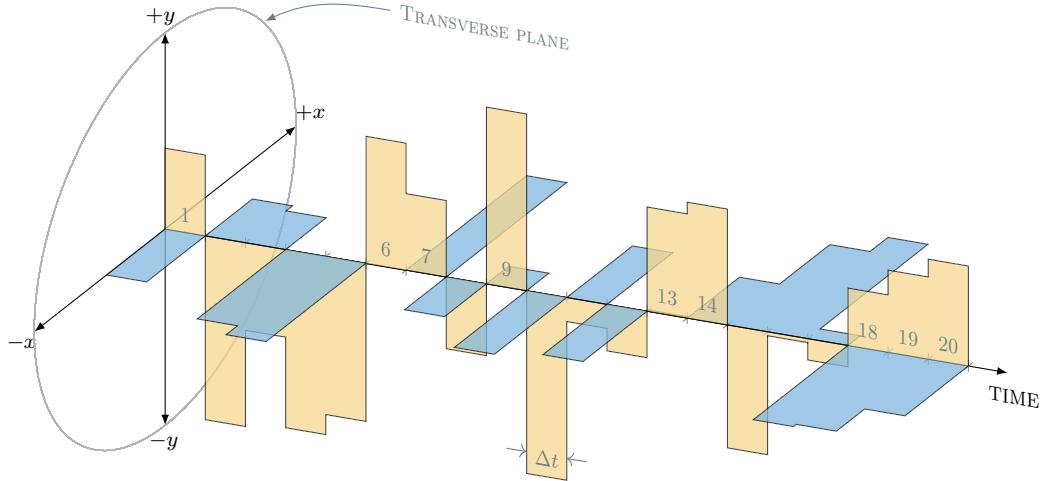


FIGURE 3.3: Diagram of the piecewise-constant approximation, for 20 equal time slices of length Δt . Pulses are performed simultaneously in the x and y directions of the transverse magnetisation plane.

$\hat{\mathcal{H}}_0$ is the Hamiltonian describing the internal part of the system called the *drift* or *free Hamiltonian*, and $\hat{\mathcal{H}}_k$ are the K parts of the Hamiltonian that can be changed externally, called the *controls*, and $c_k(t)$ are their time-dependent coefficients. In the case of magnetic resonance the instrumentally controllable subsystem consists of radio frequency/microwave control fields defined in §2.4.3. $\hat{\mathcal{R}}$ is the relaxation superoperator characterising dissipation from the system, and has been separated from $\hat{\mathcal{H}}_0$ for convenience [103, 166, 250].

Systems with equations of motion that are linear both in the drift and controls [2], as in Eq. (3.57), are known as *bilinear control systems* [274, 275, 291, 336] in classical control theory. Optimal control of the bilinear system in Eq. (3.57) is the task of finding the time-dependent control amplitudes $c_k(t)$ – controlling the system with their corresponding controls $\hat{\mathcal{H}}_k$, while under the influence of the time-independent drift $\hat{\mathcal{H}}_0^{\{\mathbf{I}\}}$ – taking the system from a defined initial state, $|\rho_0\rangle$, to a desired final state, $|\sigma\rangle$, a time T later.

§ 3.2.2 Piecewise constant approximation

The evolution of a quantum ensemble is governed by the Liouville-von Neumann equation (EQ. (2.51)). Using a column-wise vector representation of the density operator, $\hat{\rho}$, the Liouville-von Neumann equation has the general solution from EQ. (2.57). With the separated Hamiltonian formalism of EQ. (3.57), this gives

$$|\rho(T)\rangle = \exp_{(o)} \left[-i \int_0^T \left(\hat{\mathcal{H}}_0 + i\hat{\mathcal{R}} + \sum_{k=1}^K c_k(t) \hat{\mathcal{H}}_k \right) dt \right] |\rho_0\rangle \quad (3.58)$$

An approximation of the integral in EQ. (3.58) can be made by assuming that $c_k(t)$ are piecewise-constant [43, 154], depicted in FIG. 3.3; *slicing* the problem into discrete time intervals to obtain control sequences

$$c_k(t) \longrightarrow \mathbf{c}_k \triangleq \begin{pmatrix} c_k(t_1) & c_k(t_2) & \cdots & c_k(t_N) \end{pmatrix}, \quad t_1 < t_2 < \cdots < t_N \quad (3.59)$$

where the individual elements $c_k(t_n) = c_{k,n}$ are treated as continuous parameters. With this approximation, the evolution of the system from a state $\rho(t_0)$ to $\rho(t_N)$ is

$$|\rho(t_N)\rangle = \left[\prod_{n=1}^N \exp \left[-i \left(\hat{\mathcal{H}}_0 + i\hat{\mathcal{R}} + \sum_{k=1}^K c_{k,n} \hat{\mathcal{H}}_k \right) \Delta t \right] \right] |\rho(t_0)\rangle \quad (3.60)$$

where the product must be *time-ordered* and the time interval Δt . Time intervals considered in the remainder of this thesis will be the same for each time slice, although this is for convenience and not a restriction of the piecewise constant approximation.

Although the physical arrangement of the control vector \mathbf{c} is as K rows of control sequences vectors, the design of an optimisation algorithm is simpler with a single vector representing the many variables of its scale objective function. For this reason, the control vector is concatenated to column vector when used within the

^{1}Relaxation will be neglected in many of the derivations in this thesis. For convenience, it will be considered an uncontrollable *damping term*. Although this is only an approximation, the control of open systems is beyond the scope of this thesis.

optimiser:

$$\mathbf{c} = \begin{pmatrix} c_{1,1} & c_{1,2} & \cdots & c_{1,N} \\ c_{2,1} & c_{2,2} & \cdots & c_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ c_{K,1} & c_{K,2} & \cdots & c_{K,N} \end{pmatrix} \longmapsto \begin{pmatrix} c_{1,1} \\ c_{2,1} \\ \vdots \\ c_{K,1} \\ c_{1,2} \\ c_{2,2} \\ \vdots \\ c_{K,2} \\ \vdots \\ \vdots \\ c_{1,N} \\ c_{2,N} \\ \vdots \\ c_{K,N} \end{pmatrix} \quad (3.61)$$

Although the order (k, n) is used here, the ordering is arbitrary and could equally be ordered (n, k) as long as the vectorisation of the control amplitudes is consistent within the optimiser.

§ 3.2.3 The Pontryagin maximum principle

To investigate the gradient that would be needed for numerical optimisation, in § 3.1, a principle from optimal control theory needs to be imposed, called the *Pontryagin maximum principle* [92, 246]. It follows from the classical Lagrangian mechanics [174], which introduces a *Lagrangian*, $\mathcal{L}(t, \rho(t), \dot{\rho}(t))$, – a function that characterises the dynamics a system. In this case, the Lagrangian \mathcal{L} is defined by the variables of time t , the state of the system $\rho(t)$, and the equation of motion of the state $\dot{\rho}(t)$. Lagrangian mechanics is the calculus of variations, and so too is optimal control.

The Pontryagin maximum principle sets an optimal control problem in a similar formulation, with the modification of replacing the equation of motion with the system controls, giving the Lagrangian as $\mathcal{L}(t, \rho(t), c_k(t))$ and the equation of motion depends on the controls. In addition to its Lagrangian formulation, optimal control is also the application of mathematical optimisation, and its solutions are set out by the conditions of Pontryagin maximum principle.

Physical limitations of the instrumentation and constraints on control variable are part of the optimal control problem, and should be formulated within the problem [298]. In the context of magnetic resonance, the optimal control problem is set out as [77, 176, 287]:

$$\max_{c(t)} (\mathcal{L}), \quad \mathcal{L}(c(t)) \triangleq \mathcal{J}(\rho(T)) + \int_0^T \mathcal{K}(t, \rho(t), c(t)) dt \quad (3.62)$$

where the scalar function $\mathcal{J}(\rho(T))$ is a *terminal cost* depending only on the final state of the system at a time T , and the integral of the scalar function $\mathcal{K}(t, \rho(t), c(t))$ is a *running cost* which depends on the controls and the state of the system during the time interval $[0, T]$. The controls, $c(t)$, affect the equation of motion of the system – the Liouville-von Neumann equation of Eq. (2.56)

$$\frac{d}{dt} |\rho(t)\rangle = -i \left[\int_0^T \hat{\mathcal{H}}(c(t)) dt \right] |\rho_0\rangle \quad (3.63)$$

where $\hat{\mathcal{H}}(t)$ can be separated as a bilinear system in Eq. (3.57).

In analogy to *Lagrange multipliers* of Lagrangian mechanics [174] a variable called the *adjoint state* (also known as the *costate*) is introduced to solve the optimal control problem. Without exploring the dry depths of Lagrangian mechanics, the *adjoint state* will be introduced by example.

The formal mathematical description of the adjoint states will follow, but it is useful to show an interpretation of their physical significance with an example of state-to-state transfer in FIG. 3.4. The state of the system at each time slice forms a state trajectory over a time interval $[0, T]$, containing information of the basis state population at each time increment. If propagated from a defined initial state $|\rho_0\rangle$ with Eq. (3.75), the control pulses form a unique trajectory of the system under the influence of the drift.

FIG. 3.4 shows a projection of the state of the system at each time interval onto a representation of magnetisation population local to each spin. The aim is to transfer magnetisation from one spin to the other in a coupled two-spin system with calculated control pulses. It is easy to see the population gradually transfers from one spin to the other over the time interval $[0, T]$.

The state trajectory is the trajectory formed by propagating the system from a defined initial state under the effect of control pulses. The *adjoint state* also forms a

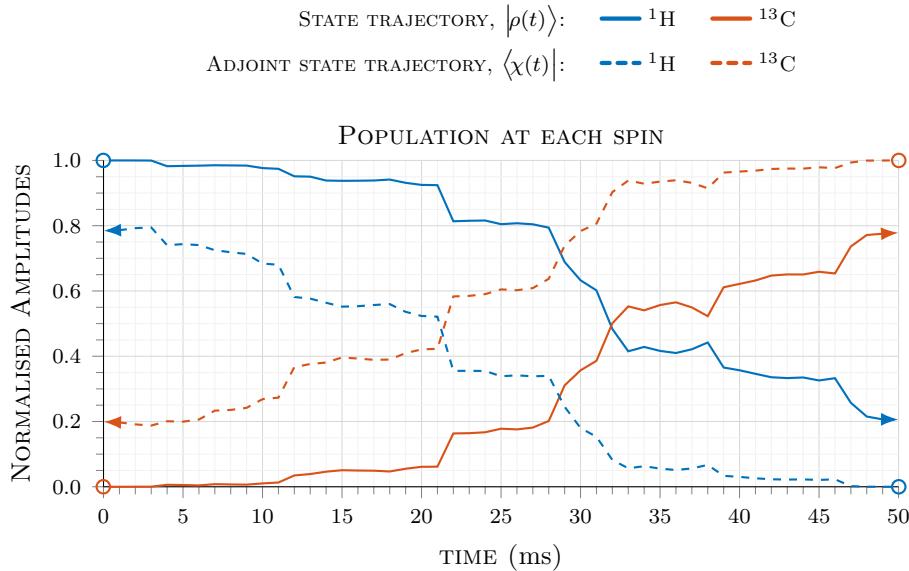


FIGURE 3.4: An example of two state vector trajectories, and the corresponding adjoint state trajectories. The state vector propagates forward in time from the initial condition, under the influence of drift and calculated controls. The adjoint state propagates backwards in time from the desired state, under the influence of the time-reversed drift and controls. This example shows polarisation transfer from a hydrogen spin to a coupled carbon spin (140 Hz). The two trajectories show polarisation local to each of the spins. In this case, nearly 80% of the polarisation has been transferred with calculated control pulses (not shown).

trajectory, propagated under the influence of the same drift and the same controls. However, the adjoint state is a propagation backwards in time from the desired final state of the system. The *adjoint state* trajectory is the control problem in reverse.

When an optimal trajectory is found, the state trajectory should end at the desired final state, and the adjoint state should start at the defined initial state of the system. This can be seen in FIG. 3.4, where an optimal trajectory has not yet been found; there is a population difference between the adjoint state trajectory and the state trajectory at times $t = 0$ and at times $t = T$. The adjoint state $|\chi(T)\rangle = |\sigma\rangle$, and the state $|\rho(0)\rangle = |\rho_0\rangle$ form the boundary conditions for the optimal control problem.

The *Pontryagin maximum principle* requires a number of conditions to be met for an optimal trajectory to be found. Just as in the area of numerical optimisation of §3.1, necessary conditions of the first and second derivatives must

be met (Eqs. (3.7), (3.8)), in this case applied to the function called a *pseudo-Hamiltonian*^{1} [77, 176], derived from Eq. (3.62) and the *adjoint state* [92, 246]:

$$\mathcal{H}_P(\rho(t), c(t)) \triangleq \left\langle \chi(t) \left| -i \left[\int_0^T \hat{\mathcal{H}}(c(t)) dt \right] \right| \rho(t) \right\rangle + \mathcal{K}(\rho(t), c(t)) \quad (3.64)$$

The first and second order optimality conditions of Eqs. (3.7), (3.8), now become

$$\frac{\partial \mathcal{H}_P}{\partial c(t)} = 0 \quad (3.65a)$$

$$\frac{\partial^2 \mathcal{H}_P}{\partial c(t)^2} > 0 \quad (3.65b)$$

in the vicinity of a maximum. The *Pontryagin maximum principle* [246] also requires consideration of the relationship between the state trajectory and the adjoint state trajectory. These give the further necessary conditions

$$\frac{d}{dt} |\chi(t)\rangle = -\frac{\partial \mathcal{H}_P}{\partial |\rho(t)\rangle} \quad (3.66a)$$

$$\frac{d}{dt} |\rho(t)\rangle = \frac{\partial \mathcal{H}_P}{\partial \langle \chi(t)|} \quad (3.66b)$$

which formally sets out the conclusions made in discussion of FIG. 3.4 above: as the state trajectory approaches as maximum, the state trajectory as a time $t = T$ should completely overlap with the desired target state, and that the adjoint state at a time $t = 0$ should overlap completely with the initial state of the system.

The formulation of Eq. (3.62) admits three types of optimal control problem [45, 298]: *The Lagrange problem* has only a running cost; *The Mayer problem* has only a terminal cost; *The Bolza problem* both the running cost and the terminal cost are included. The Mayer problem, where the final state of the system forms the optimisation problem, will be the main consideration of this thesis until §6, when control amplitude penalties are included in a Bolza problem.

For an unconstrained optimal control problem $\mathcal{K} = 0$ [77] and the optimal control problem reduces to one of the Mayer problem. The *terminal cost* \mathcal{J} in Eq. (3.62) will be the focus of the next section on *fidelity* measures.

^{1}The term *pseudo-Hamiltonian* is introduced to distinguish it from Hamiltonians of quantum mechanics. The *pseudo-Hamiltonian* is a canonical system and therefore it is a Hamiltonian function [92, 298]

§ 3.2.4 Fidelity measures

Optimal solutions can be found numerically, by optimising *fidelity* – the overlap between the final state of the system and the desired destination state [99, 154, 202]. The optimum of the fidelity functional $\mathcal{J}(c)$ with respect to the control sequences $c_k(t)$ in Eq. (3.57), subjected to experimental constraints of feasible amplitudes and frequencies, is calculated with *optimal control theory* [191, 246]. For Hermitian states, the fidelity function is

$$\mathcal{J}_0 \triangleq \langle \sigma | \rho(T) \rangle \quad \mathcal{J}_0 \in [-1, +1] \quad (3.67)$$

where angular bracket denotes column-wise vectorization of the corresponding density matrix, $|\rho(T)\rangle$ is the final state, $|\sigma\rangle$ is the desired destination state. Finding the optimal value of this inner product, is that of finding its maximum.

Non-Hermitian states can be accounted for with two further definitions of the fidelity functional, either by using only the real part of the overlap in Eq. (3.67)

$$\mathcal{J}_1 \triangleq \text{Re} \langle \sigma | \rho(T) \rangle \quad \mathcal{J}_1 \in [-1, +1] \quad (3.68)$$

or by using the modulus square^{I} of the overlap in Eq. (3.67)

$$\mathcal{J}_2 \triangleq \left| \langle \sigma | \rho(T) \rangle \right|^2 \quad \mathcal{J}_2 \in [0, +1] \quad (3.69)$$

The definition of \mathcal{J}_1 allows the optimisation to span a space from that parallel to, orthogonal to, and anti-parallel to σ . The definition of \mathcal{J}_2 is phase insensitive. For the fidelity to have this physical meaning, the initial state and the system, $|\rho_0\rangle$, and the desired target state, $|\sigma\rangle$, should be be normalised.

In addition to these state-to-state definitions of fidelity, similar measures can be designed for multi-state mapping:

$$\mathcal{J}_{\text{UR}} = \frac{1}{q_{\max}} \text{Re} \left(\sum_{q=1}^{q_{\max}} \langle \sigma_q | \rho_q(T) \rangle \right) \quad (3.70)$$

^{I}The modulus square of a complex valued vector is $|\mathbf{v}|^2 = \mathbf{v}\mathbf{v}^*$, where $*$ denotes the complex conjugate.

where σ_k define the desired state propagated in time from corresponding initial states $\rho_k(0)$ respectively. When Eq. (3.70) is averaged over 3 state-to-state problems, together completely characterising a rotation of the state in Cartesian coordinates, the optimal control problem becomes a universal rotation in 3D e.g.

$$\begin{aligned} |\rho_1(0)\rangle &= +|L_z\rangle \longrightarrow -|L_x\rangle = |\sigma_1\rangle \\ |\rho_2(0)\rangle &= +|L_x\rangle \longrightarrow -|L_z\rangle = |\sigma_2\rangle \\ |\rho_3(0)\rangle &= +|L_y\rangle \longrightarrow -|L_y\rangle = |\sigma_3\rangle \end{aligned}$$

These 3 state-to-state problems can be restated with a fidelity definition to find an *effective propagator* [154] or an *effective Hamiltonian* [310] over the time period of, and subject to, the control sequence. A measure of the distance between two matrices can be defined by the desired propagation over the total time period $[0, T]$.

The fidelity definition for this desired *effective propagator* is

$$\mathcal{J}_3 \triangleq \text{Re}\langle \hat{\mathcal{U}}_D | \hat{\mathcal{P}}(T) \rangle \quad \mathcal{J}_3 \in [-1, +1] \quad (3.71)$$

where $\hat{\mathcal{U}}_D = \exp(-i\hat{\mathcal{H}}_D T)$ is the desired effective propagator for the effective Hamiltonian $\hat{\mathcal{H}}_D$. In analogy to Eq. (3.69), the modulus square of this overlap is

$$\mathcal{J}_4 \triangleq \left| \langle \hat{\mathcal{U}}_D | \hat{\mathcal{P}}(T) \rangle \right|^2 \quad \mathcal{J}_4 \in [0, +1] \quad (3.72)$$

In Eqs. (3.71), (3.72), the overlap of two propagators is defined by the Frobenius inner product as the trace of the operator product:

$$\langle \hat{\mathcal{U}}_D | \hat{\mathcal{P}}(T) \rangle \triangleq \text{Tr}(\hat{\mathcal{U}}_D^\dagger \hat{\mathcal{P}}(T)) \quad (3.73)$$

Although not used in this thesis, the definitions of fidelity in Eqs. (3.71), (3.72) are reported to avoid local minima which Eq. (3.70) encounter [154, 310].

§ 3.2.5 Fidelity gradient

To make use of Newton-type methods of §3.1, a gradient of the fidelity functional with respect to the control amplitudes is required. The first step to achieve this is to expand the fidelity definitions in Eqs. (3.67)–(3.69). As a function of control

amplitudes, the fidelity definitions become

$$\mathcal{J}_0(\mathbf{c}_n) = \langle \sigma | \hat{\mathcal{P}}_N \hat{\mathcal{P}}_{N-1} \cdots \hat{\mathcal{P}}_2 \hat{\mathcal{P}}_1 | \rho_0 \rangle \quad (3.74a)$$

$$\mathcal{J}_1(\mathbf{c}_n) = \text{Re} \langle \sigma | \hat{\mathcal{P}}_N \hat{\mathcal{P}}_{N-1} \cdots \hat{\mathcal{P}}_2 \hat{\mathcal{P}}_1 | \rho_0 \rangle \quad (3.74b)$$

$$\mathcal{J}_2(\mathbf{c}_n) = \left| \langle \sigma | \hat{\mathcal{P}}_N \hat{\mathcal{P}}_{N-1} \cdots \hat{\mathcal{P}}_2 \hat{\mathcal{P}}_1 | \rho_0 \rangle \right|^2 \quad (3.74c)$$

where the time propagator of EQ. (2.53) is now piecewise constant

$$\hat{\mathcal{P}}_n(\mathbf{c}_n) = \exp \left[-i \left(\hat{\mathcal{H}}_0 + i \hat{\mathcal{R}} + \sum_{k=1}^K c_{k,n} \hat{\mathcal{H}}_k \right) \Delta t \right] \quad (3.75)$$

where the definition of the propagator at each time slice, n , is dependent on the K control amplitudes in the vector array

$$\mathbf{c}_n = \begin{pmatrix} c_1(t_n) \\ c_2(t_n) \\ \vdots \\ c_K(t_n) \end{pmatrix} \quad (3.76)$$

Once the controllable part of the Hamiltonian becomes parametrised, the variation $\delta \mathcal{J} / \delta \hat{\mathcal{H}}_k$ becomes a gradient $\nabla \mathcal{J}$ (derived in §5.3.1) in the parameter space and the process of maximising \mathcal{J} becomes an instance of a non-linear optimisation problem for a continuous function in §3.1.

Elements of the gradient are the derivative of the fidelity functional with respect to the control amplitudes $c_{k,n}$. The only dependence of the fidelity on the control amplitudes is in the propagator at that time slice, $\hat{\mathcal{P}}_n$ – calculation of the time propagator derivatives will be set out in §5.3.1.

Once the $K \times N$ propagator derivatives are calculated, they can be used to give the derivative of the fidelity functional with respect to the control amplitude. From the previous section §3.2.3, the fidelity at a time slice is the overlap, the inner product, of state of the system and its adjoint state at that time slice.

The state trajectory can be calculated from one forward-time propagation, and the adjoint trajectory from one backward-time propagation. The forward propagation from ρ_0 to ρ_n is

$$|\rho_n\rangle = \hat{\mathcal{P}}_n \cdots \hat{\mathcal{P}}_2 \hat{\mathcal{P}}_1 |\rho_0\rangle \quad (3.77)$$

and the backward propagation from σ to ρ_{n+1} is

$$|\chi_n\rangle = \hat{\mathcal{P}}_n^\dagger \cdots \hat{\mathcal{P}}_{N-1}^\dagger \hat{\mathcal{P}}_N^\dagger |\sigma\rangle \quad (3.78)$$

Using Eqs. (3.62), (3.75), (3.77), (3.78) in the piecewise constant approximation gives gradient elements of the fidelity functional, used for a gradient ascent method [154] or a quasi-Newton method [86]:

$$\nabla \mathcal{J}_0(c_{k,n}) = \frac{\partial \mathcal{J}_0}{\partial c_{k,n}} = \left\langle \chi_{n+1} \left| \frac{\partial \hat{\mathcal{P}}_n}{\partial c_{k,n}} \right| \rho_{n-1} \right\rangle \quad (3.79a)$$

$$\nabla \mathcal{J}_1(c_{k,n}) = \frac{\partial \mathcal{J}_1}{\partial c_{k,n}} = \text{Re} \left\langle \chi_{n+1} \left| \frac{\partial \hat{\mathcal{P}}_n}{\partial c_{k,n}} \right| \rho_{n-1} \right\rangle \quad (3.79b)$$

$$\nabla \mathcal{J}_2(c_{k,n}) = \frac{\partial \mathcal{J}_2}{\partial c_{k,n}} = 2 \text{Re} \left\langle \chi_{n+1} \left| \frac{\partial \hat{\mathcal{P}}_n}{\partial c_{k,n}} \right| \rho_{n-1} \right\rangle \left\langle \rho_{n-1} \left| \sigma \right. \right\rangle \quad (3.79c)$$

A particular strength of the GRAPE method is that the gradient of the fidelity functional has the same complexity as the fidelity functional [86, 94, 154, 287, 311] and reduces to finding the derivative of a propagator at each time slice and for each control. Calculation of these derivatives will be the focus of §5.

The elements of the gradient should be arranged as a vector of derivatives with respect to control amplitudes, $c_{k,n}$, for a control k at a time steps n :

$$\nabla \mathcal{J}(\mathbf{c}) = \begin{pmatrix} \nabla \mathcal{J}(c_{1,1}) & \nabla \mathcal{J}(c_{1,2}) & \cdots & \nabla \mathcal{J}(c_{1,N}) \\ \nabla \mathcal{J}(c_{2,1}) & \nabla \mathcal{J}(c_{2,2}) & \cdots & \nabla \mathcal{J}(c_{2,N}) \\ \vdots & \vdots & \ddots & \vdots \\ \nabla \mathcal{J}(c_{K,1}) & \nabla \mathcal{J}(c_{K,2}) & \cdots & \nabla \mathcal{J}(c_{K,N}) \end{pmatrix} \mapsto \begin{pmatrix} \nabla \mathcal{J}(c_{1,1}) \\ \nabla \mathcal{J}(c_{2,1}) \\ \vdots \\ \nabla \mathcal{J}(c_{K,1}) \\ \nabla \mathcal{J}(c_{1,2}) \\ \nabla \mathcal{J}(c_{2,2}) \\ \vdots \\ \nabla \mathcal{J}(c_{K,2}) \\ \vdots \\ \nabla \mathcal{J}(c_{1,N}) \\ \nabla \mathcal{J}(c_{2,N}) \\ \vdots \\ \nabla \mathcal{J}(c_{K,N}) \end{pmatrix} \quad (3.80)$$

and should be consistent with the arrangement of the control vector in Eq. (3.61).

During the forward-time propagation of the state ρ_0 with the propagators in Eq. (3.75), a trajectory of the system dynamics is produced. This can be stored in an array of state vectors, $(\rho_0 \ \rho_1 \ \dots \ \rho_N)$, and used for trajectory analysis of the controlled system dynamics [166].

§ 3.2.6 Algorithm

An effective optimisation should be preconditioned [229], which means that the optimisation variables should not be too small or too large. For the GRAPE method this means that the control amplitudes should be normalised, requiring a nominal power level to be separated from the control amplitudes

$$\mathbf{c} \rightarrow p_c \tilde{\mathbf{c}} \quad (3.81)$$

where p_c is a control power level

The GRAPE algorithm can be summarised in the following way:

- ① Define the system uncontrollable Hamiltonian $\hat{\mathcal{H}}_0$, relaxation $\hat{\mathcal{R}}$, and control operators $\hat{\mathcal{H}}_k$.
- ② Set an initial state $|\rho_0\rangle$, and target state $|\sigma\rangle$ of the system.
- ③ Start with an initial guess^{1} of control amplitudes, $\tilde{\mathbf{c}}$. and nominal power level p_c with $\max |\tilde{\mathbf{c}}| \approx +1$ and $|\min \tilde{\mathbf{c}}| \approx 0$.
- ④ Call GRAPE as the objective function and multiply the control amplitudes by the nominal power level.
- ⑤ Propagate the system forward in time from $|\rho_0\rangle$ with the time propagators in Eq. (3.75) and store the states at each time slice in Eq. (3.77)
- ⑥ Propagate the system backward in time from ρ_0 with the complex conjugate of the time propagators in Eq. (3.75) and store the states at each time slice in Eq. (3.78)

^{1}Can be a random guess

- ⑦ Calculate gradient elements with EQ. (3.79b) and the stored states from EQs. (3.77), (3.78).
- ⑧ Multiply the gradient vector with p_c and send back to the optimiser.
- ⑨ Update the control amplitudes with an update rule in EQ. (3.48).
- ⑩ Test termination conditions, and an optimal solution hasn't been found, return to STEP ④

§ 4

Hessian Calculations

“A man of genius makes no mistakes. His errors are volitional and are the portals of discovery.”

– James Joyce, ULYSSES

A method of optimal control, called GRAPE, was reviewed in §3.2 with application to magnetic resonance systems of quantum ensembles in §2. Applying rapidly converging numerical optimisation methods of §3.1 to optimal control algorithms will be the focus of this chapter. The Newton-Raphson version of GRAPE (Newton-GRAPE) has the potential to achieve quadratic convergence, compared to the superlinear convergence of state-of-the-art methods currently used in this area of optimal control, as outlined in §1.

Scientific instruments used in many applications of quantum theory have reached the limits of what is physically, legally or financially possible. Examples include power deposition safeguards in MRI instruments [49], sample heating thresholds in biomolecular NMR spectroscopy [110] and the steep dependence of the cost of superconducting magnets on the induction they generate [133]. Some limits, such as the length of time a patient can be persuaded to stay inside an MRI machine, are psychological [260], but in practice no less real. In these time-constrained situations faster convergence and greater code parallelisation, compared to other quantum control algorithms in the GRAPE family, have the prospect of giving financial and medical benefits.

This chapter demonstrates that the Hessian of the GRAPE fidelity functional is unusually cheap, having the same asymptotic complexity scaling as the functional itself. This leads to the possibility of using very efficient numerical optimisation

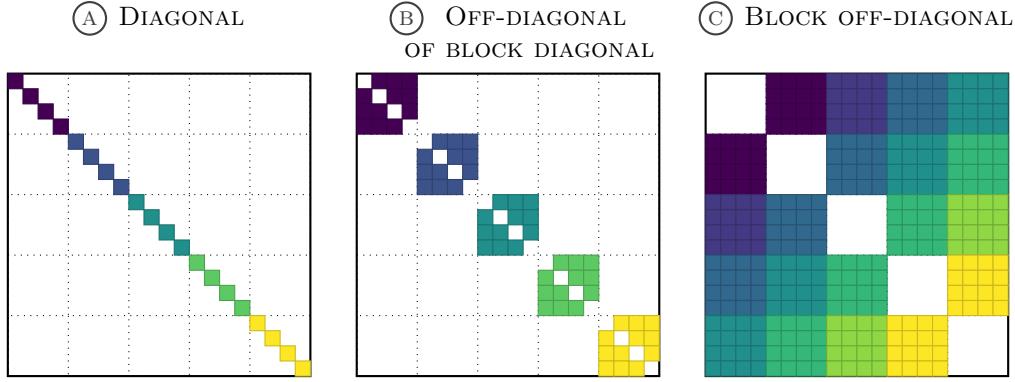


FIGURE 4.1: Diagram of the block structure of the Hessian matrix (here, with $K = 4$ and $N = 5$). (A) shows the diagonal proper, with $(K \times N)$ elements calculated with Eqs. (4.5a)–(4.5c). (B) shows the off-diagonal elements within the block diagonal, with $(K(K - 1) \times N)$ elements calculated with Eqs. (4.6a)–(4.6c). (C) shows the block off-diagonal elements of Eq. (4.4), with $(K^2 \times N(N - 1))$ elements calculated with Eqs. (4.7a)–(4.7c).

techniques. In particular, the Newton-Raphson method with a regularised and well-conditioned Hessian (§§4.2.1–4.2.3) is shown to require fewer system trajectory evaluations than any other algorithm in the GRAPE family. The methods presented in this chapter were published by Goodwin and Kuprov in [108] and are implemented in *Spinach* v2.0 and later^{11} and results are presented in §4.3.

§ 4.1 Fidelity Hessian

The Newton-Raphson method in §3.1.2 requires a Hessian matrix so that local curvature information can be used to improve a search direction compared to a gradient-based method. From Eq. (3.15), the Newton-step search direction of an iteration j leads the update of the control amplitudes to maximise a fidelity function

$$\mathbf{c}^{(j+1)} = \mathbf{c}^{(j)} + \varepsilon \left[\nabla^2 \mathcal{J}(\mathbf{c}^{(j)}) \right]^{-1} \nabla \mathcal{J}(\mathbf{c}^{(j)}) \quad (4.1)$$

where the step length is $\varepsilon = 1$ for a Newton-step, or with ε calculated with a line search method, such as those in §3.1.4.

^{11}Available at www.spindynamics.org.

§ 4.1.1 Hessian structure

The fidelity Hessian with more than one control is a tensor, and should be arranged as a block matrix of $N \times N$ blocks of $K \times K$ matrices, consistent with Eqs. (3.61), (3.80):

$$\nabla^2 \mathcal{J}(\mathbf{c}) \triangleq \begin{pmatrix} \mathbf{D}_1^2 & \mathbf{D}_{1,2} & \cdots & \mathbf{D}_{1,N} \\ \mathbf{D}_{2,1} & \mathbf{D}_2^2 & \cdots & \mathbf{D}_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{D}_{N,1} & \mathbf{D}_{N,2} & \cdots & \mathbf{D}_N^2 \end{pmatrix} \quad (4.2)$$

where the matrices $\mathbf{D}_{m,n}$ are each of size $K \times K$. The block diagonals, depicted in FIG. 4.1 (A)–(B), are

$$\mathbf{D}_n^2 \triangleq \begin{pmatrix} \frac{\partial^2 \mathcal{J}}{\partial c_{1,n}^2} & \frac{\partial^2 \mathcal{J}}{\partial c_{1,n} \partial c_{2,n}} & \cdots & \frac{\partial^2 \mathcal{J}}{\partial c_{1,n} \partial c_{K,n}} \\ \frac{\partial^2 \mathcal{J}}{\partial c_{2,n} \partial c_{1,n}} & \frac{\partial^2 \mathcal{J}}{\partial c_{2,n}^2} & \cdots & \frac{\partial^2 \mathcal{J}}{\partial c_{2,n} \partial c_{K,n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 \mathcal{J}}{\partial c_{K,n} \partial c_{1,n}} & \frac{\partial^2 \mathcal{J}}{\partial c_{K,n} \partial c_{2,n}} & \cdots & \frac{\partial^2 \mathcal{J}}{\partial c_{K,n}^2} \end{pmatrix} \quad (4.3)$$

and the block off-diagonals depicted in FIG. 4.1 (C) are

$$\mathbf{D}_{m,n} \triangleq \begin{pmatrix} \frac{\partial^2 \mathcal{J}}{\partial c_{1,m} \partial c_{1,n}} & \frac{\partial^2 \mathcal{J}}{\partial c_{1,m} \partial c_{2,n}} & \cdots & \frac{\partial^2 \mathcal{J}}{\partial c_{1,m} \partial c_{K,n}} \\ \frac{\partial^2 \mathcal{J}}{\partial c_{2,m} \partial c_{1,n}} & \frac{\partial^2 \mathcal{J}}{\partial c_{2,m} \partial c_{2,n}} & \cdots & \frac{\partial^2 \mathcal{J}}{\partial c_{2,m} \partial c_{K,n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 \mathcal{J}}{\partial c_{K,m} \partial c_{1,n}} & \frac{\partial^2 \mathcal{J}}{\partial c_{K,m} \partial c_{2,n}} & \cdots & \frac{\partial^2 \mathcal{J}}{\partial c_{K,m} \partial c_{K,n}} \end{pmatrix}, \quad n \neq m \quad (4.4)$$

§ 4.1.2 Hessian elements

The second derivatives of the fidelity [5] have a similar form to the gradient elements in Eqs. (3.79a)–(3.79c), defined for the fidelity functional in Eqs. (3.67)–(3.69). The second derivatives of the fidelity, with respect to a single time slice

and a single control, are the diagonal elements of Hessian in EQ. (4.3), depicted in FIG. 4.1 (Ⓐ), and their equations are:

$$\frac{\partial^2 \mathcal{J}_0}{\partial c_{k,n}^2} = \left\langle \chi_{n+1} \left| \frac{\partial^2 \hat{\mathcal{P}}_n}{\partial c_{k,n}^2} \right| \rho_{n-1} \right\rangle \quad (4.5a)$$

$$\frac{\partial^2 \mathcal{J}_1}{\partial c_{k,n}^2} = \text{Re} \left\langle \chi_{n+1} \left| \frac{\partial^2 \hat{\mathcal{P}}_n}{\partial c_{k,n}^2} \right| \rho_{n-1} \right\rangle \quad (4.5b)$$

$$\frac{\partial^2 \mathcal{J}_2}{\partial c_{k,n}^2} = 2\text{Re} \left\langle \chi_{n+1} \left| \frac{\partial^2 \hat{\mathcal{P}}_n}{\partial c_{k,n}^2} \right| \rho_{n-1} \right\rangle \quad (4.5c)$$

The off-diagonal elements of the block diagonal in EQ. (4.3), depicted in FIG. 4.1 (Ⓑ), are similarly defined:

$$\frac{\partial^2 \mathcal{J}_0}{\partial c_{k,n} \partial c_{j,n}} = \left\langle \chi_{n+1} \left| \frac{\partial^2 \hat{\mathcal{P}}_n}{\partial c_{k,n} \partial c_{j,n}} \right| \rho_{n-1} \right\rangle \quad (4.6a)$$

$$\frac{\partial^2 \mathcal{J}_1}{\partial c_{k,n} \partial c_{j,n}} = \text{Re} \left\langle \chi_{n+1} \left| \frac{\partial^2 \hat{\mathcal{P}}_n}{\partial c_{k,n} \partial c_{j,n}} \right| \rho_{n-1} \right\rangle \quad (4.6b)$$

$$\frac{\partial^2 \mathcal{J}_2}{\partial c_{k,n} \partial c_{j,n}} = 2\text{Re} \left\langle \chi_{n+1} \left| \frac{\partial^2 \hat{\mathcal{P}}_n}{\partial c_{k,n} \partial c_{j,n}} \right| \rho_{n-1} \right\rangle \quad (4.6c)$$

where $j \neq k$. The states $|\rho_{n-1}\rangle$ and $|\chi_{n+1}\rangle$ are the forward and backward propagated states respectively, from Eqs. (3.77), (3.78), and propagators are those in EQ. (3.75).

The block off-diagonal elements in EQ. (4.4) have the useful relation in that they require only first derivatives of the fidelity [5], which would have been calculated previously in evaluating the fidelity gradient in Eqs. (3.79a)–(3.79c). The equations of the block off-diagonal Hessian elements in EQ. (4.4), depicted in FIG. 4.1 (Ⓐ), are

$$\frac{\partial^2 \mathcal{J}_0}{\partial c_{k,n} \partial c_{j,m}} = \left\langle \chi_{n+1} \left| \frac{\partial \hat{\mathcal{P}}_n}{\partial c_{k,n}} \hat{\mathcal{P}}_{n-1} \cdots \hat{\mathcal{P}}_{m+1} \frac{\partial \hat{\mathcal{P}}_m}{\partial c_{j,m}} \right| \rho_{m-1} \right\rangle \quad (4.7a)$$

$$\frac{\partial^2 \mathcal{J}_1}{\partial c_{k,n} \partial c_{j,m}} = \text{Re} \left\langle \chi_{n+1} \left| \frac{\partial \hat{\mathcal{P}}_n}{\partial c_{k,n}} \hat{\mathcal{P}}_{n-1} \cdots \hat{\mathcal{P}}_{m+1} \frac{\partial \hat{\mathcal{P}}_m}{\partial c_{j,m}} \right| \rho_{m-1} \right\rangle \quad (4.7b)$$

$$\frac{\partial^2 \mathcal{J}_2}{\partial c_{k,n} \partial c_{j,m}} = 2\text{Re} \left\langle \chi_{n+1} \left| \frac{\partial \hat{\mathcal{P}}_n}{\partial c_{k,n}} \hat{\mathcal{P}}_{n-1} \cdots \hat{\mathcal{P}}_{m+1} \frac{\partial \hat{\mathcal{P}}_m}{\partial c_{j,m}} \right| \rho_{m-1} \right\rangle \quad (4.7c)$$

where $n > m^{\{\text{I}\}}$.

Eqs. (4.5a)–(4.5c), (4.6a)–(4.6c), (4.7a)–(4.7c) show the Hessian has almost the same numerical computational complexity scaling as the fidelity functional in Eqs. (3.67)–(3.69) [5], just as the gradient does in Eqs. (3.79a)–(3.79c) [154]. This situation is highly unusual in non-linear optimization theory – Hessians are normally so expensive that a significant body of work exists on the subject of avoiding their calculation and recovering second derivative information in an approximate way from the gradient history [34, 79, 194, 229]. The recent BFGS-GRAPE algorithm [86] is an example of such approach. The fact that the Hessian is relatively cheap suggests Newton-Raphson type algorithms [79, 229] with the control sequence update rule of Eq. (4.1).

However, three logistical problems present themselves that must be solved before the method becomes useful in practice:

- ① Efficient calculation of second derivatives of the propagators in Eq. (3.75).
This problem will be considered in detail in §5.
- ② Propagator derivative recycling between gradient elements in Eqs. (3.79a)–(3.79c) and block off-diagonal Hessian elements in Eqs. (4.7a)–(4.7c), required for efficient scaling. Outlined in [107], the latency of the cache storage device means that for small matrices it may be faster to recalculate the function. A caching procedure becomes beneficial for larger matrices.
- ③ Regularisation and conditioning of the Hessian matrix in Eq. (4.1) is needed to avoid the numerical difficulties associated with its inverse – a Hessian matrix should be positive definite, from Eq. (3.9), and well-conditioned [79, 229]. This will be the main focus of the remainder of this chapter.

§ 4.1.3 Control operator commutativity

The full Hessian requires $(K \times N)$ second-order propagator derivative evaluations of Eqs. (4.5a)–(4.5c) and FIG. 4.1 ①. It was pointed out that the block off-diagonal second-order propagator derivatives of Eqs. (4.7a)–(4.7c) and FIG. 4.1

^①If $n < m$ then n and m are interchanged in Eqs. (4.7a)–(4.7c). If $n = m + 1$, there are no propagators in between the two first-order propagator derivatives. If $n = m + 2$, there is one propagator $\hat{\mathcal{P}}_{m+1} = \hat{\mathcal{P}}_{n-1}$ in between the two first-order propagator derivatives.

© are essentially free, having already been calculated with a gradient evaluation. There is a further useful property of the Hessian matrix: that it should be real and symmetric. Potentially, this can halve the number of calculations of the type in Eqs. (4.6a)–(4.6c).

However, it is noted in [195, 211] that the additive law fails unless there is commutativity:

$$e^{t\mathbf{A}} e^{t\mathbf{B}} = e^{t(\mathbf{A}+\mathbf{B})}, \quad \text{iff} \quad \mathbf{AB} = \mathbf{BA} \quad (4.8)$$

$$\Rightarrow \frac{\partial^2 \mathcal{J}}{\partial c_{k,n} \partial c_{j,n}} = \frac{\partial^2 \mathcal{J}}{\partial c_{j,n} \partial c_{k,n}}, \quad \text{iff} \quad [\hat{\mathcal{H}}_k, \hat{\mathcal{H}}_j] = 0 \quad (4.9)$$

The relation is only valid if the associated control operators commute with each other. To ensure the Hessian matrix is symmetric^{I} when condition of EQ. (4.9) is not met, the following formula must be used

$$\left. \begin{aligned} \nabla^2 \mathcal{J}(c_{j,n}, c_{k,n}) \\ \nabla^2 \mathcal{J}(c_{k,n}, c_{j,n}) \end{aligned} \right\} = \frac{1}{2} \left(\frac{\partial^2 \mathcal{J}}{\partial c_{k,n} \partial c_{j,n}} + \frac{\partial^2 \mathcal{J}}{\partial c_{j,n} \partial c_{k,n}} \right), \quad \text{if} \quad [\hat{\mathcal{H}}_k, \hat{\mathcal{H}}_j] \neq 0 \quad (4.10)$$

which requires two second-order propagator derivative evaluations.

§ 4.2 Hessian Regularisation

Maximisation with Newton-Raphson and quasi-Newton methods relies on the necessary conditions for the truncated Taylor series approximation to be valid. This was introduced using a local quadratic approximation of the objective function in EQ. (3.13), restated here for the fidelity functional:

$$\Delta \mathcal{J} = \mathcal{J}(\mathbf{c}^{(j+1)}) - \mathcal{J}(\mathbf{c}^{(j)}) \approx \left\langle \nabla \mathcal{J}(\mathbf{c}^{(j)}) \middle| \mathbf{c}^{(j)} \right\rangle + \frac{1}{2} \left\langle \mathbf{c}^{(j)} \middle| \nabla^2 \mathcal{J}(\mathbf{c}^{(j)}) \middle| \mathbf{c}^{(j)} \right\rangle \quad (4.11)$$

In addition to the gradient being zero in the vicinity of a stationary point $\mathbf{c}^{(*)}$,

$$\lim_{\mathbf{c} \rightarrow \mathbf{c}^{(*)}} \left\| \nabla \mathcal{J}(\mathbf{c}^{(*)}) \right\|_{\infty} = 0 \quad (4.12)$$

^{I}When a Hessian is not symmetric, its eigenvalues are not real.

the second-order sufficient condition of EQ. (3.9) must also be true in the vicinity of a stationary point:

$$\lim_{\mathbf{c} \rightarrow \mathbf{c}^{(*)}} \left\langle \mathbf{c} \left| \nabla^2 \mathcal{J}(\mathbf{c}^{(*)}) \right| \mathbf{c} \right\rangle > 0, \quad \forall \mathbf{c} \in \mathbb{R}^{(K \cdot N)} \quad (4.13)$$

i.e. the Hessian must be positive definite near a stationary point $\mathbf{c}^{(*)}$. This is evident from EQ. (4.1), in which a negative definite Hessian would result in a step being performed down, rather than up, the corresponding gradient direction. Furthermore, an indefinite Hessian is an indication of a saddle point, shown in FIG. 3.1 (C), which should be avoided during maximisation of the fidelity functional. Forcing a Hessian to be a definite matrix ensures that steps are made that avoid saddle points. Forcing a positive definite Hessian, as opposed to a negative definite Hessian, ensures that a step is always made in the correct ascent/descent direction.

A significant problem is that, far away from a stationary point, the Hessian is not expected to be positive definite [85, 237–239, 251, 296]. Small Hessian eigenvalues are also problematic because they result in overly long steps that can be detrimental because most fidelity functionals are not actually quadratic. A significant amount of research has gone into modifying the Hessian in such a way as to avoid these undesired behaviours [14–16, 40, 101, 102, 112, 121, 214, 215, 281].

§ 4.2.1 Cholesky factorisation

One fairly cheap way to work around an indefinite Hessian is to attempt *Cholesky factorisation* [96], which exists for any invertible positive definite matrix [106]:

$$\left[\nabla^2 \mathcal{J} \right] = \mathbf{L} \mathbf{L}^T \quad (4.14)$$

$$\Rightarrow \left[\nabla^2 \mathcal{J} \right]^{-1} = \left[\mathbf{L}^{-1} \right]^T \mathbf{L}^{-1} \quad (4.15)$$

where \mathbf{L} is a lower triangular matrix with positive diagonal elements and \mathbf{L}^T is its transpose^{11}. If a Cholesky factorisation finds solution, the Hessian is positive definite. If a Cholesky factorisation fails, the Hessian is indefinite, and an identity matrix may be used as a substitute for the Hessian, effectively reverting to a gradient ascent step for any iterations that produce an indefinite Hessian [105].

The problem with this approach is that indefinite Hessians become more common as the dimension of the problem increases, making the maximiser spend most of the time in the gradient ascent mode and destroying any advantage of the second-order method over simple gradient ascent – this technique is not recommended.

However, there is a further use of the Cholesky factorisation test for an indefinite Hessian: an algorithm can be designed to iteratively test the Hessian, modifying the Hessian at each stage. Using the Cholesky decomposition on a Hessian [96] with a multiple of the unit matrix added [102, 121, 214, 215]:

$$\left[\nabla^2 \mathcal{J} \right] + \sigma \mathbf{1} \mathbf{1}^T = \mathbf{L} \mathbf{L}^T, \quad \sigma \geq 0 \quad (4.16)$$

The choice of σ is made to produce a positive definite Hessian satisfying Eq. (4.13), with the trial value

$$\sigma = \begin{cases} \left\| \nabla^2 \mathcal{J} \right\|_F - \min \left[\nabla^2 \mathcal{J} \right]_{ii} & \min \left[\nabla^2 \mathcal{J} \right]_{ii} < 0 \\ \left\| \nabla^2 \mathcal{J} \right\|_F & \min \left[\nabla^2 \mathcal{J} \right]_{ii} \geq 0 \end{cases} \quad (4.17)$$

chosen in that way because the Frobenius norm of the Hessian is an upper bound on the largest absolute eigenvalue. The value of σ is increased iteratively until the Cholesky decomposition succeeds [229] and the inverse Hessian may be obtained.

A benefits of this iterative Cholesky factorisation method are that an inverse Hessian is automatically produced from Eq. (4.15), which is a simple task to compute for invertible triangular matrices [106], and that the method avoids the expensive

^{11}The argument of the Hessian has been dropped here, and for the rest of the chapter. It should be assumed that within equations and definitions, the Hessian, gradient, and controls are all of the same iteration, j , unless stated otherwise:

$$\begin{aligned} \nabla^2 \mathcal{J}(\mathbf{c}^{(j)}) &\rightarrow \nabla^2 \mathcal{J} \\ \nabla \mathcal{J}(\mathbf{c}^{(j)}) &\rightarrow \nabla \mathcal{J} \\ \mathbf{c}^{(j)} &\rightarrow \mathbf{c} \end{aligned}$$

operation of calculating an eigendecomposition, which is time-consuming for large Hessian matrices.

§ 4.2.2 Trust Region Method

A more sophisticated workaround is to use the eigenvalue shifting method, also called the trust region method, adapted from the Levenberg-Marquardt algorithm [95, 204]. Hessian eigenvalues may be computed explicitly [112]

$$\left[\nabla^2 \mathcal{J} \right] = \mathbf{Q} \boldsymbol{\Lambda} \mathbf{Q}^{-1} \quad (4.18)$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix containing the eigenvalues of $\nabla^2 \mathcal{J}$ and \mathbf{Q} is the matrix with columns of corresponding eigenvectors. A precise estimate can now be made for the largest negative eigenvalue of the Hessian, and added to the diagonal matrix of eigenvalues to reform the Hessian from this modified eigendecomposition:

$$\left[\nabla^2 \mathcal{J} \right]_{\text{reg}} = \mathbf{Q} (\boldsymbol{\Lambda} + \sigma \mathbf{1}) \mathbf{Q}^{-1}, \quad \sigma = \max (0, \delta - \lambda_{\min}) \quad (4.19)$$

where λ_{\min} is the minimum eigenvalue of the Hessian, $\min(\boldsymbol{\Lambda}_{ii})$. The arbitrary positive value of δ is included to make the Hessian positive definite. The primary problem with this method is that, for poorly conditioned or near-singular Hessian matrices, the regularisation procedure destroys much of the curvature information and the technique effectively becomes a combination of the Newton-Raphson and gradient ascent methods. Formally, Eqs. (4.18), (4.19) do solve the step direction problem, but practical testing indicates the convergence rate can fail to achieve true quadratic convergence near a maximum.

§ 4.2.3 Rational Function Optimisation

A refined regularisation method, called rational function optimization [15, 16, 281], does allow a modified Hessian to be a well-conditioned, positive definite matrix. It replaces the truncated Taylor series in Eq. (4.11) with a [2/1] Padé approximant [127, 234], defined in §5.1.2, approximating as a rational function:

$$\Delta \mathcal{J} = \frac{\langle \nabla \mathcal{J} | \mathbf{c} \rangle + \frac{1}{2} \langle \mathbf{c} | \nabla^2 \mathcal{J} | \mathbf{c} \rangle}{1 + \langle \mathbf{c} | \mathbf{S} | \mathbf{c} \rangle} \quad (4.20)$$

This preserves the derivative information of EQ. (4.11), leaving the necessary conditions unchanged, because the derivatives of $[1 + \langle \mathbf{c} | \mathbf{S} | \mathbf{c} \rangle]^{-1}$ give contributions only through higher orders in \mathbf{c} . The nature of the rational function means that the asymptotes of $\Delta\mathcal{J}$ and its gradients remain finite for $\mathbf{c} \rightarrow \pm\infty$, determined by the Hessian and the symmetric scaling matrix \mathbf{S} . The first-order necessary condition for EQ. (4.12) gives the following eigenvalue equation:

$$\begin{pmatrix} [\nabla^2\mathcal{J}] & |\nabla\mathcal{J}\rangle \\ \langle\nabla\mathcal{J}| & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{c} \\ \mathbf{1} \end{pmatrix} = 2\Delta\mathcal{J} \begin{pmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{pmatrix} \begin{pmatrix} \mathbf{c} \\ \mathbf{1} \end{pmatrix} \quad (4.21)$$

Choosing a uniform scaling matrix $\mathbf{S} = \alpha^{-2}\mathbf{1}$ [15], where $0 < \alpha \leq 1$, reduces this equation to

$$\begin{pmatrix} \alpha^2[\nabla^2\mathcal{J}] & \alpha|\nabla\mathcal{J}\rangle \\ \alpha\langle\nabla\mathcal{J}| & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{c}/\alpha \\ \mathbf{1} \end{pmatrix} = 2\Delta\mathcal{J} \begin{pmatrix} \mathbf{c}/\alpha \\ \mathbf{1} \end{pmatrix} \quad (4.22)$$

Rational function optimisation proceeds in a similar way to eigenvalue shifting methods described above, except the shifting is applied to the augmented Hessian matrix. Defining the eigendecomposition of the augmented Hessian matrix as

$$[\nabla^2\mathcal{J}]^{\text{aug}} = \begin{pmatrix} \alpha^2[\nabla^2\mathcal{J}] & \alpha|\nabla\mathcal{J}\rangle \\ \alpha\langle\nabla\mathcal{J}| & \mathbf{0} \end{pmatrix} = \mathbf{Q}\Lambda\mathbf{Q}^{-1} \quad (4.23)$$

a similar shifting formula to EQ. (4.19) can be used to ensure a positive definite augmented Hessian matrix

$$[\nabla^2\mathcal{J}]_{\text{reg}}^{\text{aug}} = \frac{1}{\alpha^2}\mathbf{Q}(\Lambda + \sigma\mathbf{1})\mathbf{Q}^{-1}, \quad \sigma = \max(0, -\lambda_{\min}) \quad (4.24)$$

The top left corner block of the regularised augmented Hessian is then used for the Newton-Raphson step [15, 16, 281] in EQ. (4.1).

§ 4.2.4 Hessian Conditioning

Motivated by the *Wolfe conditions* of Eqs. (3.51), (3.53), (3.54) [333, 334], it may be necessary to place restrictions on the search direction by excluding directions near-orthogonal to the steepest ascent vector. This condition, called the *angle criterion*, states that for any angle $\theta^{(j)}$ between the search direction, \mathbf{d}_s , and the gradient vector should be bounded away from orthogonality. Using the derivation

set out by Fletcher [79], the critical angle is should be

$$\theta^{(j)} \leq \frac{\pi}{2} - \mu \quad \forall j \quad (4.25)$$

with $\mu > 0$ and independent of j , and the angle $\theta^{(j)}$ is calculated as

$$\cos(\theta^{(j)}) = \frac{\langle \nabla \mathcal{J}(\mathbf{c}^{(j)}) | \mathbf{c}^{(j)} \rangle}{\|\nabla \mathcal{J}(\mathbf{c}^{(j)})\|_2 \cdot \|\mathbf{c}^{(j)}\|_2} \quad (4.26)$$

For Newton-type methods with a continuous gradient, using inexact line searches of Eqs. (3.51), (3.53), (3.54), maximisation is bounded from above. This leads to the conclusion that, as the change in fidelity tends to zero, $\Delta \mathcal{J} \rightarrow 0$, then so does the slope $\langle \nabla \mathcal{J} | \mathbf{c} \rangle \rightarrow 0$. Two cases remain: either $\nabla \mathcal{J}^{(j)} \rightarrow 0$ or $\mathcal{J}^{(j)} \rightarrow \infty$ [79].

The condition number defined here as the ratio of its largest Hessian eigenvalue, $\max(\mathbf{\Lambda}_{ii})$, to the smallest Hessian eigenvalue, $\min(\mathbf{\Lambda}_{ii})$:

$$\zeta = \frac{\lambda_{\max}}{\lambda_{\min}} \quad (4.27)$$

The sufficient condition for ascent is that the Hessian is positive definite. This gives a sufficient condition for the angle criterion of Eqs. (4.25), (4.26), requiring the condition number of the Hessian matrix also to be bounded above. With the inequalities

$$\|\nabla^2 \mathcal{J} | \nabla \mathcal{J} \rangle\|_2 \leq \lambda_{\max} \quad (4.28)$$

$$\langle \nabla \mathcal{J} | \nabla^2 \mathcal{J} | \nabla \mathcal{J} \rangle \geq \lambda_{\min} \langle \nabla \mathcal{J} | \nabla \mathcal{J} \rangle \quad (4.29)$$

the sufficient condition for the angle criterion is

$$\theta^{(j)} \leq \frac{\pi}{2} - \frac{1}{\zeta^{(j)}} \quad (4.30)$$

With this condition satisfied, the Armijo condition in Eq. (3.51) is also satisfied. The methods that follow will ensure that the condition number of the Hessian matrix is bounded so the above conditions are met.

The condition number of the Hessian should be bound; one way to achieve this is to allow the scaling constant α in Eqs. (4.23), (4.24) to vary according to the condition of the Hessian. The value of α is reduced until the condition number becomes acceptable (nearer to $\alpha = 1$, when the Hessian is well-conditioned), for

example:

$$\alpha_{r+1} = \phi\alpha_r \quad \text{while} \quad \zeta > \frac{1}{\sqrt[3]{\text{EPS}}} \quad (4.31)$$

where EPS is machine precision and $\alpha_0 = 1$. The cube-root of machine precision reflects the allowable condition number for an acceptable cubic interpolation within the sectioning phase of a line search in §3.1.4. The factor $0 < \phi < 1$ is used to iteratively decrease the condition number of the Hessian – this is the method used to condition the Hessian in the examples presented below. It should be noted that a value $\phi > 1$ may be used to increase the condition number of the Hessian when it is very small and the inequality of EQ. (4.31) is reversed.

Quantum mechanics is rich in situations that produce very small values of Hessian and gradient elements. Choosing α_0 to be the value that would shift the smallest eigenvalues to be above 1, giving a similar effect to choosing $\delta = 1$ in EQ. (4.19), accounts for optimisation problems that are inherently ill-conditioned in this way. When the choice

$$\alpha_0 = \frac{1}{\sqrt[3]{|\lambda_{\min}|}} \quad (4.32)$$

is made, the augmented Hessian in EQs. (4.23), (4.24) becomes

$$\begin{bmatrix} \nabla^2 \mathcal{J} \end{bmatrix}^{\text{aug}} = \begin{pmatrix} \frac{1}{\lambda_{\min}} \begin{bmatrix} \nabla^2 \mathcal{J} \end{bmatrix} & \frac{1}{\sqrt{\lambda_{\min}}} \langle \nabla \mathcal{J} \rangle \\ \frac{1}{\sqrt{\lambda_{\min}}} \langle \nabla \mathcal{J} \rangle & \mathbf{0} \end{pmatrix} = \mathbf{Q} \boldsymbol{\Lambda} \mathbf{Q}^{-1} \quad (4.33)$$

$$\begin{bmatrix} \nabla^2 \mathcal{J} \end{bmatrix}_{\text{reg}}^{\text{aug}} = \lambda_{\min} \mathbf{Q} (\boldsymbol{\Lambda} + \sigma \mathbf{1}) \mathbf{Q}^{-1}, \quad \sigma = \max(0, -\lambda_{\min}) \quad (4.34)$$

Practical testing indicates that this combination of using initial scaling, then accepting large condition numbers for the Hessian, allows the Newton-Raphson method to avoid getting stuck at inflection points. The condition number would grow to a large value around the inflection point and then shrink back when the point has been avoided. Due to the tendency of the Hessian to increase condition number as it approaches the maximiser, machine precision could eventually become a limit to this type of conditioning. To avoid a slowdown at the final stages of the optimisation, an upper bound is placed on the α parameter in EQ. (4.31) – this guarantees that the terminal convergence is always quadratic.

In practice, at each optimisation step the function code attempts to compute the Cholesky decomposition of EQ. (4.14). If that is successful then no regularisation is needed, otherwise the function proceeds to regularise with the methods described

above. Once the ascent direction and the initial step length are obtained from Eq. (4.1) it is advantageous to perform a line search procedure.

§ 4.3 Convergence analysis

In comparing the performance of this Newton-Raphson optimal control, the following optimisation problems will be used^{I}:

- State-to-state transfer over a scalar coupled hydrofluorocarbon fragment spin system, as described in [108].
- State-to-state transfer in a quadrupolar ^{14}N spin, transferring the population between the $T_{1,0}$ and $T_{2,2}$ states of Eqs. (2.66), (2.76).
- Phase-modulated universal rotation pulses with a sine function amplitude profile, and relaxation. The system is a chain of 101 non-interacting ^{13}C spins spread over 50 kHz.

For fair comparison, all optimisations are run a number of times from different initial guesses^{II}, allowing identification of the optimisation convergence characteristics.

§ 4.3.1 Scalar coupled three-spin state-to-state population transfer

This system consists of a hydrofluorocarbon fragment, as described in [108], with spins ^1H – ^{13}C – ^{19}F coupled in a linear chain, in a 9.4 tesla magnetic field. The scalar coupling, J-coupling, is 140 Hz between ^1H – ^{13}C and –160 Hz between ^{13}C – ^{19}F . Using Eq. (2.89), this gives the drift Hamiltonian of the system as

$$\begin{aligned}\hat{\mathcal{H}}_0 &= 280\pi(\hat{\vec{L}}^{(\text{H})} \cdot \hat{\vec{L}}^{(\text{C})}) - 320\pi(\hat{\vec{L}}^{(\text{C})} \cdot \hat{\vec{L}}^{(\text{F})}) \\ &= 280\pi(\hat{L}_z^{(\text{H})}\hat{L}_z^{(\text{C})}) - 320\pi(\hat{L}_z^{(\text{C})}\hat{L}_z^{(\text{F})})\end{aligned}\quad (4.35)$$

to be used in Eq. (3.57).

^{I}The templates for these optimal control simulations are included in *Spinach*–v2.0, downloadable from www.spindynamics.org

^{II}Random numbers were seeded with `rng(32)` at the start of each convergence analysis, and the same starting guesses were used for different methods of optimisation.

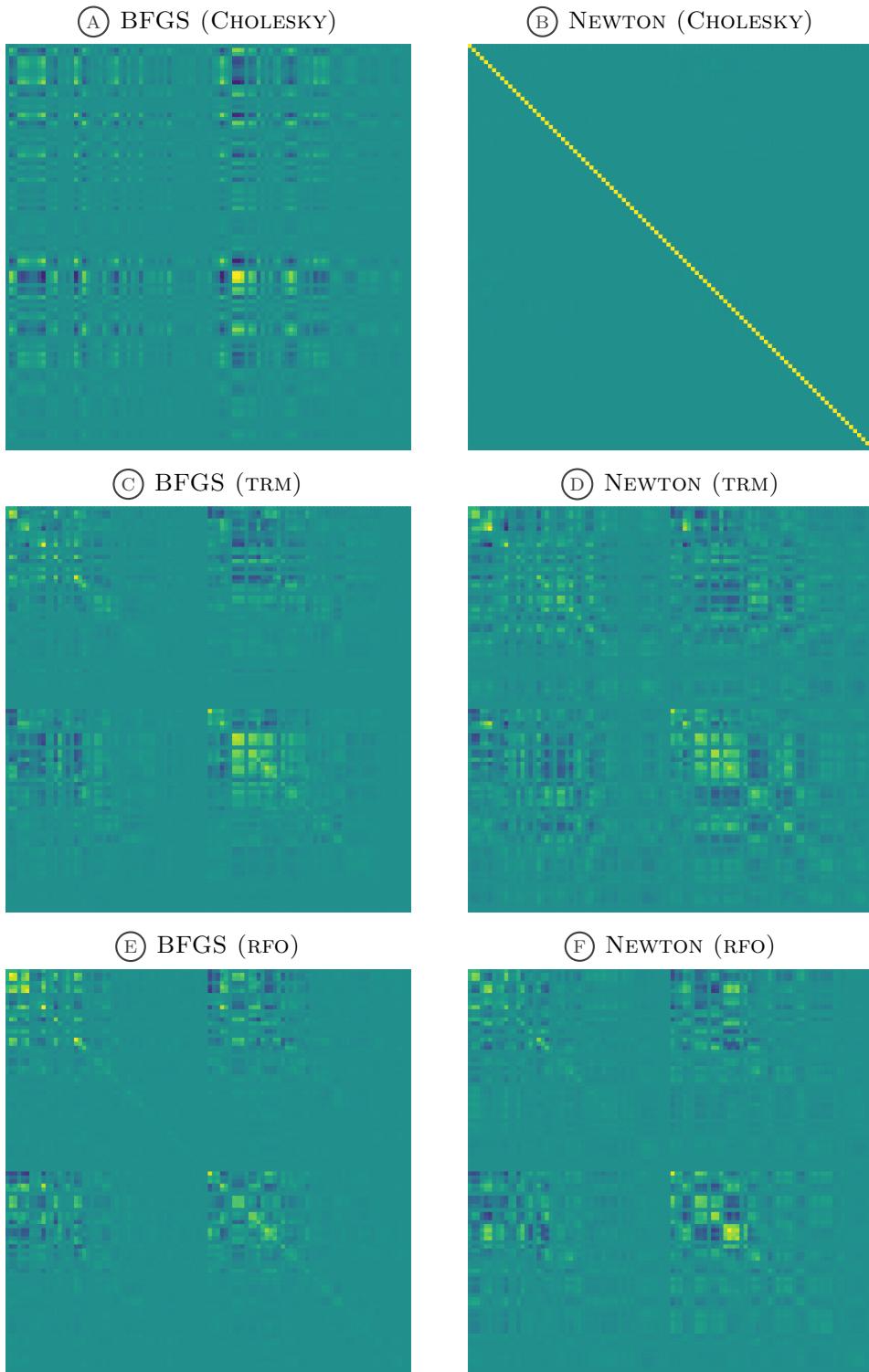


FIGURE 4.2: Converged Hessian matrices, arranged $N \times K$ (opposite to FIG. 4.1 to clearly see the time evolution of control channels, $\hat{\mathcal{H}}_x^{(H)}$ and $\hat{\mathcal{H}}_y^{(H)}$, produced by one of the random seeds in FIGS. 4.6, 4.7. (B) Newton method almost give a multiple of $\mathbb{1}$, (A) has structure – indicating an indefinite matrix from the Newton method and a positive definite matrix from the Quasi-Newton method (the BFGS method has a single iteration of the Cholesky regularisation).

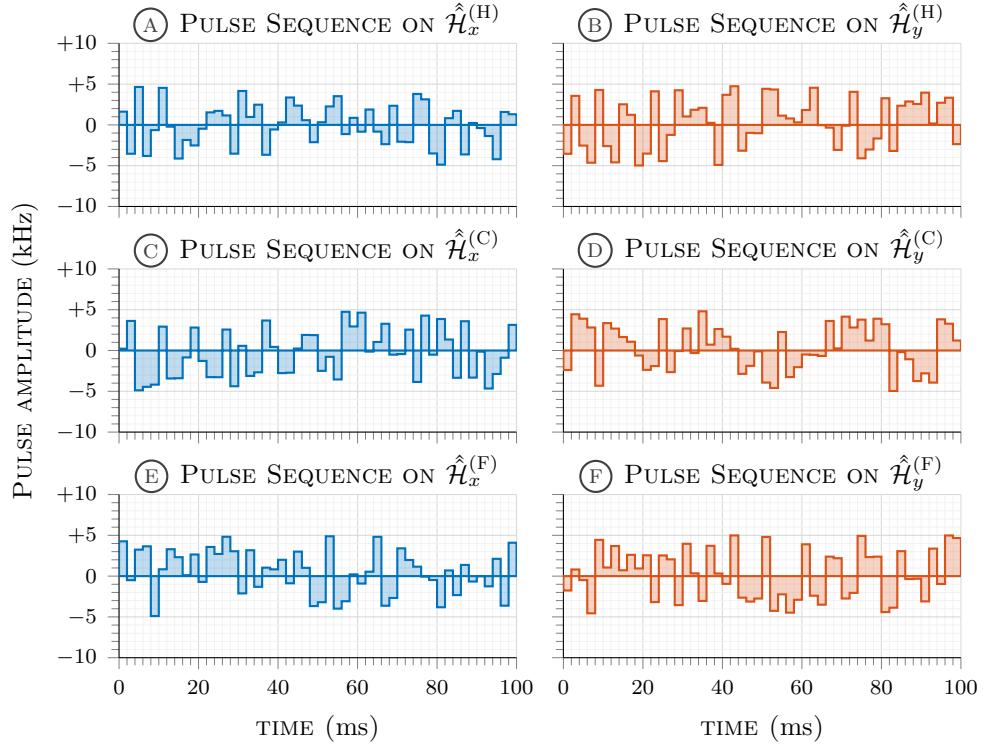


FIGURE 4.3: The six channel control pulse sequences producing the trajectories in FIG. 4.4. The pulses are a result of the Newton-GRAPE optimisation, which reached maximum fidelity, for a state-to-state transfer in a scalar coupled hydrofluorocarbon fragment, transferring the population between the $\hat{L}_z^{(H)}$ and $\hat{L}_z^{(F)}$ states

The initial state is magnetisation population in the $\hat{L}_z^{(H)}$ state in Liouville space, and the desired target is the $\hat{L}_z^{(F)}$ state, with both states normalised.

The total control pulse duration is 100 ms and the sequence is discretised to 50 equal time slices. The six control operators are $\hat{H}_x^{(H)}$, $\hat{H}_y^{(H)}$, $\hat{H}_x^{(C)}$, $\hat{H}_y^{(C)}$, $\hat{H}_x^{(F)}$, and $\hat{H}_y^{(F)}$. Each of the control power levels allowed is simulated with a nominal power level of 10 kHz. The maximum fidelity achievable for this system is $\mathcal{J}_1 = 1$. The results of one of these, fully converged, optimisations is depicted in FIG. 4.4, showing the evolution of the correlation orders and polarisation at each spin over the total pulse time. For completeness, the six control pulse sequences are shown in FIG. 4.3.

Three regularisation methods are compared for Newton-GRAPE and BFGS-GRAPE:

- ① Comparison of the BFGS and inverse-BFGS method.

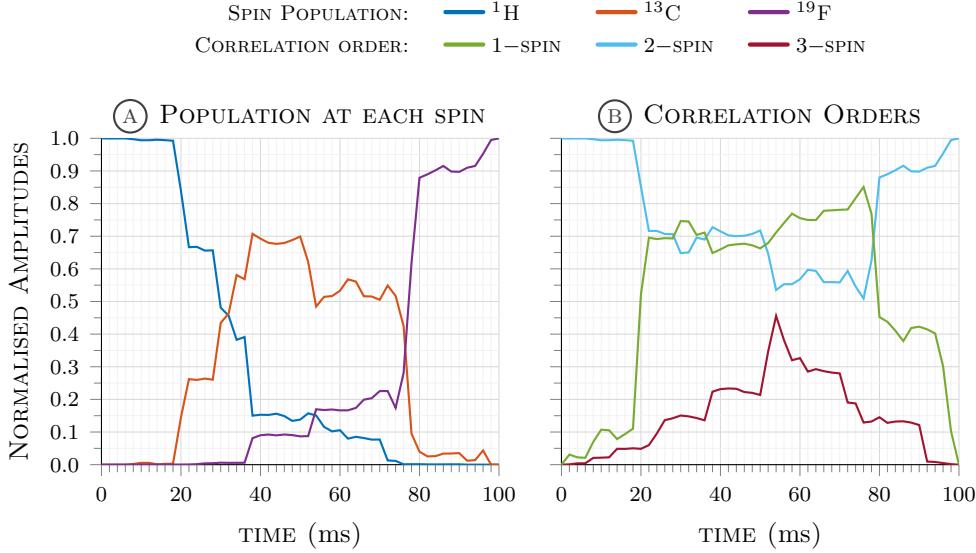


FIGURE 4.4: Trajectories analysis for a state-to-state transfer, in a scalar coupled hydrofluorocarbon fragment, transferring the population between the $\hat{L}_z^{(\text{H})}$ and $\hat{L}_z^{(\text{F})}$ states. (A) Local spin population. (B) Spin correlation order amplitude. The pulse sequence producing these trajectories has 6 control channels and is shown in FIG. 4.3.

- ② Newton-GRAPE and BFGS-GRAPE methods, using iterative Cholesky factorisation as a regularisation method. At least one iteration of the Cholesky factorisation was forced for the BFGS method.
- ③ Newton-GRAPE and BFGS-GRAPE methods, using TRM as a regularisation method. The minimum eigenvalue of the Hessian is shifted with $\delta = 1$ in Eq. (4.19). At least one iteration of TRM was forced for the BFGS method.
- ④ Newton-GRAPE and BFGS-GRAPE methods, using RFO as a regularisation and conditioning method. The iterative conditioning bounds the Hessian to $\zeta < 1 \times 10^4$ with $\phi = 0.9$ in Eq. (4.31). At least one iteration of RFO was forced for the BFGS method.

Newton-Raphson methods that need Hessian regularisation need a framework of comparison to current leading [86] quasi-Newton methods. However, the ℓ -BFGS method does not store an explicit Hessian matrix, only calculating the resulting search direction vector. For the use of testing regularisation methods, standard BFGS methods will be used – which should give very similar convergence since the ℓ -BFGS and BFGS methods are both rank-2 updates [229].

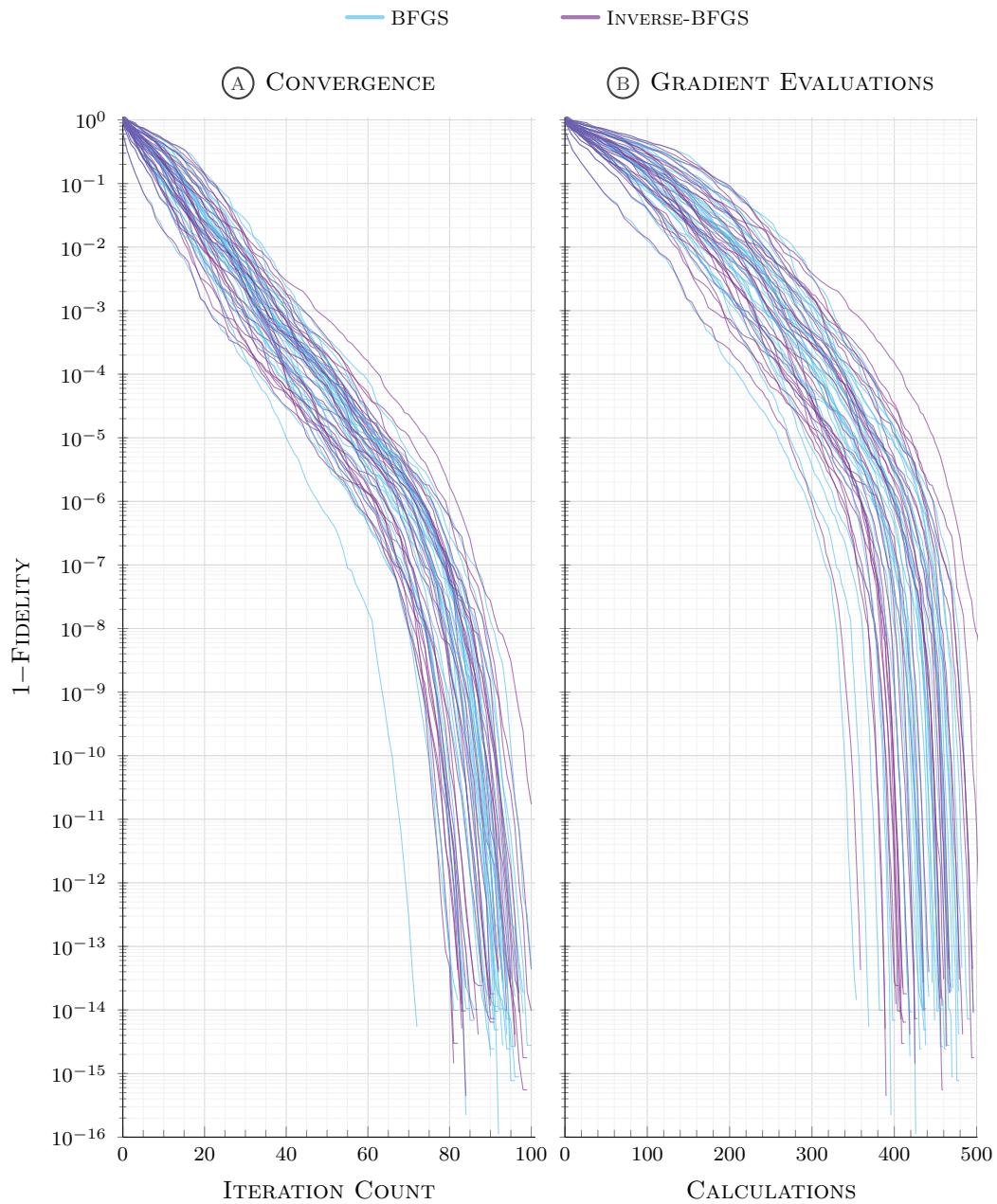


FIGURE 4.5: Comparison of BFGS updating the inverse Hessian, and BFGS updating the Hessian approximation. Simulation was run multiple times, from different initial guesses of control pulses. (A) Showing iterate count. (B) Showing gradient calculation count.

The DFP and BFGS methods perform a rank-2 update, commonly on the inverse Hessian using Eqs. (3.36), (3.37), and with a defined positive definite inverse Hessian approximation. However, regularisation methods of Eqs. (4.19), (4.23), (4.24) may have other beneficial effects on the Hessian matrix; making the Hessian sufficiently positive definite and well-conditioned. A fair comparison of regularised Newton-Raphson methods should be with quasi-Newton methods which update the Hessian proper with Eq. (3.37) with the same, forced, regularisation and conditioning as the Newton-Raphson Hessian. Concentrating on the arguably superior rank-2 update of the BFGS method, FIG. 4.5 shows a comparison between the two BFGS update formulas in Eqs. (3.37), (3.38).

The plots of FIG. 4.5 clearly show there is no appreciable difference of performance or computational cost in the BFGS Hessian update and the BFGS inverse-Hessian update methods. This justifies the use of a BFGS Hessian update of Eq. (3.37) as a benchmark of comparison using regularised and conditioned Hessian matrices. FIG. 4.5 should be used as a benchmark for the current best GRAPE method with ℓ -BFGS optimisation [86].

Correcting an indefinite Hessian with Cholesky factorisation

Initial testing of the Newton-Raphson method failed to proceed further than a single iterate, failing to find a search direction based from an indefinite Hessian matrix. Further to this, forcing a step along the direction of the Newton-step shows the direction is not an ascent direction and an optimisation cannot proceed based on violation of the Wolfe conditions.

An iterative Cholesky factorisation, of Eq. (4.14), can be used to correct the non-ascent direction from an indefinite Hessian – forcing the positive definite property, allowing a Newton-Raphson method to proceed. Results of this optimisation are shown compared with the equivalent BFGS method, where at least one Cholesky factorisation is forced, in FIG. 4.6. Both methods use a bracketing and sectioning line search with cubic interpolation.

The convergence characteristics in FIG. 4.6 are not promising: linear convergence at best. They show that an iterative Cholesky factorisation does allow the Newton-Raphson method to proceed, finding an ascent direction, correcting an indefinite Hessian. This also shows that regularisation is needed for the Newton-GRAPE method.

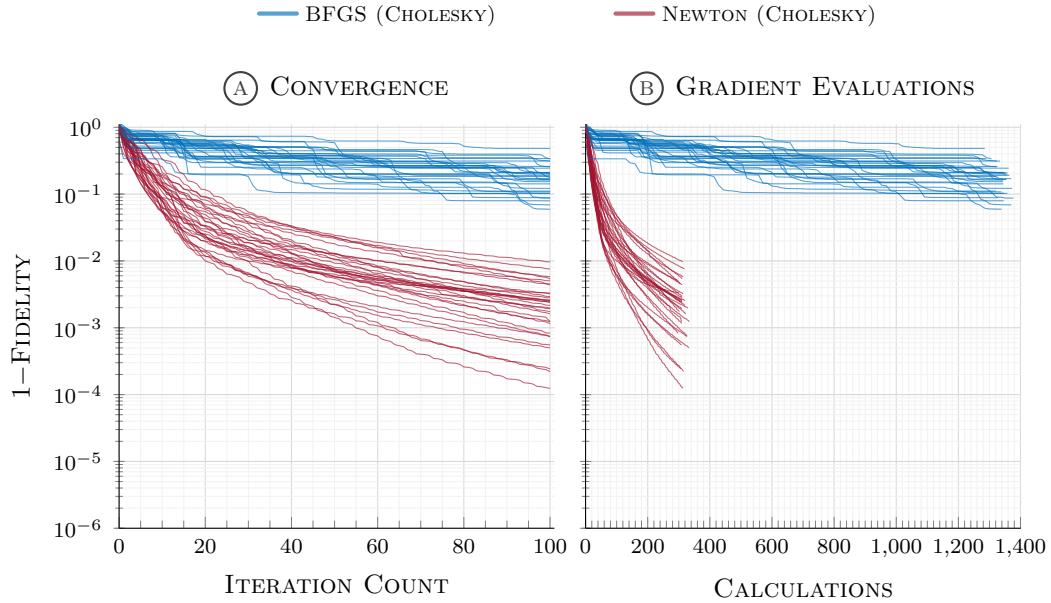


FIGURE 4.6: Comparison of optimisation methods for state-to-state transfer in a scalar coupled hydrofluorocarbon fragment, transferring the population between the $\hat{L}_z^{(H)}$ and $\hat{L}_z^{(F)}$ states (pulse and trajectory shown in FIG. 4.4): Newton-GRAPE and BFGS-GRAPE methods are regularised with a Cholesky factorisation (forced in the case of BFGS-GRAPE). Both methods use a bracketing and sectioning line search. (A) Comparing the iterations count. (B) Comparing the gradient/Hessian calculation count.

However, the convergence for the Newton-Raphson method is not quadratic, reverting to the linear convergence of a gradient ascent method. Furthermore, there is a heavy load on the line search, which is indicated by the large amount of gradient calculations in FIG. 4.6 (B).

Comparison of the Newton method to the BFGS method, both with a Cholesky factorisation show that this method of regularisation quickly destroys any useful information from the Hessian and the Hessian approximation – both methods effectively rely on the line search to make progress, with the second order information only showing a vague ascent direction.

The iterative Cholesky factorisation reduces the Newton and BFGS methods to a gradient ascent method, which is clearly not adequate for a Newton-Raphson method in GRAPE.

Although a Cholesky factorisation is not ideal for regularising a Hessian, it is a cheap way to recognise an indefinite matrix. Within a finished Newton-Raphson method product, the check of its positive definite property should be done with an

attempted Cholesky factorisation: if it succeeds then no regularisation is required, if not then the Newton-Raphson method should proceed to a more sophisticated regularisation method.

Correcting an ill-conditioned Hessian with RFO & TRM

As indicated in §4.2.4, if the condition number of the Hessian is not bounded, a search direction can get too close to the vector orthogonal to the gradient. One method that can control the condition number is the TRM of Eq. (4.19). Here, a multiple of the identity matrix is added to the Hessian eigendecomposition, shifting all eigenvalues to a defined level above zero. More specifically, making a shift to $\lambda_{\min} = 1$ ensures that small eigenvalues do not inflate the condition number.

Results in FIG. 4.7 indicate, in addition to correcting for a non-ascent direction and allowing a Newton-Raphson method to proceed, that a quadratic convergence phase does occur in the vicinity of the maximiser. Comparison of a TRM conditioned BFGS method in FIG. 4.7 to that without TRM in FIG. 4.5 show a slight improvement in convergence, with 70% reduction in optimisation iterations and a large saving in computational cost of 65% reduction in gradient calculations. This indicates that a better search direction is produced with TRM than without it.

The conditioned Newton-Raphson method cannot be compared fairly with an unconditioned Newton-Raphson method, and it must be inferred that TRM has a similar effect in finding a good search direction. When comparing the Newton-TRM with BFGS-TRM in FIG. 4.7, the benefits of calculating an explicit Hessian are evident, showing its expected quadratic convergence: Newton-TRM approaches its quadratic phase in about half the number of iterations and at about half the calculation cost of BFGS-TRM. However, there is a difference in the character of the quadratic phase of the two methods, the Newton-Raphson method follows a much steeper quadratic convergence, and the BFGS method follows a quadratic convergence with a similar profile to that of BFGS without forced regularisation and conditioning in FIG. 4.5.

The iteratively conditioned RFO method of Eqs. (4.33), (4.34) including scaling of the Hessian with its corresponding gradient, is the final method compared in FIG. 4.7. The Hessian condition number is bound below 10^4 (with $\phi = 0.9$ in

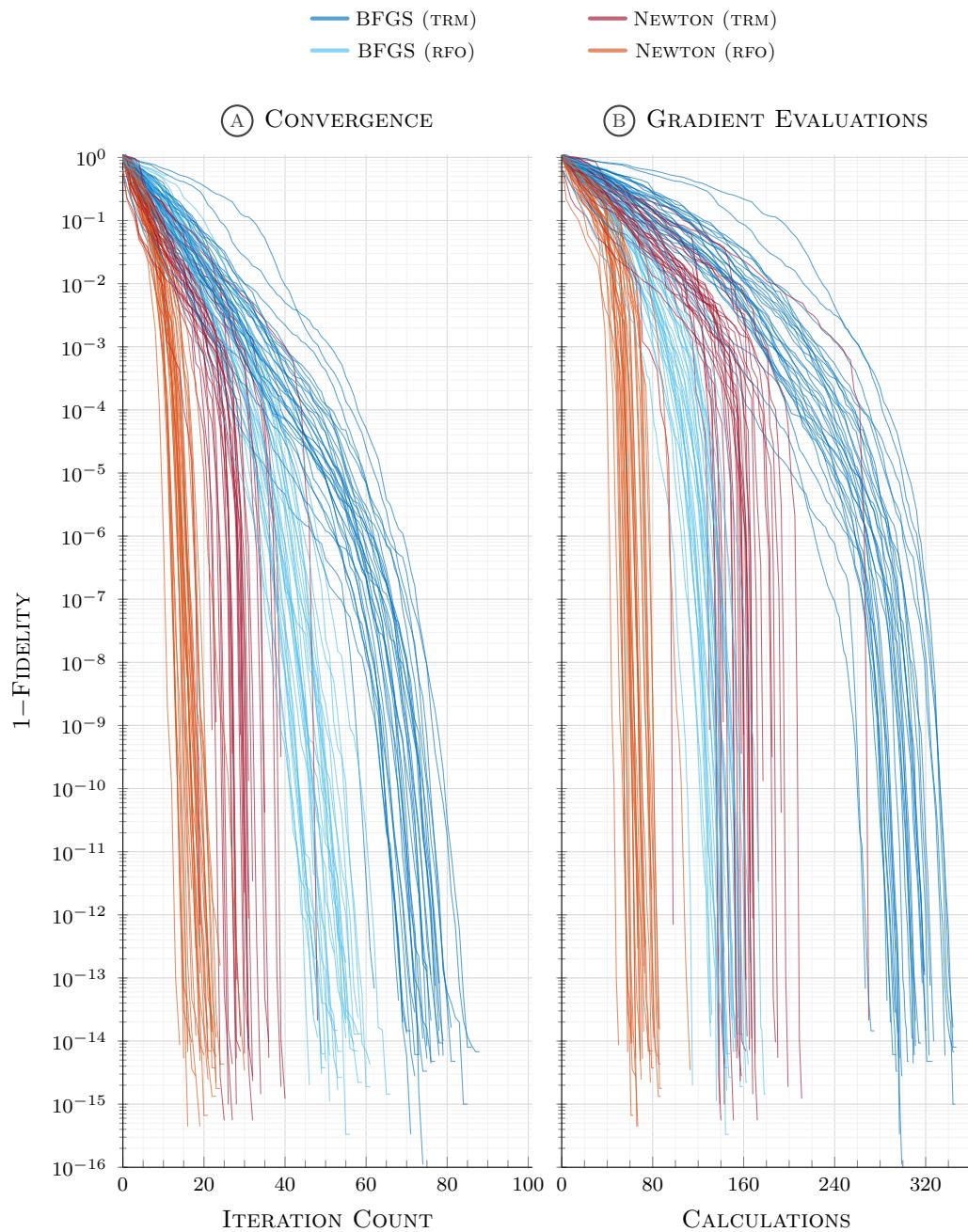


FIGURE 4.7: Comparison of optimisation methods for state-to-state transfer in a scalar coupled hydrofluorocarbon fragment, transferring the population between the $\hat{L}_z^{(H)}$ and $\hat{L}_z^{(F)}$ states (pulse and trajectory shown in FIG. 4.4): All methods use a bracketing and sectioning line search. (A) Comparing the iterations count. (B) Comparing the gradient/Hessian calculation count.

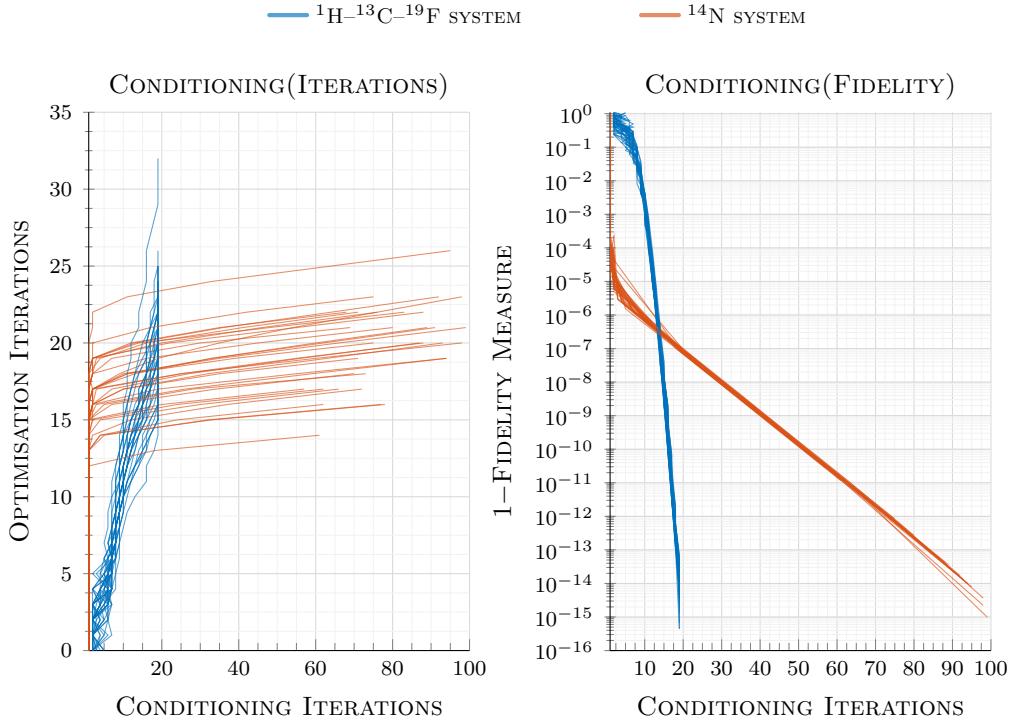


FIGURE 4.8: Hessian conditioning iterations for Newton-Raphson with RFO. Showing how the number of conditioning iterations increases for both optimisation problems of FIG. 4.11 and FIG. 4.7; (A) as the optimisation iterations increase, and (B) as the fidelity decreases.

EQ. (4.31))^{I}. The need to increase the number of conditioning iterations as the maximum is approached can clearly be seen in FIG. 4.8, indicating that the Hessian becomes more ill-conditioned the closer it gets to a maximum. The RFO methods both outperform their TRM counterparts in convergence and computational cost. It should also be noted that the BFGS RFO has less computational cost compared to the Newton-Raphson TRM. This is because the self-scaling provided by the RFO greatly reduces the strain on the line search algorithm. Essentially, RFO scales the Hessian and gradient, which is usually performed within the line search giving a smaller step size for a correspondingly large condition number of the Hessian.

The Newton-Raphson method with RFO regularisation and iterative conditioning has 85% reduction in the computational cost of the current best BFGS method in [86] (when comparing to FIG. 4.5), and has a predictable convergence (evident from the small variance from the average convergence trajectories), with a reduction of 80% in optimisation iterations, compared to the BFGS method. Further

^{I}Testing find a condition number bound of $\zeta < 10^4$ particularly effective. The theoretical bound of $\zeta < \sqrt[3]{\text{EPS}} \approx 1.6 \times 10^5$ performs slightly worse in comparison, reaching quadratic convergence a couple of iterations later than with $\zeta < 10^4$.

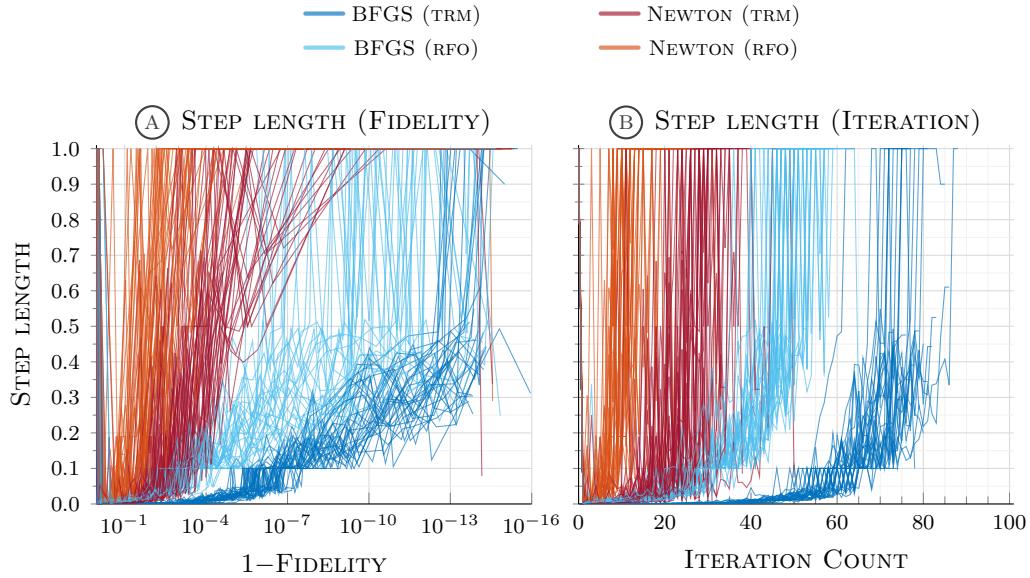


FIGURE 4.9: Step lengths for optimisations is FIG. 4.7, (A) as a function of $1 - \mathcal{J}_1$, and (B) as a function of iteration count.

information on the Hessian structure is depicted in FIG. 4.2, indicating RFO regularisation preserves more of the Hessian structure compared to other regularisation methods.

Looking deeper into the line search, step lengths at each iterate show how the Newton-Raphson method with RFO has the best convergence characteristics of regularisation methods tested. Step lengths at each iterate for those simulations in FIG. 4.7 are shown in FIG. 4.9. BFGS and inverse-BFGS methods without regularisation have the same spread of step-lengths at each iterate, just as they have the same convergence characteristics in FIG. 4.5 (not shown here). The general trend in all step length calculations of FIG. 4.9 is that they start small and converge to $\varepsilon = 1$ near the maximiser.

Quadratic convergence is reached very quickly in 10 iterations for the Newton-GRAPE method with RFO. This is evident from FIG. 4.9 (B), where the step length $\varepsilon = 1$ and remains there until a optimum is reached in (to the accuracy of EPS)^{I}.

BFGS methods with regularisation have a similar shape in step length evolution in FIG. 4.9 to BFGS method without regularisation. This improvement is seen at the start of the step-length evolution, approximately the first ten iterations,

^{I}The only time there is true quadratic convergence is when the step length has converged to 1 and remains 1 for the rest of the simulation. This is characteristic of the Newton-Raphson method and is a good indication that Hessian calculations and regularisation calculations have no errors - and the method is indeed a Newton-Raphson method proper.

where step-lengths are larger than those of the BFGS methods without regularisation. The regularised BFGS methods then follow a similar shape to non-regularised methods, having a similar rate of change per iterate. The conclusion here is that regularisation does indeed improve the BFGS method, and this is through better initial step lengths calculations.

It should be noted that when a Hessian calculation is made, there is no need for a separate gradient calculation (see §5). The only gradient calculations needed for the Newton-GRAPE method are those during the line search, where no Hessian is calculated. There is redundancy when the Newton-Raphson method reaches quadratic convergence, when the step length is the Newton-step; a gradient calculation must be made at the start of a line search to check that no line search is needed. This accounts for the larger than expected calculation count in FIG. 4.7 (B). If this redundancy can be removed, with a method that can switch the line search off when it detects quadratic convergence – the gradient count for the Newton-GRAPE method with RFO would have between 10-20 fewer gradient calculations^{I}.

§ 4.3.2 Quadrupolar spin state-to-state population transfer

This system consists of a single spin – 1 ^{14}N in a 14.1 tesla magnetic field, with coupling tensor of EQ. (2.88) in Hz

$$\mathbf{V} = 2\pi \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & -3 \end{pmatrix} \cdot 10^4 \quad (4.36)$$

specifying the quadrupolar interaction, giving the drift Hamiltonian of the system as

$$\hat{\mathcal{H}}_0 = \hat{\mathbf{L}} \cdot \mathbf{V} \cdot \hat{\mathbf{L}} \quad (4.37)$$

to be used in EQ. (3.57).

The initial state is the irreducible spherical tensor operator $\hat{T}_{1,0}$ state of EQ. (2.66) in Liouville space, and the desired target state is the $\hat{T}_{2,2}$ state of EQ. (2.76), with both states normalised.

^{I}There is no such switching method for the BFGS method because the Hessian update occurs after the line search, whereas the Newton-Raphson method calculates a Hessian before a line search.

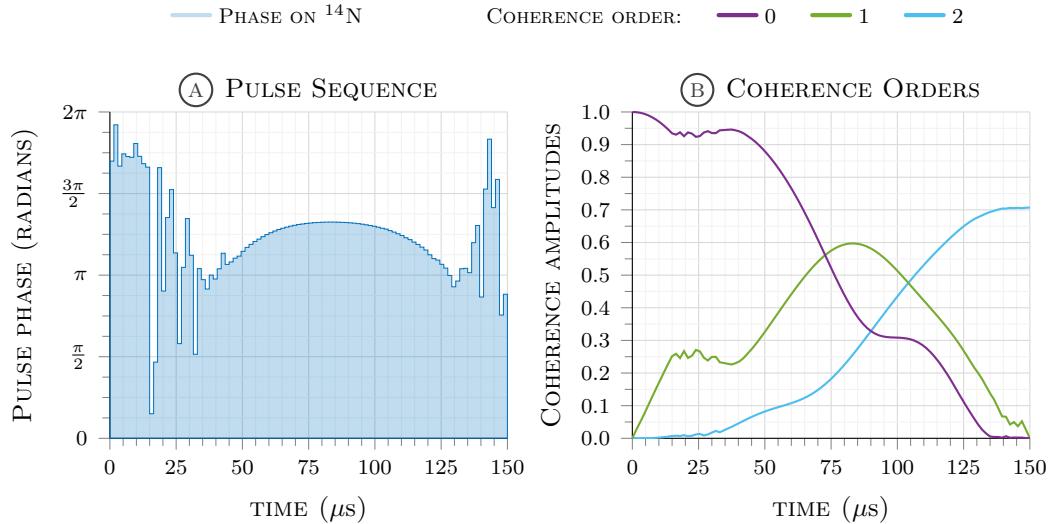


FIGURE 4.10: Results of a Newton-Raphson optimisation for a state-to-state transfer in a quadrupolar ^{14}N spin, transferring the population between the $T_{1,0}$ and $T_{2,2}$ states (§A). The convergence characteristics for this optimisation FIG. 4.11. (A) The phase modulated pulse sequence at the end of the optimisation. The amplitude profile is kept at a constant level of $4\sqrt{2} \times 10^4$ Hz. (B) Trajectory of the coherence order evolution during the control pulse sequence.

The total control pulse duration is $150 \mu\text{s}$ and the sequence is discretised to 100 equal time slices. Control operators are $\hat{\mathcal{H}}_x^{(N)}$ and $\hat{\mathcal{H}}_y^{(N)}$ operators, although the optimisation variables are the pulse sequence phase. The transform for this is

$$\mathbf{c}_A = \sqrt{\mathbf{c}_x^2 + \mathbf{c}_y^2} \quad \mathbf{c}_\varphi = \text{atan2}(\mathbf{c}_y, \mathbf{c}_x) \quad (4.38)$$

During the optimisation, the control phase, \mathbf{c}_φ is modified and the control amplitude, \mathbf{c}_A is kept at the constant level of $\sqrt{2} \times 40$ kHz.

The results of one of these, fully converged, optimisations is depicted in FIG. 4.10, showing the final phase modulated pulse sequence and the evolution of the coherence order over the total pulse time. The final fidelity reaches the maximum bound of $\mathcal{J}_1 = \frac{1}{\sqrt{2}}$ after $150 \mu\text{s}$ of pulsing.

Three optimisation methods were compared:

- ① Newton-GRAPE method using its Newton-step i.e. without a line search. Hessian regularisation is performed with RFO, and iterative conditioning bounds the Hessian to $\zeta < 1 \times 10^4$ with $\phi = 0.9$ in EQ. (4.31).

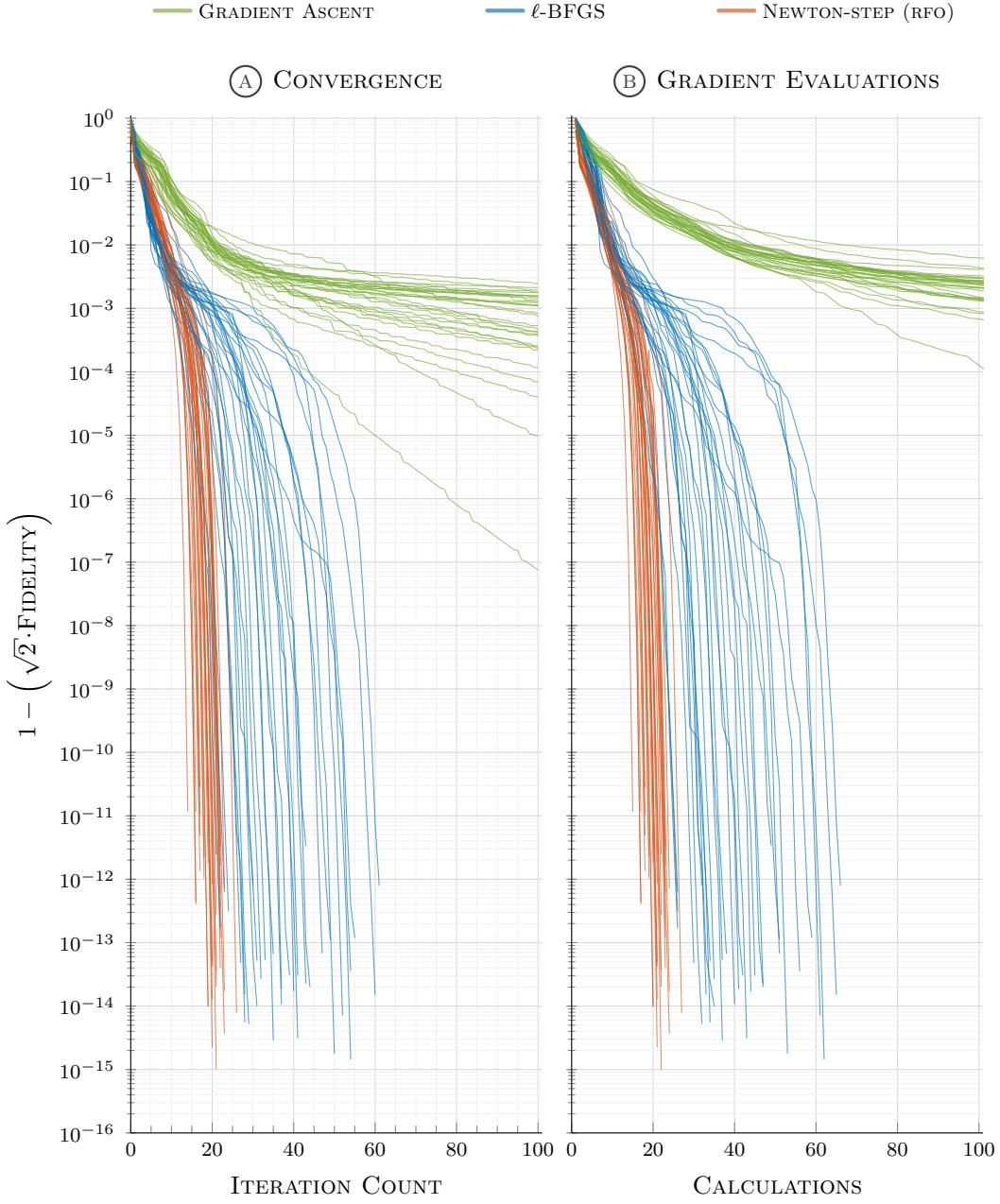


FIGURE 4.11: Comparison of optimisation methods for state-to-state transfer in a quadrupolar ^{14}N spin, transferring the population between the $T_{1,0}$ and $T_{2,2}$ states (pulse and trajectory shown in FIG. 4.10): Newton-GRAPE method using its Newton-step (i.e. without a line search); ℓ -BFGS-GRAPE using a bracketing and sectioning line search; the gradient ascent method, also using a bracketing and sectioning line search. (A) Comparing the iterations count. (B) Comparing the gradient/Hessian calculation count.

- ② ℓ -BFGS-GRAPE using a bracketing and sectioning line search, with cubic interpolation during the sectioning phase.
- ③ GRAPE method (gradient ascent) using a bracketing and sectioning line search, with cubic interpolation during the sectioning phase.

FIG. 4.11 compares the convergence of the three optimisation methods, plotting $1 - \sqrt{2} \cdot \mathcal{J}_1$ on a log scale to clearly see the convergence to the maximum fidelity value of $\mathcal{J}_1 = \frac{1}{\sqrt{2}}$.

The Newton-GRAPE performs well for this optimisation – without the need for a line search, the method reaches quadratic convergence after a small number of iterations and full quadratic convergence occurs in a small range of iterations, 15 – 20. Admittedly, this is an easy optimisation with only a single spin, and the ℓ -BFGS-GRAPE method also performs well considering it only uses gradient calculations. Furthermore, the ℓ -BFGS-GRAPE method rarely makes use of its available line search, with its rank-2 update giving a Newton-like step length at most iterations. The rank-2 update uses a gradient history to give its search direction, and it should be expected that the algorithm gets better as it has a longer history of gradients – after around 20 iterations in FIG. 4.11 ①.

The benefit of using a Newton-step for the Newton-GRAPE method, needing far fewer gradient evaluations, becomes a small benefit considering that the ℓ -BFGS-GRAPE method rarely uses the available line search. Furthermore, running the same Newton optimisations with any line search method (not shown here) results in the same – the method rarely uses a step length other than 1. The effect is to double the number of gradient evaluations without any increase in the convergence rate.

There are a few important uses of this test system. The first is that it shows the calculated Hessian is the correct one, removing any doubt that it contains numerical errors. The second is that it is a useful test to show how the condition number inflates compared to the optimisation of the previous section §4.3.1. FIG. 4.8 compares the number of conditioning iterations to those of §4.3.1 – showing a very different profile of how the condition number increases as the optimisation progresses. The HCF system gradually increases its condition number, starting from the first iteration, but the ^{14}N system does not need any conditioning until it reaches a certain fidelity, at which point the condition number increases exponentially with the fidelity increase.

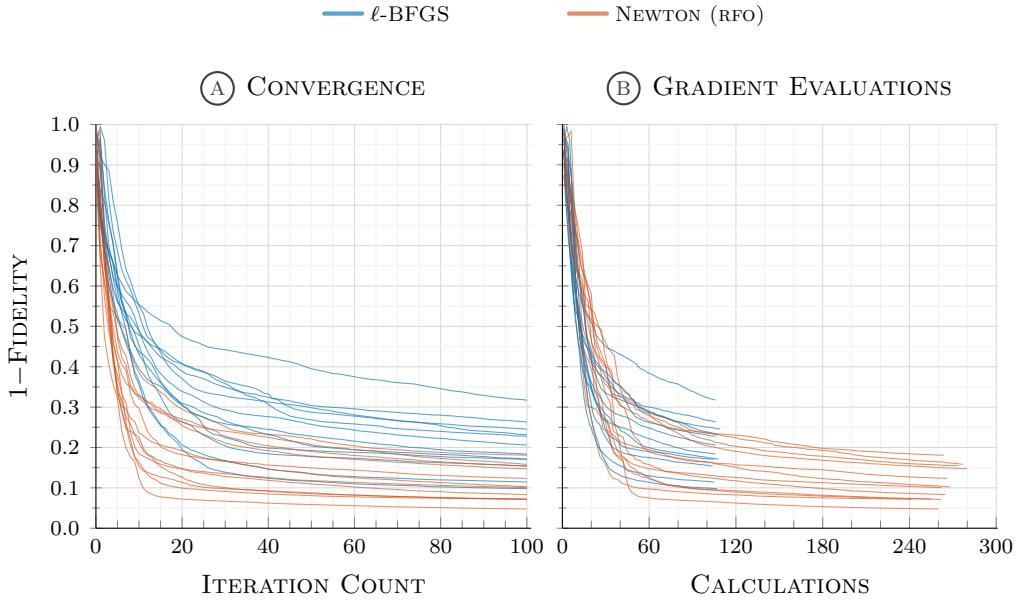


FIGURE 4.12: Comparison of optimisation methods for a universal rotation in 3D, on a 101 non-interacting spin – 1 ^{13}C spins, spread over 50 kHz. The Hamiltonian includes relaxation with the T_1/T_2 approximation. Two methods are compared: Newton-GRAPE method using a bracketing and sectioning line search; ℓ -BFGS-GRAPE using a bracketing and sectioning line search. (A) Comparing the iterations count. (B) Comparing the gradient/Hessian calculation count.

§ 4.3.3 Universal rotation

This system consists of a 101 non-interacting spin – 1 ^{13}C spins, spread over 50 kHz, in a 14.1 tesla magnetic field. Relaxation is included in the form of the T_1/T_2 approximation, keeping only the diagonal terms of the relaxation superoperator. Longitudinal states, corresponding to \hat{L}_z , relax with a rate of 10 Hz, and the transverse states, corresponding to \hat{L}_\pm , relax at a rate of 20 Hz.

The initial and final states of the system are set as a universal rotation in 3D, mentioned in §3.2.4:

$$\begin{aligned} \left| \rho_1(0) \right\rangle &= +\left| L_z \right\rangle \quad \longrightarrow \quad -\left| L_x \right\rangle = \left| \sigma_1 \right\rangle \\ \left| \rho_2(0) \right\rangle &= +\left| L_x \right\rangle \quad \longrightarrow \quad -\left| L_z \right\rangle = \left| \sigma_2 \right\rangle \\ \left| \rho_3(0) \right\rangle &= +\left| L_y \right\rangle \quad \longrightarrow \quad -\left| L_y \right\rangle = \left| \sigma_3 \right\rangle \end{aligned}$$

The total control pulse duration is 5 ms and the sequence is discretised to 400 equal time slices. Control operators are $\hat{\mathcal{H}}_x^{(\text{C})}$ and $\hat{\mathcal{H}}_y^{(\text{C})}$ operators, although the

optimisation variables are the pulse sequence phase in Eq. (4.38). During the optimisation, the control phase, \mathbf{c}_φ is modified and the control amplitude, \mathbf{c}_A is kept constant as a shaped amplitude profile: $\frac{1}{2}(1 + \sin(\theta))$, $\forall \theta \in [0, \pi]$. Pulses are designed to be robust to B_1 miscalibration by simulating over a range of power levels between 3.75 kHz and 6.25 kHz. This is achieved by simulating 5 optimisations over the given range, then averaging their fidelity, gradient and Hessian to be fed back to the optimisation algorithm.

As can be seen in FIG. 4.12, quadratic convergence is not reached. This may be attributed to a solution not existing for this universal rotation, or that the Newton-GRAPE method may not perform well when finding robust solutions. Evidence leans to the latter, where the ℓ -BFGS-GRAPE reaches a similar convergence after an initial phase of fast convergence. Finding a solution robust over the given range, in addition to finding a universal rotation pulse that works at each power level in the range, seem too much for either algorithm to handle effectively. However, the Newton-GRAPE method did show a marginally quicker initial convergence, although the number of gradient/Hessian calculations was similar. The tests on this type of optimisation problem indicate further work is required to find a more effective algorithm.

§ 5

Propagator Directional Derivatives

“Learning does not make one learned: there are those who have knowledge and those who have understanding. The first requires memory and the second philosophy.”

– Alexandre Dumas, THE COUNT OF MONTE CRISTO

In a general context, magnetic resonance simulation frequently requires chained exponential integrals of the form

$$\int_0^t dt_1 \int_0^{t_1} dt_2 \cdots \int_0^{t_{r-2}} dt_{r-1} \left\{ e^{(t-t_1)\mathbf{A}_1} \mathbf{B}_1 e^{(t_1-t_2)\mathbf{A}_2} \mathbf{B}_2 \cdots \mathbf{B}_{r-1} e^{t_r \mathbf{A}_r} \right\} \quad (5.1)$$

where \mathbf{A}_r and \mathbf{B}_r are square matrices. Examples include perturbative relaxation theories [103, 164, 250], reaction yield expressions in radical pair dynamics [115, 140, 306], average Hamiltonian theory [116], and pulsed field gradient propagators in nuclear magnetic resonance [67]. Their common feature is the complexity of evaluation: expensive matrix factorisations are usually required [1, 73, 103].

The exponentiation of a matrix [172] is far from a trivial operation. The operation was recognised as an important tool for solving differential equations [88] – optimal control methods rely heavily on it. The literature review in §1 identified the bottleneck in computation during the GRAPE method as calculation of a matrix exponential [10, 86, 111, 278, 311]. An outline of the GRAPE method was presented in §3, with the stages of forward/backward propagation Eqs. (3.75), (3.77), (3.78), and calculation of propagator derivatives Eqs. (3.75), (3.79a)–(3.79c), requiring

exponentiation of a matrix. The second-order derivatives of propagators used to construct a Hessian matrix in §4 also require the exponentiation of a matrix Eqs. (4.5a)–(4.5c), (4.6a)–(4.6c).

This chapter will introduce the matrix exponential in §5.1 and outline some of the “*nineteen dubious way to compute the exponential of a matrix*” to compute this, from the well cited work of Cleve Moler^{11} and Charles Van Loan [211, 212]. Much of the work used in this chapter can be found in textbooks on computations of matrix functions [106, 127].

The derivatives required for GRAPE are directional derivatives of the matrix exponential and these are defined in §§5.2.1, 5.3.2. With specific application to optimal control [86, 154, 311], chained exponential integrals will be used to find to analytic derivatives required by GRAPE, with a novel technique presented §5.2.2 and algorithmic subtleties presented in §5.4 with a measure of how parallel the algorithm is in §5.5.3 using propagator recycling in §5.5.

§ 5.1 The Matrix Exponential

The matrix exponential operation is not a trivial one [195] and approximations must be used to numerically compute the matrix exponential within Matlab [3, 126]. Much of the material in this section is taken from derivations in [106, 127]. An indispensable resource, by Moler and van Loan [211, 212] outlines an extensive review, showing “*nineteen dubious way to compute the exponential of a matrix*”.

§ 5.1.1 Taylor series approximation

The Taylor series [302] has already been used to form the quadratic model used as the basis of Newton-type methods of numerical optimisation in Eq. (3.3). The Taylor series is also the first call in approximating the function of matrix, $f(\mathbf{M})$,

^{11}It is no coincidence that Cleve Moler was the inventor of Matlab and at the forefront of research into numerical computation and matrix computations. He was also the doctoral advisor of Charles Van Loan.

and is obtained with the direct application of its scalar counterpart:

$$\begin{aligned} f(\mathbf{M}) &= f(z_0) + \frac{\nabla f(z_0)}{1!}(\mathbf{M} - z_0 \mathbf{1}\mathbf{1}^\top) + \frac{\nabla^2 f(z_0)}{2!}(\mathbf{M} - z_0 \mathbf{1}\mathbf{1}^\top)^2 + \dots \\ &= \sum_{r=0}^{\infty} \frac{\nabla^r f(z_0)}{r!}(\mathbf{M} - z_0 \mathbf{1}\mathbf{1}^\top)^r \end{aligned} \quad (5.2)$$

where $\nabla^r f(z_0)$ is the r^{th} derivative of f at z_0 . More specifically, the MacLaurin series [201], when $z_0 = 0$ above, of the scalar exponential function translates directly to the matrix exponential function of $e^{t\mathbf{M}}$ [172]:

$$e^{t\mathbf{M}} = \mathbf{1}\mathbf{1}^\top + \frac{t\mathbf{M}}{1!} + \frac{t^2\mathbf{M}^2}{2!} + \frac{t^3\mathbf{M}^3}{3!} + \dots = \sum_{r=0}^{\infty} \frac{t^r \mathbf{M}^r}{r!}, \quad \mathbf{M} \in \mathbb{C}^{(n \times n)} \quad (5.3)$$

In computing the exponential of a matrix, the Taylor series above should be summed to an appropriate number of terms, until adding another term with floating point arithmetic does not alter the numbers stored in the computer. This method is known to give bad results, even in the scalar case, when the series is truncated too soon – when the approximation of what “does not alter the numbers” is a large increment compared to EPS [211]. The error bound on this truncation is considered in more depth in [127, 193, 206].

The attraction of the Taylor series approximation, when truncated appropriately, is that it only involves matrix multiplications. Furthermore, the computation of the powers of a matrix \mathbf{M}^r in Eq. (5.3) is made more efficient by expressing the matrix power in a binary expansion [106] e.g. \mathbf{M}^{13} requires 5 matrix multiplications:

$$\mathbf{M}^{13} = \left((\mathbf{M}^2)^2 \right)^2 (\mathbf{M}^2)^2 \mathbf{M}$$

The *Spinach* [130] implementation of the matrix exponential uses a Taylor series approximation with scaling and squaring (see §5.1.4 below), designed to preserve sparsity, essential for large matrix dimensions – NMR systems are in the millions, common in NMR [66]).

Spin Hamiltonians are guaranteed to be sparse in the Pauli basis [61, 67, 316] and their exponential propagators are also sparse when $\|\mathbf{H}\Delta t\| < 1$, if care is taken to eliminate insignificant elements after each matrix multiplication in the scaled and squared Taylor series procedure.

§ 5.1.2 Padé approximation

A rational function approximation of a matrix is directly translated from that of a scalar function to give

$$R_{pq}(\mathbf{M}) = [D_{pq}(\mathbf{M})]^{-1} N_{pq}(\mathbf{M}) \quad (5.4)$$

where N_{pq} and D_{pq} are polynomials in \mathbf{M} of at most degree- p of N_{pq} and q of D_{pq} . The numerator polynomial is

$$N_{pq}(\mathbf{M}) = \sum_{k=0}^p \frac{(p+q-k)!p!}{(p+q)!k!(p-k)!} \mathbf{M}^k \quad (5.5)$$

and the denominator polynomial is

$$D_{pq}(\mathbf{M}) = \sum_{k=0}^q \frac{(p+q-k)!q!}{(p+q)!k!(q-k)!} (-\mathbf{M})^k \quad (5.6)$$

When $q = 0$, the rational function approximation in EQ. (5.4) is the Taylor series truncated to order- p .

If $R_{pq}(\mathbf{M}) \in \mathbb{R}^{(p \times q)}$ and $D_{pq}(\mathbf{0}) = \mathbf{1}$, the rational function approximation in EQ. (5.4) is the $[p/q]$ Padé approximant [234]. The exact expression for the Padé approximation of the matrix exponential is [211, 212]

$$e^{t\mathbf{M}} = R_{pq}(t\mathbf{M}) + \frac{(-1)^q}{(p+q)!} (t\mathbf{M})^{p+q+1} [D_{pq}(t\mathbf{M})]^{-1} \int_0^1 u^p (1-u)^q e^{t\mathbf{M}(1-u)} du \quad (5.7)$$

Diagonal approximants, where $p = q$, are more accurate than when $p \neq q$ and are preferred in practice because $R_{rr}(\mathbf{M})$, with $r = \max(p, q)$, has the same computational cost as when $p \neq q$ [127].

As it stands in EQ. (5.7), the Padé approximant is only a good approximation of the matrix exponential near the origin, when $\|\mathbf{M}\|$ is small [106, 211, 212] (although the Chebyshev rational approximation can perform well in some circumstance §5.1.3) – requiring the *scaling and squaring* technique in §5.1.4 to overcome the problem.

It should be noted that calculation of the “perilous inverse” matrix $[D_{pq}(\mathbf{M})]^{-1}$ is required, which can be an undesirable calculation [196].

§ 5.1.3 Chebyshev rational approximation

If \mathbf{M} is a Hermitian matrix, then approximating its exponential amounts to approximating the exponential function on the spectrum of eigenvalues of \mathbf{M} [127]. The Chebyshev rational approximation^{1}, like the Padé approximant of §5.1.2 above, is based on the rational function approximation of Eq. (5.4).

$$C_{qq}(\mathbf{M}) = [D_{qq}(\mathbf{M})]^{-1} N_{qq}(\mathbf{M}) \quad (5.8)$$

where only the diagonal approximants are considered. The polynomials, $D_{qq}(\mathbf{M})$ and $N_{qq}(\mathbf{M})$, of order- q are those defined in Eqs. (5.5), (5.6). A minimiser for $\|C_{qq}(\mathbf{M}) - e^{\mathbf{M}}\|$ can be found from coefficients determined in [38, 42], where \mathbf{M} is Hermitian with negative eigenvalues. This is a useful method for sparse matrices. Unlike the Padé approximation of the matrix exponential, which is only valid near the origin, the Chebyshev rational approximation is valid for the whole negative real axis (or the whole positive real axis for $e^{-\mathbf{M}}$).

A success of this approximation is, when considering $e^{\mathbf{M}\Delta t}$ with a time step Δt , that the Chebyshev approximation allows much larger time steps compared to the Padé approximant [91]. However, Chebyshev series diverge with non-Hermitian matrices [219], where the error bounds for the scalar problem do not translate to matrices [127] – the method is expected to fail when considering dissipative dynamics.

§ 5.1.4 Scaling and squaring

The scaling and squaring technique [180, 325] exploits the relationship

$$e^{t\mathbf{M}} = \left(e^{\frac{t\mathbf{M}}{s}} \right)^s, \quad \mathbf{M} \in \mathbb{C}^{(n \times n)} \quad (5.9)$$

This can be used to control the round-off errors associated with the Taylor and Padé approximants: as $t\|\mathbf{M}\|$ or the spread of eigenvalues increases [211, 212]. The choice of s to be the smallest power of 2 for which $\|\mathbf{M}\|/s \leq 1$ [196, 211]:

$$e^{t\mathbf{M}} = \left(e^{\frac{t\mathbf{M}}{2^m}} \right)^{2^m} \quad (5.10)$$

^{1}The Chebyshev rational approximation is also known as the *Best L_∞ approximation*, or the *minimax approximation* [127].

Further efficiencies and error tolerance have been developed, the Padé approximants with scaling and squaring is the method of matrix exponential used in both Matlab and Mathematica [3, 126]. An attraction of its use with the Taylor series is that only approximate scaling is required [132, 219].

The next section §5.2 will use a block triangular form [235] of the matrix \mathbf{M} , for the calculation of directional derivatives:

$$\mathbf{M} = \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & \mathbf{D} \end{pmatrix} \quad (5.11)$$

The method of Padé approximants with scaling and squaring shows weakness when exponentiating block triangular matrices when $\|\mathbf{B}\| \gg \max(\|\mathbf{A}\|, \|\mathbf{D}\|)$ [127]: the result is that the diagonal blocks are over-scaled [54, 149].

§ 5.2 Directional derivatives

Derivatives of functions of matrices are called *directional derivatives*, or *Gâteaux derivatives*, which are defined in [127] as

$$D_{\mathbf{B}}(f(\mathbf{A})) \triangleq \lim_{h \rightarrow 0} \frac{f(\mathbf{A} + h\mathbf{B}) - f(\mathbf{A})}{h} = \frac{d}{dh} \bigg|_{h=0} f(\mathbf{A} + h\mathbf{B}) \quad (5.12)$$

This has a similar form to the finite difference equation in EQ. (3.12a) and the Taylor series truncated to first order in EQ. (3.3), and should be considered the formal definition of the derivative of the function of a matrix^{11}. The interpretation of $D_{\mathbf{B}}(f(\mathbf{A}))$ is: *the derivative of the function f at \mathbf{A} in the direction \mathbf{B}* [127, 218].

^{11}A directional derivative has a stronger notion of differentiability when it is a *Frechét derivative*. This is true if $\exists D_{\mathbf{B}}(f(\mathbf{A}))$ is a linear function of \mathbf{B} and continuous in \mathbf{A} [127]. Assuming these qualifications are met, which is the case in this thesis, the terms directional derivative and Frechét derivative are equivalent. The remainder of this thesis will use the more descriptive term *directional derivative*.

§ 5.2.1 Directional derivatives of the matrix exponential

The directional derivative of the matrix exponential, $e^{t\mathbf{A}}$, is then

$$D_{\mathbf{B}}(e^{t\mathbf{A}}) = \lim_{h \rightarrow 0} \frac{e^{t(\mathbf{A}+h\mathbf{B})} - e^{t\mathbf{A}}}{h} = \frac{d}{dh} \bigg|_{h=0} e^{t(\mathbf{A}+h\mathbf{B})} \quad (5.13)$$

which can also be expanded as a Taylor series [127]:

$$\begin{aligned} D_{\mathbf{B}}(e^{t\mathbf{A}}) &= e^{t(\mathbf{A}+\mathbf{B})} - e^{t\mathbf{A}} \\ &= \frac{t}{1!} \mathbf{B} + \frac{t^2}{2!} (\mathbf{AB} + \mathbf{BA}) + \frac{t^3}{3!} (\mathbf{A}^2\mathbf{B} + \mathbf{ABA} + \mathbf{BA}^2) + \dots \end{aligned} \quad (5.14)$$

This relates to the power directional derivatives [218] by

$$\begin{aligned} D_{\mathbf{B}}(e^{t\mathbf{A}}) &= \sum_{r=0}^{\infty} \frac{t^r}{r!} \frac{d^r}{dt^r} D_{\mathbf{B}}(e^{t\mathbf{A}}) \bigg|_{t=0} \\ &= \sum_{r=1}^{\infty} \frac{t^r}{r!} D_{\mathbf{B}}(e^{\mathbf{A}^r}) \end{aligned} \quad (5.15)$$

which leads to the *Magnus expansions* [116, 203].

An alternative, and more useful representation than Eqs. (5.13), (5.14) [218] is the integral representation [74, 144]:

$$D_{\mathbf{B}}(e^{t\mathbf{A}}) = \int_0^t e^{(t-s)\mathbf{A}} \mathbf{B} e^{s\mathbf{A}} ds \quad (5.16)$$

This will become a useful representation in the following sections. It is useful to note that rearranging this formula [218] leads to a commutator series [120, 203]

$$\begin{aligned} D_{\mathbf{B}}(e^{t\mathbf{A}}) &= \left[\int_0^t e^{s\mathbf{A}} \mathbf{B} e^{-s\mathbf{A}} ds \right] e^{t\mathbf{A}} \\ &= \left\{ \mathbf{B}, \int_0^t e^{s\mathbf{A}} ds \right\} e^{t\mathbf{A}} = \left\{ \mathbf{B}, \frac{e^{t\mathbf{A}} - \mathbb{1}}{\mathbf{A}} \right\} e^{t\mathbf{A}} \\ &= \left[\sum_{r=0}^{\infty} \frac{t^{r+1}}{(r+1)!} \{ \mathbf{B}, \mathbf{A}^r \} \right] e^{t\mathbf{A}} \end{aligned} \quad (5.17)$$

where the notation for the commutator power series relates to the commutator by $\{\mathbf{B}, \mathbf{A}^r\} = [\{\mathbf{B}, \mathbf{A}^{r-1}\}, \mathbf{A}]$.

§ 5.2.2 Block triangular auxiliary matrix exponential

A method for computing these chained exponential integrals was proposed by Van Loan in 1978 [196]. While investigating the sensitivity of the matrix exponential [195], he noted that the integrals in question are solutions to linear block matrix differential equations and suggested that block matrix exponentials are used to compute them. In the simplest case of a single integral [143], a block triangular auxiliary matrix [235] is exponentiated:

$$\mathbf{M} = \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & \mathbf{A} \end{pmatrix} \Rightarrow \begin{cases} e^{\mathbf{M}} = \begin{pmatrix} e^{\mathbf{A}} & \int_0^1 e^{(1-s)\mathbf{A}} \mathbf{B} e^{s\mathbf{A}} ds \\ \mathbf{0} & e^{\mathbf{A}} \end{pmatrix} \\ e^{t\mathbf{M}} = \begin{pmatrix} e^{t\mathbf{A}} & \int_0^t e^{(t-s)\mathbf{A}} \mathbf{B} e^{s\mathbf{A}} ds \\ \mathbf{0} & e^{t\mathbf{A}} \end{pmatrix} \end{cases} \quad (5.18a)$$

$$(5.18b)$$

where the above-diagonal block is easily identified as directional derivative in the integral representation of Eq. (5.16) [74, 144]. Now Eq. (5.18b) can be recast in a form to show explicitly the first directional derivative of the matrix exponential:

$$\mathbf{M} = \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & \mathbf{A} \end{pmatrix} \Rightarrow e^{t\mathbf{M}} = \begin{pmatrix} e^{t\mathbf{A}} & D_{\mathbf{B}}(e^{t\mathbf{A}}) \\ \mathbf{0} & e^{t\mathbf{A}} \end{pmatrix} \quad (5.19)$$

Further to Eq. (5.16), the r^{th} directional derivative can be defined recursively [218] by

$$D_{\mathbf{B}}^r(e^{t\mathbf{A}}) = r \int_0^t e^{\mathbf{A}(t-s)} \mathbf{B} D_{\mathbf{B}}^{r-1}(e^{s\mathbf{A}}) ds \quad (5.20)$$

It is useful to see the pattern when using larger block-bidiagonal, semi-circulant auxiliary matrices. Higher order directional derivatives of the matrix exponential can be calculated by constructing and exponentiating sparse block-bidiagonal

semi-circulant matrices [218]:

$$\mathbf{M} = \begin{pmatrix} \mathbf{A} & \mathbf{B} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A} & \mathbf{B} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{B} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{A} \end{pmatrix}$$

$$\implies e^{t\mathbf{M}} = \begin{pmatrix} \frac{1}{0!} D_{\mathbf{B}}^0 & \frac{1}{1!} D_{\mathbf{B}}^1 & \frac{1}{2!} D_{\mathbf{B}}^2 & \cdots & \frac{1}{r!} D_{\mathbf{B}}^r \\ \mathbf{0} & \frac{1}{0!} D_{\mathbf{B}}^0 & \frac{1}{1!} D_{\mathbf{B}}^1 & \cdots & \frac{1}{(r-1)!} D_{\mathbf{B}}^{r-1} \\ \mathbf{0} & \mathbf{0} & \frac{1}{0!} D_{\mathbf{B}}^0 & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \frac{1}{1!} D_{\mathbf{B}}^1 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \frac{1}{0!} D_{\mathbf{B}}^0 \end{pmatrix} \quad (5.21)$$

This is set out in a general way in [218], and for this specific case of the first directional derivative in [81]; derivation of the latter and for that of the second directional derivative is set out in the next section §5.3.

Van Loan's method was subsequently refined by Carbonell et. al. [37], deriving a convenient expression using the exponential of a block-bidiagonal semi-circulant auxiliary matrix:

$$\mathbf{M} = \begin{pmatrix} \mathbf{A}_1 & \mathbf{B}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 & \mathbf{B}_2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_3 & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{B}_{r-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{A}_r \end{pmatrix}$$

$$\implies e^{\mathbf{M}t} = \begin{pmatrix} \mathbf{D}_{11} & \mathbf{D}_{12} & \mathbf{D}_{13} & \cdots & \mathbf{D}_{1r} \\ \mathbf{0} & \mathbf{D}_{22} & \mathbf{D}_{23} & \cdots & \mathbf{D}_{2r} \\ \mathbf{0} & \mathbf{0} & \mathbf{D}_{33} & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{D}_{r-1,r} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D}_{rr} \end{pmatrix} \quad (5.22)$$

The integrals in question populate the rows of the resulting block matrix, having the chained integral form of EQ. (5.1), e.g.

$$\mathbf{D}_{1r} = \int_0^t dt_1 \int_0^{t_1} dt_2 \cdots \int_0^{t_{r-2}} dt_{r-1} \left\{ e^{(t-t_1)\mathbf{A}_1} \mathbf{B}_1 e^{(t_1-t_2)\mathbf{A}_2} \mathbf{B}_2 \cdots \mathbf{B}_{r-1} e^{t_r \mathbf{A}_r} \right\} \quad (5.23)$$

§ 5.3 Directional derivatives of the Propagator

Fidelity functional derivatives of optimal control [86, 311] are a form of chained exponential integrals which have expensive matrix factorization in their evaluation [1, 73, 103]. A solution to this problem is the observation that matrix exponentiation does not require factorisations and preserves the spin operator sparsity [130, 166] and the use of an auxiliary matrix technique [37, 81, 196, 218].

The GRAPE method was shown in §§3, 4 to discretise time and solve Eq. (3.60) using thin slice propagators. From the computational efficiency point of view, the central problem is therefore the calculation of first derivatives of the fidelity functional in Eqs. (3.79a)–(3.79c) and mixed second derivatives in Eqs. (4.7a)–(4.7c), e.g.

$$\frac{\partial \mathcal{J}_0}{\partial c_{k,n}} = \left\langle \chi_{n+1} \left| D_{\mathbf{H}_k}(\mathbf{P}_n) \right| \rho_{n-1} \right\rangle \quad (5.24a)$$

$$\frac{\partial^2 \mathcal{J}_0}{\partial c_{k,n} \partial c_{j,m}} = \left\langle \chi_{n+1} \left| D_{\mathbf{H}_k}(\mathbf{P}_n) \mathbf{P}_{n-1} \cdots \mathbf{P}_{m+1} D_{\mathbf{H}_j}(\mathbf{P}_m) \right| \rho_{m-1} \right\rangle \quad (5.24b)$$

where time slice propagators (assuming a fixed time grid step Δt) are defined in Eq. (3.75) as

$$\mathbf{P}_n = \exp \left[-i \mathbf{H}_n \Delta t \right] = \exp \left[-i \left(\mathbf{H}_0 + \sum_{k=1}^K c_{k,n} \mathbf{H}_k \right) \Delta t \right] \quad (5.25)$$

where the relaxation term is removed and the double-hat superoperator notation is replaced by a matrix representation bold-font for notational brevity.

In addition to the first derivative of the propagators, the second derivatives are required for the construction of the Hessian block diagonal. After separating the cases where efficiencies can be made with commuting control operators, discussed in §4.1.3, the remaining calculations from Eqs. (4.5a)–(4.5c), (4.6a)–(4.6c) are

$$\frac{\partial^2 \mathcal{J}_0}{\partial c_{k,n}^2} = \left\langle \chi_{n+1} \left| D_{\mathbf{H}_k}^2(\mathbf{P}_n) \right| \rho_{n-1} \right\rangle \quad (5.26a)$$

$$\frac{\partial^2 \mathcal{J}_0}{\partial c_{k,n} \partial c_{j,n}} = \left\langle \chi_{n+1} \left| D_{\mathbf{H}_k \mathbf{H}_j}^2(\mathbf{P}_n) \right| \rho_{n-1} \right\rangle \quad (5.26b)$$

The primary task therefore is to calculate first and second directional derivatives of the slice propagators.

§ 5.3.1 First directional derivatives

The original derivation of the first propagator directional derivative from [154] follows by considering a small perturbation to the control amplitude $\delta c_{k,n}$ on the control operator to derive the change in the propagator, $\delta \mathbf{P}_n$, then to form a derivative from $\delta \mathbf{P}_n / \delta c_{k,n}$. Using Eqs. (5.12), (5.25) this can be written as

$$\frac{\delta \mathbf{P}_n}{\delta c_n} = D_{\mathbf{H}_k}(\mathbf{P}_n) = \left(\lim_{\{c_{k,n} \rightarrow 0\}} \frac{\exp[-i\mathbf{H}_n \Delta t] - \exp[-i\mathbf{H}_0 \Delta t]}{c_{k,n}} \right) \quad (5.27)$$

The procedure in [154] then finds solution by truncating Eq. (5.17) to first order in Δt , giving

$$D_{\mathbf{H}_k}(\mathbf{P}_n) = \Delta t \left[\mathbf{H}_k, \mathbf{P}_n \right], \quad \Delta t \ll \left\| \mathbf{H}_0 + \sum_{k=1}^K c_{k,n} \mathbf{H}_k \right\|^{-1} \quad (5.28)$$

An alternative representation of Eq. (5.27) is the integral form of Eq. (5.16):

$$D_{\mathbf{H}_k}(\mathbf{P}_n) = \int_0^{-i\Delta t} e^{(-i\Delta t - \tau)\mathbf{H}_n} \mathbf{H}_k e^{\tau\mathbf{H}_n} d\tau \quad (5.29)$$

Using Eq. (5.18b) and setting $\mathbf{A} \leftarrow \mathbf{H}_n$, $\mathbf{B} \leftarrow \mathbf{H}_k$, and $t \leftarrow -i\Delta t$, gives the block triangular auxiliary matrix exponential as [81]

$$\mathbf{L}_2 = \begin{pmatrix} \mathbf{H}_n & \mathbf{H}_k \\ \mathbf{0} & \mathbf{H}_n \end{pmatrix} \quad \Rightarrow \quad e^{-i\mathbf{L}_2 \Delta t} = \begin{pmatrix} \mathbf{P}_n & D_{\mathbf{H}_k}(\mathbf{P}_n) \\ \mathbf{0} & \mathbf{P}_n \end{pmatrix} \quad (5.30)$$

where the above-diagonal is the first directional derivative required for gradient element calculation in Eqs. (5.24a), (5.24b).

§ 5.3.2 Second directional derivatives

The integral representation of the second directional derivatives form a chained integral as in EQ. (5.1):

$$D_{\mathbf{H}_k}^2(\mathbf{P}_n) = \int_0^{-i\Delta t} \int_0^{\tau} e^{(-i\Delta t - \tau)\mathbf{H}_n} \mathbf{H}_k e^{(\tau - v)\mathbf{H}_n} \mathbf{H}_k e^{v\mathbf{H}_n} dv d\tau \quad (5.31a)$$

$$D_{\mathbf{H}_k \mathbf{H}_j}^2(\mathbf{P}_n) = \int_0^{-i\Delta t} \int_0^{\tau} e^{(-i\Delta t - \tau)\mathbf{H}_n} \mathbf{H}_k e^{(\tau - v)\mathbf{H}_n} \mathbf{H}_j e^{v\mathbf{H}_n} dv d\tau \quad (5.31b)$$

Just as in the previous calculation of first directional derivatives in EQ. (5.30), EQ. (5.21) shows a similar auxiliary matrix technique can be used to calculate higher order directional derivatives. Calculation of the second directional derivatives requires a 3×3 bidiagonal block matrix auxiliary matrix from EQ. (5.21),

$$\mathbf{L}_3 = \begin{pmatrix} \mathbf{H}_n & \mathbf{H}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_n & \mathbf{H}_j \\ \mathbf{0} & \mathbf{0} & \mathbf{H}_n \end{pmatrix} \quad (5.32)$$

where the upper-diagonal blocks contain the two control operators needed for the mixed second directional derivatives. Following from Eqs. (5.22), (5.23), the exponential of the auxiliary matrix \mathbf{L}_3 is

$$\Rightarrow e^{-i\mathbf{L}_3\Delta t} = \begin{pmatrix} \left(\begin{array}{ccc} \mathbf{P}_n & D_{\mathbf{H}_k}(\mathbf{P}_n) & \frac{1}{2}D_{\mathbf{H}_k}^2(\mathbf{P}_n) \\ \mathbf{0} & \mathbf{P}_n & D_{\mathbf{H}_k}(\mathbf{P}_n) \\ \mathbf{0} & \mathbf{0} & \mathbf{P}_n \end{array} \right), & j = k \\ \left(\begin{array}{ccc} \mathbf{P}_n & D_{\mathbf{H}_k}(\mathbf{P}_n) & \frac{1}{2}D_{\mathbf{H}_k \mathbf{H}_j}^2(\mathbf{P}_n) \\ \mathbf{0} & \mathbf{P}_n & D_{\mathbf{H}_j}(\mathbf{P}_n) \\ \mathbf{0} & \mathbf{0} & \mathbf{P}_n \end{array} \right), & j \neq k \end{pmatrix} \quad (5.33a)$$

$$\Rightarrow e^{-i\mathbf{L}_3\Delta t} = \begin{pmatrix} \left(\begin{array}{ccc} \mathbf{P}_n & D_{\mathbf{H}_k}(\mathbf{P}_n) & \frac{1}{2}D_{\mathbf{H}_k}^2(\mathbf{P}_n) \\ \mathbf{0} & \mathbf{P}_n & D_{\mathbf{H}_k}(\mathbf{P}_n) \\ \mathbf{0} & \mathbf{0} & \mathbf{P}_n \end{array} \right), & j = k \\ \left(\begin{array}{ccc} \mathbf{P}_n & D_{\mathbf{H}_k}(\mathbf{P}_n) & \frac{1}{2}D_{\mathbf{H}_k \mathbf{H}_j}^2(\mathbf{P}_n) \\ \mathbf{0} & \mathbf{P}_n & D_{\mathbf{H}_j}(\mathbf{P}_n) \\ \mathbf{0} & \mathbf{0} & \mathbf{P}_n \end{array} \right), & j \neq k \end{pmatrix} \quad (5.33b)$$

where the upper-right blocks are the second directional derivatives required for Hessian element calculation in Eqs. (5.26a), (5.26b).

§ 5.4 Krylov subspace techniques

There is an important efficiency to be made when considering the propagator derivatives of in that only the resulting vector $\left| \frac{\partial^2 \rho_{n+1}}{\partial c_{k,n} \partial c_{j,n}} \right\rangle$ is required in Eqs. (5.24a),

(5.26a), (5.26b). The solutions to these equations are of the form $f(\mathbf{M})b$ – the action of $f(\mathbf{M})$ on the vector b is required [321]. This type of problem is readily solved with *Krylov subspace methods*, such as the *Arnoldi process* [9] or the *Lanczos process* [173], avoiding explicit calculation of the matrix exponential [106, 127, 128, 256].

The Arnoldi process is implemented in *Spinach* to efficiently calculate [165, 170] the exponential of a Liouvillian multiplied by a state vector. The effect is that forward and backward propagation can be calculated in this way. Furthermore, the effect of first and second derivatives on the forward propagated state vector can be calculated with their auxiliary matrices:

FORWARD PROPAGATION:

$$\left[\exp(-i\Delta t \mathbf{H}_n) \right] \left| \rho_n \right\rangle = \left| \rho_{n+1} \right\rangle \quad (5.34a)$$

BACKWARD PROPAGATION:

$$\left[\exp(+i\Delta t \mathbf{H}_n) \right] \left| \chi_n \right\rangle = \left| \chi_{n-1} \right\rangle \quad (5.34b)$$

PROPAGATED FIRST DIRECTIONAL DERIVATIVE:

$$\left[\exp\left(-i\Delta t \begin{pmatrix} \mathbf{H}_n & \mathbf{H}_k \\ \mathbf{0} & \mathbf{H}_n \end{pmatrix}\right) \right] \begin{bmatrix} \mathbf{0} \\ \left| \rho_n \right\rangle \end{bmatrix} = \begin{bmatrix} \left| \frac{\partial \rho_{n+1}}{\partial c_{k,n}} \right\rangle \\ \mathbf{0} \end{bmatrix} \quad (5.34c)$$

PROPAGATED SECOND DIRECTIONAL DERIVATIVE:

$$\left[\exp\left(-i\Delta t \begin{pmatrix} \mathbf{H}_n & \mathbf{H}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_n & \mathbf{H}_j \\ \mathbf{0} & \mathbf{0} & \mathbf{H}_n \end{pmatrix}\right) \right] \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \left| \rho_n \right\rangle \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \left| \frac{\partial^2 \rho_{n+1}}{\partial c_{k,n} \partial c_{j,n}} \right\rangle \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (5.34d)$$

§ 5.5 Propagator recycling

Two computational efficiencies should be taken into account when calculating the full Hessian matrix of FIG. 4.1. Firstly, the first propagator direction derivatives do not need to be calculated with EQ. (5.30) when a Hessian calculation is made: these directional derivatives are already contained in the above-diagonal of EQ. (5.33b). Secondly, Krylov subspace techniques can be used to calculate the effect of the propagator directional derivative on the state $\left| \rho_n \right\rangle$ in some of Eqs. (5.24a), (5.24b),

(5.26a), (5.26b). The following lists outline the type of calculation recommended to construct fidelity derivatives. There are two cases:

- ① A fidelity and gradient vector calculation for a gradient ascent method, quasi-Newton method, or a line search method.
- ② The simultaneous calculation of the fidelity, gradient vector and Hessian matrix for a Newton step calculation.

§ 5.5.1 Fidelity and gradient calculation

① FORWARD AND BACKWARD PROPAGATION

Number of matrix exponential calculations = $2 \times K \times N$.

- I. Calculate and store forward propagated states with a Krylov subspace technique in Eq. (5.34a).
- II. Calculate and store backward propagated states with a Krylov subspace technique in Eq. (5.34b).
- III. Calculate fidelity as the overlap of the forward propagated state of STEP ①.I at the time step $n = N$, and the desired target state using one of Eqs. (3.67)–(3.69).

② GRADIENT ELEMENTS

Number of matrix exponential calculations = $K \times N$.

- I. Use an auxiliary matrix in Eq. (5.30) and forward propagated state from STEP ①.I with a Krylov subspace technique in Eq. (5.34c) and store the resulting vector.
- II. Calculate the gradient elements as the overlap of the propagated gradient from STEP ②.I and the backward propagated state from STEP ①.II using one of Eqs. (3.79a)–(3.79c).

§ 5.5.2 Fidelity, gradient and Hessian calculation

① FORWARD AND BACKWARD PROPAGATION

Number of matrix exponential calculations = $2 \times K \times N$.

- I. Calculate and store forward propagated states with a Krylov subspace technique in Eq. (5.34a).
- II. Calculate and store backward propagated states with a Krylov subspace technique in Eq. (5.34b).
- III. Calculate fidelity as the overlap of the forward propagated state of STEP ①.I at the time step $n = N$, and the desired target state using one of Eqs. (3.67)–(3.69).

② DIAGONAL HESSIAN ELEMENTS – FIG. 4.1 (A)

Number of matrix exponential calculations = $K \times N$.

- I. Calculate explicit propagators with Eq. (5.33b), exponentiated using a sparsity preserving Taylor series of §5.1.1 with scaling and squaring of §5.1.4.
- II. Store the propagator from block (1, 1).
- III. Store first directional derivatives from block (1, 2).
- IV. Store second directional derivatives from block (1, 3).
- V. Calculate diagonal Hessian elements with Eq. (5.26a) using the stored states from STEPS ①.I, ①.II.

③ OFF-DIAGONAL ELEMENTS OF THE HESSIAN BLOCK DIAGONAL – FIG. 4.1 (B)

Needed when $K > 1$.

Control operator commutativity must be considered from §4.1.3.

$(K-1)! \times N < \text{Number of matrix exponential calculations} < K(K-1) \times N$.

- I. Use an auxiliary matrix in Eq. (5.33b) with a Krylov subspace technique as in Eq. (5.34d) and store the resulting vector.

II. Calculate off-diagonal elements of the Hessian block diagonal with the backward propagated states from STEP ①.II subject commutativity conditions in Eqs. (4.8)–(4.10).

④ BLOCK OFF-DIAGONAL HESSIAN ELEMENTS – FIG. 4.1 ④

Requires only matrix-matrix and matrix-vector multiplications.

I. Retrieve the two required first directional derivatives from STEP ②.III.

II. Calculate off-diagonal Hessian elements with Eq. (5.24b), using the stored states from STEPS ①.I, ①.II and the stored propagators in STEP ②.II.

⑤ GRADIENT ELEMENTS

Requires only matrix-matrix and matrix-vector multiplications.

I. Use the first directional derivative from STEP ②.III and forward propagated state from STEP ①.I and store the resulting vector.

II. Calculate the gradient elements as the overlap of the propagated gradient from STEP ⑤.I and the backward propagated state from STEP ①.II using one of Eqs. (3.79a)–(3.79c).

§ 5.5.3 Parallelisation

The numerical implementations of gradient and Hessian calculations should be parallelised with respect to the number of time slices in Eq. (4.5c). Gradient calculation uses 2×2 augmented exponentials Eq. (5.34c) and Hessian calculation uses the 3×3 augmented exponentials shown in Eq. (5.33b). The Hessian function has a further parallel loop when calculating $n \neq m$ blocks in Equation Eq. (4.7c). At that stage, the first derivatives have already been calculated when solving Eq. (5.33b); they are recycled.

Parallelisation efficiency analysis is given in FIG. 5.1. The scaling depends on the number of time slices in the control sequence and the parallelisation is efficient all the way to the number of CPU cores being half the number of time slices (over which the parallel loop is running). Given the same computing resources, a Hessian calculation takes approximately five-ten times longer than a gradient calculation. Because propagator derivatives are recycled during Hessian calculation, significant

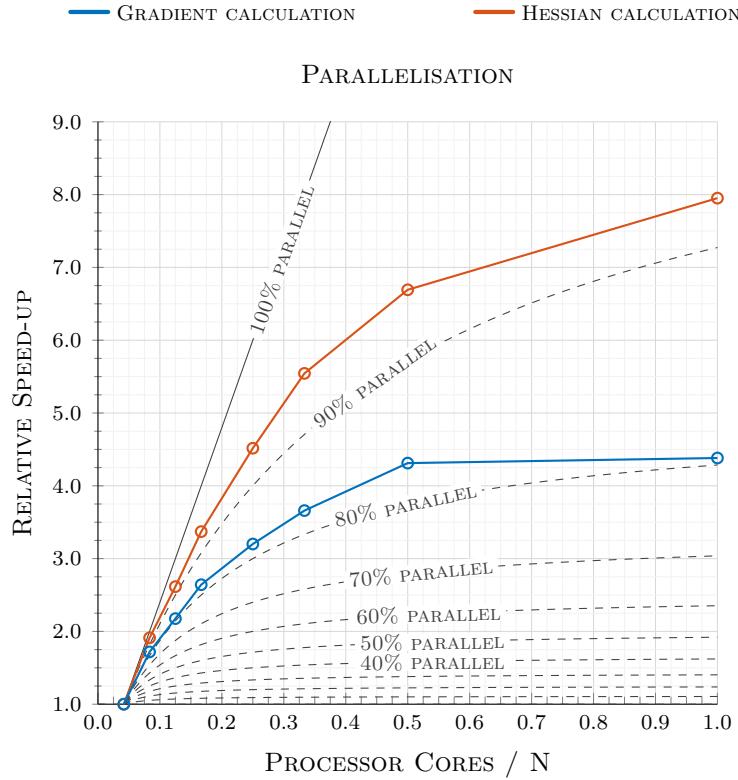


FIGURE 5.1: Amdahl’s law [4] parallelisation efficiency analysis for the Hessian calculation compared to the gradient calculation within *Spinach* implementation of GRAPE [86, 130]. The optimal control problem involves $N = 24$ time slices and $K = 6$ control channels, yielding a fidelity functional gradient vector with 144 elements and a 144×144 Hessian matrix.

efficiency gains may be made by optimising their storage and indexing, which is outlined in [108].

Using Amdahl’s law, the parallelism of code from FIG. 4.7 was computed to be just less than 70% parallel. This indicates that a maximum of about $3\times$ speed increase to the code can be achieved using about 16 cores for parallel computation, although using only 8 cores will give a speed increase of over $2.5\times$.

§ 6

Constrained Optimal Control

In the end, man is not entirely guilty – he did not start history. Nor is he wholly innocent – he continues it.

– Albert Camus, THE REBEL

The Pontryagin maximum principle of §3.2.3 set out a control problem with a *terminal cost* and a *running cost*. The *terminal cost* was investigated fully as fidelity measures in EQS. (3.67)–(3.69). This chapter will set out a number of *running costs* in the form of penalty functions \mathcal{K} .

The idea of a penalty function is one borne from constrained optimisation [23, 79, 229]. The constraint may be in the form of an inequality in addition to the objective function e.g. in the context of control operators; the control amplitudes may have a maximum allowed value corresponding to a maximum amount of power the control pulse may have – in this case the constraint would be an experimental constraint.

The traditional area of constrained optimisation is to define an allowable region of the objective function; then only to allow the objective function evaluations that fall within this region. In the penalty functions that follow in this chapter, the idea of a penalty function from constrained optimisation is relaxed to allow objective function evaluations outside the constrained area, but at a cost.

This chapter outlines penalty functions as strategies to push solutions to the objective function into a desired region of its space. The scalar function to maximise is (using the fidelity measure of EQ. (3.68));

$$\mathcal{J}_1 \triangleq \text{Re}\langle \sigma | \rho(T) \rangle - \mathcal{K}(\rho(t), \mathbf{c}(t)) \quad (6.1)$$

where \mathcal{K} is a penalty functional enforcing the experimental constraints on the optimisation e.g. a maximum amplitude of control pulses. The task to include penalties is now to design the penalty measure, and then to find its associated gradient and Hessian.

The simplest method, but maybe affecting convergence the most, is to change the variables so that controls more strictly defined in §6.1, or more elegantly defined in §6.1.1. Penalties to pulse power are set out in §6.2, with the simplest favouring lower power penalties in §6.2.1 and a more calibrated penalty that only penalises excursions outside a desired bound §6.2.2. An interesting penalty function is set out in §6.3, designed to find smooth solutions by penalising rapid changes in controls over their time period.

This smoothing penalty is investigated in §6.3.1 with the problem of transferring magnetisation to a singlet state. Although the analytical solution is the optimal solution, it is not robust to B_1 miscalibration or B_0 resonance offsets. These type of optimal pulses are reproduced from [178] in §6.3.2, then the optimisation problem is extended to attempt to find similar, but also smooth optimal, robust solutions in §6.3.3.

§ 6.1 Change of variables

There exists a situation in which all solutions *must* be bounded i.e. a waveform must not exceed power levels which hardware cannot handle. To constrain all possible solutions to the feasible region R [23], and to guarantee a minimiser resides within the region R , more is required than a method of weighted penalties. To constrain all solutions to the feasible region, a transform can be made to the objective function, where the transform maps the objective function to a similar one which only contains feasible points.

The variables forming the shape can be designed to form a repeating, mirrored structure; allowing all real numbers to be evaluated by an objective function in a continuous manner. The transform of a waveform, x , to include these constraints is

$$\tilde{c} = \left| c - 4 \left\lfloor \frac{c-1}{4} \right\rfloor - 3 \right| - 1 \quad (6.2)$$

where c is the unbounded waveform and $c - n \lfloor c/n \rfloor$ is the remainder from dividing c by n .

§ 6.1.1 Transforms between Cartesian & Polar Coordinates

It is normal to represent control pulses as operators of Cartesian space i.e. $\hat{\mathcal{H}}_x$ and $\hat{\mathcal{H}}_y$. However, it is common for spectrometer hardware to implement control pulses in polar coordinate space, with amplitude and phase. Furthermore, the most simple way to enforce an equality constraint on either the phase or amplitude is to allow only one of these as the optimisation variables. This was implemented in §§ 4.3.2, 4.3.3, where only the phase was allowed to vary, ensuring that the amplitude of the pulse stayed at a constant level.

To use this type of optimisation, the control pulses, the fidelity gradient, and the fidelity Hessian must be transformed between a Cartesian representation and a spherical polar coordinate system. The definitions of these transformed control pulses are

$$r = \sqrt{x^2 + y^2} \quad (6.3)$$

$$\varphi = \text{atan2} \left(\frac{y}{x} \right) \quad (6.4)$$

with atan2 defined as the four quadrant inverse tangent, and (x, y) are the components of the control pulse corresponding to $\hat{\mathcal{H}}_x$ and $\hat{\mathcal{H}}_y$. From this, the inverse transform is

$$x = r \cos \varphi \quad (6.5)$$

$$y = r \sin \varphi \quad (6.6)$$

These definitions are from basic Pythagorean theorem. However, the gradient and Hessian transformations are not so obvious, and it is useful to explicitly show their transformations here.

The gradient transform from polar to Cartesian coordinates is

$$\frac{d}{dx} = \cos \varphi \frac{d}{dr} - \frac{1}{r} \sin \varphi \frac{d}{dr} \quad (6.7)$$

$$\frac{d}{dy} = \sin \varphi \frac{d}{dr} + \frac{1}{r} \cos \varphi \frac{d}{dr} \quad (6.8)$$

and the inverse of these transforms are

$$\frac{d}{dr} = \sin \varphi \frac{d}{dy} + \cos \varphi \frac{d}{dx} \quad (6.9)$$

$$\frac{d}{d\varphi} = x \frac{d}{dy} - y \frac{d}{dx} \quad (6.10)$$

The transform of the Cartesian Hessian element to a polar Hessian element is

$$\frac{d^2}{dr^2} = \cos^2(\varphi) \cdot \frac{d^2}{dx^2} + 2 \cos(\varphi) \cdot \sin(\varphi) \frac{d^2}{dxdy} + \sin^2(\varphi) \cdot \frac{d^2}{dy^2} \quad (6.11)$$

$$\begin{aligned} \frac{d^2}{d\varphi^2} &= -r \cdot \left(\cos(\varphi) \cdot \frac{d}{dx} + \sin(\varphi) \cdot \frac{d}{dy} \right) \\ &+ r \cdot r \cdot \left(\cos^2(\varphi) \cdot \frac{d^2}{dy^2} - 2 \cos(\varphi) \cdot \sin(\varphi) \frac{d^2}{dxdy} + \sin^2(\varphi) \cdot \frac{d^2}{dx^2} \right) \end{aligned} \quad (6.12)$$

$$\begin{aligned} \frac{d^2}{drd\varphi} &= \cos(\varphi) \cdot \frac{d}{dy} - \sin(\varphi) \cdot \frac{d}{dx} \\ &+ r \cdot \cos(\varphi) \cdot \sin(\varphi) \frac{d^2}{dy^2} + r \cdot \cos^2(\varphi) \cdot \frac{d^2}{dxdy} \\ &- r \cdot \sin^2(\varphi) \cdot \frac{d^2}{dxdy} - r \cdot \cos(\varphi) \cdot \sin(\varphi) \frac{d^2}{dx^2} \end{aligned} \quad (6.13)$$

The transform of the polar Hessian elements back to Cartesian Hessian elements is

$$\begin{aligned} \frac{d^2}{dx^2} &= +2x \cdot y \cdot \frac{d}{d\varphi} \cdot \frac{1}{r^4} + y \cdot y \cdot \frac{d^2}{d\varphi^2} \cdot \frac{1}{r^4} + \frac{d}{dr} \cdot \frac{1}{r} \\ &- x \cdot x \cdot \frac{d}{dr} \cdot \frac{1}{r^3} - 2x \cdot y \cdot \frac{d^2}{drd\varphi} \cdot \frac{1}{r^3} + x \cdot x \cdot \frac{d^2}{dr^2} \cdot \frac{1}{r^2} \end{aligned} \quad (6.14)$$

$$\begin{aligned} \frac{d^2}{dy^2} &= -2x \cdot y \cdot \frac{d}{d\varphi} \cdot \frac{1}{r^4} + x \cdot x \cdot \frac{d^2}{d\varphi^2} \cdot \frac{1}{r^4} + \frac{d}{dr} \cdot \frac{1}{r} \\ &- y \cdot y \cdot \frac{d}{dr} \cdot \frac{1}{r^3} + 2x \cdot y \cdot \frac{d^2}{drd\varphi} \cdot \frac{1}{r^3} + y \cdot y \cdot \frac{d^2}{dr^2} \cdot \frac{1}{r^2} \end{aligned} \quad (6.15)$$

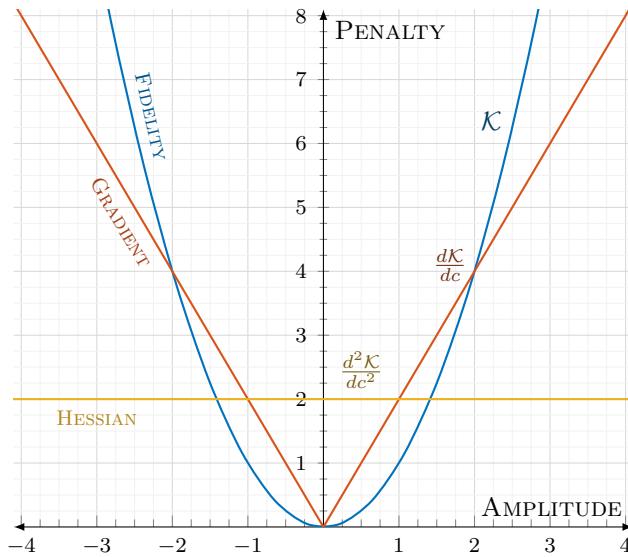


FIGURE 6.1: Norm square penalty and its derivatives

$$\begin{aligned} \frac{d^2}{dxdy} = & -\frac{d}{d\varphi} \cdot \frac{1}{r^2} + 2y \cdot y \cdot \frac{d}{d\varphi} \cdot \frac{1}{r^4} - x \cdot y \cdot \frac{d^2}{d\varphi^2} \cdot \frac{1}{r^4} - x \cdot y \cdot \frac{d}{dr} \cdot \frac{1}{r^3} \\ & + x \cdot x \cdot \frac{d^2}{drd\varphi} \cdot \frac{1}{r^3} - y \cdot y \cdot \frac{d^2}{drd\varphi} \cdot \frac{1}{r^3} + x \cdot y \cdot \frac{d^2}{dr^2} \cdot \frac{1}{r^2} \quad (6.16) \end{aligned}$$

These transforms for the Hessian elements are basic mathematics, but at the same time they are laborious algebraic differentiation using the chain rule. They are presented here to as convenient relations to enable optimisation of phase or amplitude pulses using the Newton-GRAPE method. Amplitude penalties can be implemented with these transforms and the power penalties set out below.

§ 6.2 Power penalties

Magnetic resonance pulses may not have an infinite power, they must be bounded by that which the magnetic resonance experiment can handle. A simple strategy is to penalise the vector of control pulses based on the amplitude of each pulse in the pulse set.

§ 6.2.1 Norm-square penalty

First consider a normalised set of control pulses; normalised by a bound set by the experiment being a nominal “maximum amplitude level” that a control channel

can experimentally implement. This maximum amplitude level, measured in Hz, can be arbitrarily set by an experienced experimenter or set by strict experimental constraints^{1}.

Assuming that the vector of control pulses is normalised with a maximum power amplitude, a penalty functional can be based on the square of the control pulse vector \mathbf{c} . This will give a very small penalty for low power control pulses, compared to the bound, and will give very high penalties for those above the bound. A form for this penalty is

$$\mathcal{K} = \frac{1}{N} \sum_k c_k^2 \quad \Rightarrow \quad \mathcal{K} = \frac{1}{N} \|\mathbf{c}\|^2 \quad (6.17)$$

where the scalar product of the vector with itself, $\mathbf{c} \cdot \mathbf{c}$, implies the square of the euclidean norm of the control vector, $\|\mathbf{c}\|^2$. Here, c_k denotes an element of the vector \mathbf{c} – the penalty is simply the sum of the squares of each element of the control pulse vector. The normalisation factor for this penalty, N , is the length of each control channel i.e. the number of discrete time slices. Essentially, the penalty \mathcal{K} is the average of the square of each control pulse, for each control channel.

Mentioned previously, for the use of a penalty in a gradient based optimisation scheme, we must also penalise the derivative of the penalty. Additionally, if the optimisation method calculates an explicit Hessian i.e. the Newton-Raphson method, then we must also penalise this Hessian. These derivatives of the penalty functional are with respect to the control pulse vector, just as the derivative of the fidelity functional is performed with respect to the same control pulse vector.

$$\frac{d}{dc_k} \mathcal{K} = \frac{1}{N} \sum_k 2c_k \quad \Rightarrow \quad \nabla \mathcal{K} = \frac{2}{N} \mathbf{c} \quad (6.18)$$

$$\frac{d^2}{dc_k^2} \mathcal{K} = \frac{2}{N} \quad \Rightarrow \quad \nabla^2 \mathcal{K} = \frac{2}{N} \mathbb{1} \quad (6.19)$$

The Hessian penalty is only ever a diagonal matrix because mixed derivatives of the penalty functional are zero. A graphical representation of this norm-squared penalty and its derivatives is shown in FIG. 6.1.

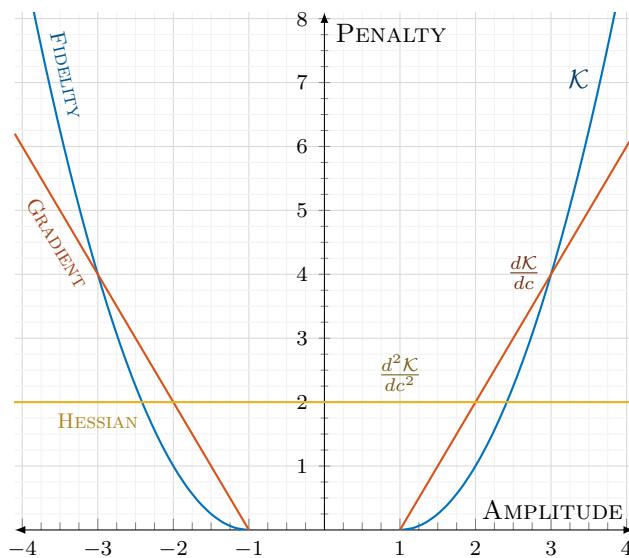


FIGURE 6.2: Spillout norm square penalty and its derivatives

§ 6.2.2 Spillout-norm-square penalty

The problem with the norm-squared penalty of Eq. (6.17) is that it penalises the objective function even when within the set bounds. This is undesirable because an optimisation will never truly converge unless there is zero amplitude on the control pulses - clearly an unrealistic solution. One way to remedy this is to introduce a level of zero penalty when within the maximum amplitude interval $[-1, +1]$. The graphical representation of this modified norm-squared penalty, penalising only that which spills out of the amplitude bounds, is shown in FIG. 6.2.

The aim of this spillout-norm-squared penalty is to only penalise the control amplitudes that exceed the set bounds of the normalised control pulses, eventually constraining solutions to the set bounds. This method also allows a quadratic convergence, assuming a solution does exist within the set bounds. The form of

{1} a reasonable estimate for a non-specific NMR spectrometer can set this bound as 10 kHz; giving a normalisation constant of $10 \times 2\pi \times 10^3$

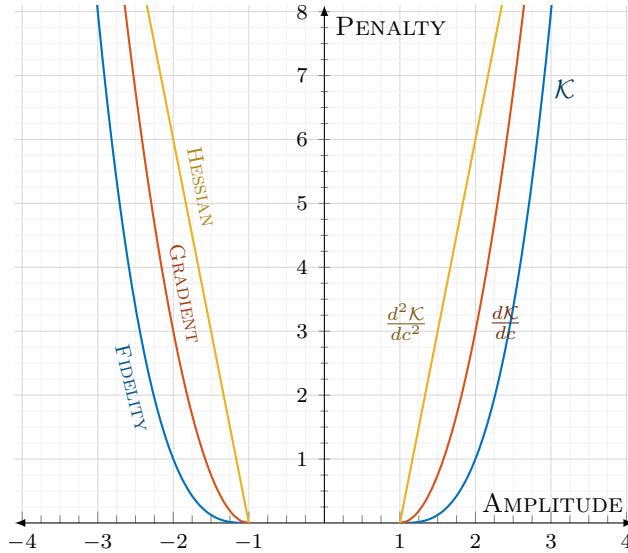


FIGURE 6.3: Spillout norm cube penalty and its derivatives

this fidelity functional and the corresponding derivatives are:

$$\mathcal{K} = \begin{cases} \frac{1}{N} \sum_k \left(|c_k| - 1 \right)^2, & |c_k| > 1 \\ 0, & |c_k| \leq 1 \end{cases} \implies \mathcal{K} = \begin{cases} \frac{1}{N} \|\mathbf{c} - \mathbf{1}\|^2, & |c_k| > 1 \\ \mathbf{0}, & |c_k| \leq 1 \end{cases} \quad (6.20a)$$

$$\frac{d}{dc_k} \mathcal{K} = \begin{cases} \frac{1}{N} \sum_k \left(|c_k| - 1 \right), & |c_k| > 1 \\ 0, & |c_k| \leq 1 \end{cases} \implies \nabla \mathcal{K} = \begin{cases} \frac{2}{N} (\mathbf{c} - \mathbf{1}), & |c_k| > 1 \\ \mathbf{0}, & |c_k| \leq 1 \end{cases} \quad (6.20b)$$

$$\frac{d^2}{dc_k^2} \mathcal{K} = \begin{cases} \frac{2}{N}, & |c_k| > 1 \\ 0, & |c_k| \leq 1 \end{cases} \implies \nabla^2 \mathcal{K} = \begin{cases} \frac{2}{N} \mathbf{1}, & |c_k| > 1 \\ \mathbf{0}, & |c_k| \leq 1 \end{cases} \quad (6.20c)$$

This penalty type has a discontinuity in the Hessian penalty, which is not expected to cause too much problem, but if this is the case a modification could be to use the a cubic penalty, shown in FIG. 6.3.

§ 6.3 Smoothed Controls

A smoothing penalty is designed to penalise non-smooth pulse shape, which would be all except flat pulses. However, the metric to penalise will do so more to rapidly

varying pulse shapes – those with a derivatives that have large changes near the vicinity of the point at which the derivative is evaluated.

Called *smoothing regularisation* in [23], a penalty of the following form is useful:

$$\mathcal{K} = \frac{1}{N} \|\hat{\Delta} \mathbf{c}\|^2 \quad (6.21)$$

where $\hat{\Delta}$ is an approximate differentiation, or second order differentiation operator of a suitable type and order, or any other appropriate transformation matrix. This penalty is based on EQ. (6.17) by replacing \mathbf{c} with $\hat{\Delta} \mathbf{c}$ and penalising the “high frequency” roughness instead of the high power.

The first and second derivatives of the penalty in EQ. (6.21) are

$$\nabla \mathcal{K} = \frac{2}{N} (\hat{\Delta}^\dagger \hat{\Delta} \mathbf{c}) \quad (6.22)$$

$$\nabla^2 \mathcal{K} = \frac{2}{N} (\hat{\Delta}^\dagger \hat{\Delta} \otimes \mathbf{1}) \quad (6.23)$$

where the identity matrix is of size K .

§ 6.3.1 Magnetisation-to-singlet transfer

The aim of examples presented here is to transfer polarisation to a singlet state, following the lead of the magnetisation-to-singlet (M2S) [190] and SLIC pulses [53,304], and improving this method with the use of optimal control pulses. Although robust optimal control pulses have already achieved this [178], the use of a quadratic smoothing penalty with a Newton-GRAPE method has not yet been realised.

Singlet states for a pair of mutually coupled spin- $1/2$ nuclei when the scalar coupling \gg difference in chemical shift resonances. The maximum fidelity of a system in a singlet state is

$$\begin{aligned} \max [\mathcal{J}_1] &= \max \left[\operatorname{Re} \langle \text{singlet state} | \rho(t) \rangle \right] = \frac{1}{\sqrt{2}} \\ \max [\mathcal{J}_2] &= \max \left[\left| \langle \text{singlet state} | \rho(t) \rangle \right|^2 \right] = \frac{1}{2} \end{aligned}$$

The SLIC pulse is essentially a hard pulse of an exact amplitude on either the x or y channel. The system consists of two ^{13}C spins with a chemical shift difference of 0.1 ppm and a J-coupling of 60 Hz. The magnetic field is set at 11.7434 tesla

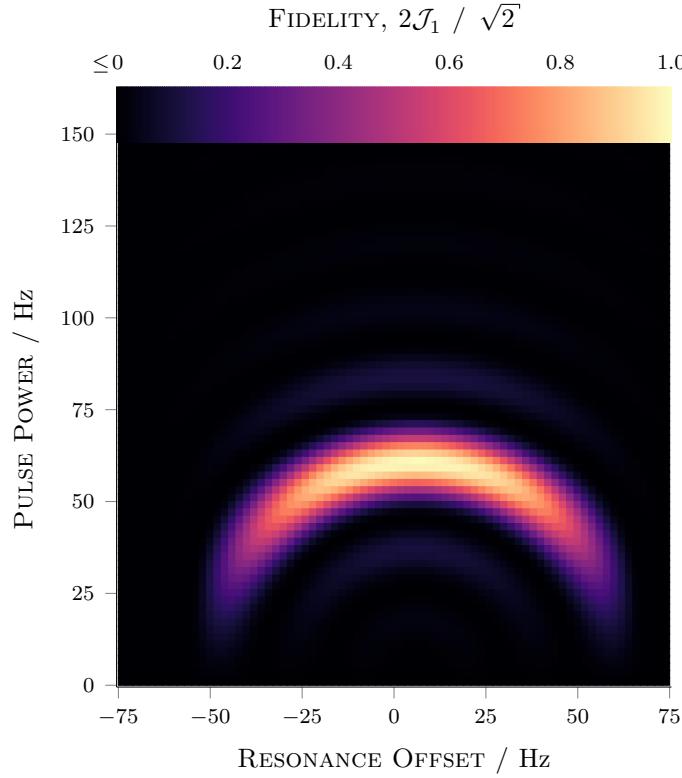


FIGURE 6.4: A plot of fidelity indicating the robustness of the SLIC pulse used as a direct comparison to optimal control pulses later in this section. The robustness is measured as a function of the nominal power level of the pulse amplitude in Hz on the y -axis, and the B_0 resonance offsets in Hz on the x -axis.

and the total pulse duration was calculated to be 56.25 ms. The aim of the SLIC pulse is to take longitudinal polarization \rightarrow 2-spin singlet state. Given these exact parameters, a Newton optimisation will find the SLIC pulse as the optimum.

The robustness of the SLIC pulses is shown in FIG. 6.4. *Good* pulses should be tolerant to power miscalibration and resonance offsets and large fidelities should be realised over a range of power levels and offsets. This is not really the case with the SLIC pulse – showing a band of robustness, but only a small area of high fidelity.

§ 6.3.2 Robust magnetisation-to-singlet transfer

Optimal control offers high fidelity state-to-state transfer and gives a shaped pulse which can be designed to be robust to B_0 resonance offsets and B_1 power miscalibration [178]. Simulation of this robustness involves running the GRAPE algorithm

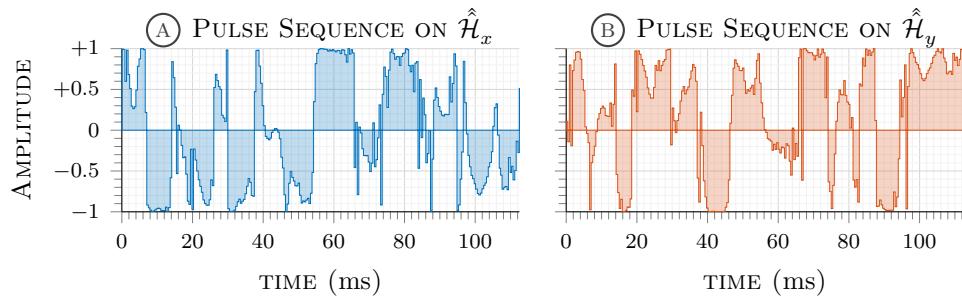


FIGURE 6.5: $\hat{\mathcal{H}}_x$ and $\hat{\mathcal{H}}_y$ components of an optimised, robust ℓ -BFGS-GRAPE pulse for a magnetisation-to-singlet transfer.

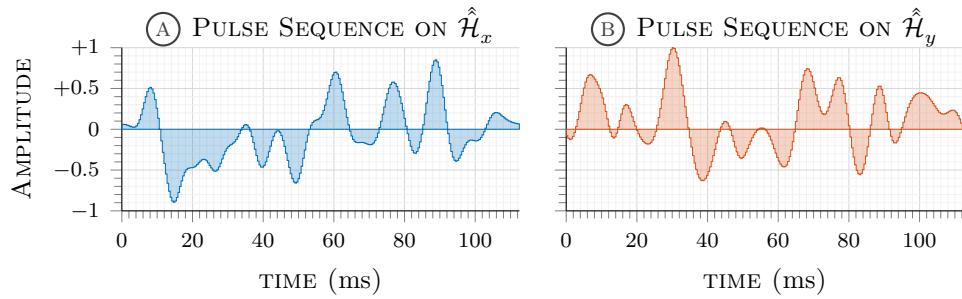


FIGURE 6.6: $\hat{\mathcal{H}}_x$ and $\hat{\mathcal{H}}_y$ components of an optimised, robust, smoothed ℓ -BFGS-GRAPE pulse for a magnetisation-to-singlet transfer.

over a grid of nominal power levels and offsets^{11} and averaged fidelities and gradients are then used by the ℓ -BFGS optimisation method.

This was done in [178], and is repeated here for comparison. To ensure there are no excursions from the power envelope, the optimisation is performed at a constant amplitude and variable phase, although the pulses shown in FIG. 6.5 have been transformed back to a Cartesian representation.

The robustness of these pulses is shown in FIG. 6.7, and it is quite clear that these pulses are far more robust to range of B_1 power and B_0 offsets than the SLIC pulse of FIG. 6.4. Not only is the pulse shape robust, it has near maximum fidelity over the robust range. A note should be made that the pulse duration is twice that of the SLIC pulse – testing shows that too short a duration too critical for optimisation – the SLIC pulse is the optimum for its calculated duration.

^{11}The resonance offset is simulated with an extra $\hat{\mathcal{H}}_z$ control at the offset power level in Hz.

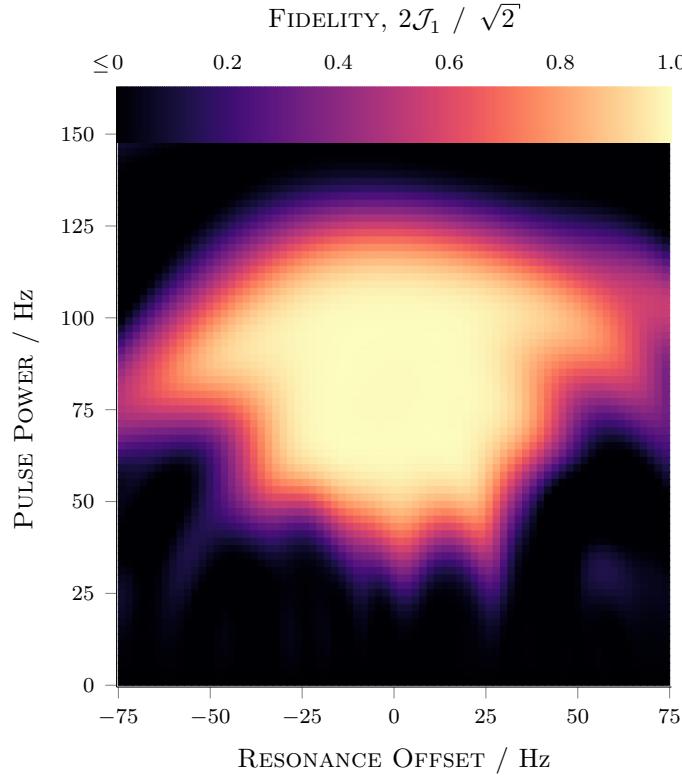


FIGURE 6.7: A plot of fidelity indicating the robustness of the optimised m2s pulses from FIG. 6.5. The robustness is measured as a function of the nominal power level of the pulse amplitude in Hz on the y -axis, and the B_0 resonance offsets in Hz on the x -axis.

§ 6.3.3 Smooth, robust magnetisation-to-singlet transfer

With the assumption that non-smooth pulses will suffer amplifier non-linearity, smoothed pulse shapes should perform better experimentally.

As was set out in §§4, 5, optimisation with a Newton-GRAPE method converges to a local maximum. The optimisation strategy is to first gain a quick solution to the state-transfer problem with a ℓ -BFGS-GRAPE method [86], robust to B_0 inhomogeneity, then to feed this solution into a quadratically convergent Newton-GRAPE method with a quadratic smoothing penalty of EQ. (6.21).

The pulses produced by this two-step optimisation are shown in FIG. 6.6 and are evidently smooth, much more smooth than in FIG. 6.5. As in FIG. 6.5, the pulse duration is twice that of the SLIC pulse for the same reason.

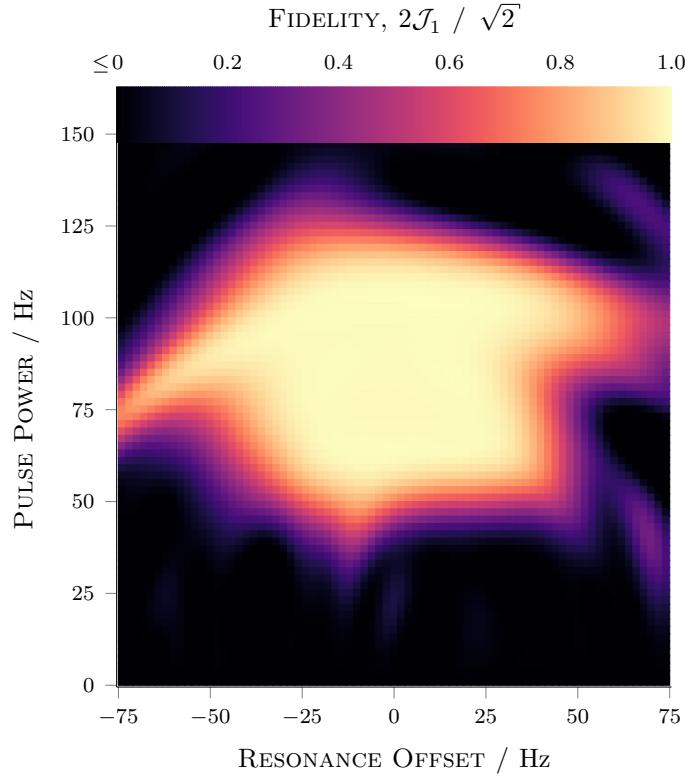


FIGURE 6.8: A plot of fidelity indicating the robustness of the smoothed optimised M2S pulses from FIG. 6.6. The robustness is measured as a function of the nominal power level of the pulse amplitude in Hz on the y -axis, and the B_0 resonance offsets in Hz on the x -axis.

The robustness of the pulse shape from FIG. 6.6 is shown in FIG. 6.8, showing very similar robustness to that of FIG. 6.7. It is expected that these pulses perform much better in experiments than both the SLIC pulse and the non-smoothed robust optimal pulses of FIG. 6.7.

The point of using the Newton-GRAPE method is that similar smoothed pulses are not converged to in the ℓ -BFGS-GRAPE method – the penalty competes with the fidelity measure during a line search and a pulse is found that is neither robust, as designed, nor smooth: finding a solution that is a trade-off between the two. The Newton-GRAPE method seems to be able to cope with this trade-off, identifying a place where the maximum fidelity and the minimum penalty coexist. One reason for this is that the Newton method uses regularisation and conditioning of §4.2.4, and is able to avoid saddle-points.

§ 7

Solid State Nuclear Magnetic Resonance

“I must not fear. Fear is the mind-killer. Fear is the little-death that brings total obliteration. I will face my fear. I will permit it to pass over me and through me. And when it has gone past I will turn the inner eye to see its path. Where the fear has gone there will be nothing. Only I will remain.”

– Frank Herbert, DUNE

Solid-state-NMR can be thought of as an ensemble of systems at different orientations, rotating relative to a laboratory frame. The rotation gives a periodically time-dependent Hamiltonian. A Lebedev quadrature [181, 182] can be used to characterise a surface integral over a 3D sphere, with weighted values at particular points on the surface (§7.1.1). This surface integral can be used to characterise an ensemble in the rotating frame in a Floquet space, [82, 282].

This chapter is divided into two sections. The first section is a comparison of the convergence of the Newton-GRAPE and ℓ -BFGS-GRAPE methods, simulating the optimal control problem over a Floquet space describing the crystalline orientations of a solid-state-NMR sample (§7.1). The success of the Newton-GRAPE method comes with a large expense and is set out in §7.1.2. The second outlines the description of the solid-state-NMR sample by a Fokker-Planck formalism from [68, 168] (§7.2), for a magic-angle-spinning experiment [231] §7.1.3 – using optimal control to excite the overtone transition in §7.2.1.

§ 7.1 Floquet Theory

The different orientation rotating relative to a laboratory frame gives periodically time-dependent coefficients of the Schrödinger equation [73, 188]:

$$\frac{d}{dt}\hat{\mathcal{U}}(t) = -i\hat{\mathcal{H}}(t)\hat{\mathcal{U}}(t) \quad (7.1)$$

suggesting the time evolution operator as a Fourier series expansion of the Hamiltonian [187]:

$$\hat{\mathcal{H}}(t) = \sum_n H_n e^{in\omega t} \quad (7.2)$$

where H_n are Hermitian matrices of periodic time-dependent functions. The solution to Eq. (7.1) is given in the following form [82, 282]:

$$\hat{\mathcal{U}}(t) = \hat{\mathcal{P}}(t)e^{-iQt}\hat{\mathcal{P}}^{-1}(t), \quad \hat{\mathcal{P}}(t) = \sum_n \hat{\mathcal{P}}_n e^{in\omega t} \quad (7.3)$$

This formulation allows a similar representation to be used when expanding the density matrix [320] from Eq. (2.52)

$$\rho(t) = \hat{\mathcal{U}}(t)\rho(0)\hat{\mathcal{U}}^{-1}(t) = \sum_n \rho_n(t)e^{in\omega t} \quad (7.4)$$

It is easy to see that $\hat{\mathcal{H}}(t + \tau) = \hat{\mathcal{H}}(t)e^{in\omega\tau}$, defining the periodic nature of the system. The diagonal elements of the diagonal matrix, Q , are called characteristic exponents [282]. Subtracting multiples of ω from the diagonal of Q can be compensated by multiplying $\hat{\mathcal{P}}(t)$ by $e^{ik\omega t}$.

The direct product of the basis states $\{|\rho_i\rangle\}$ with an infinite set of Fourier states $\{|n\rangle\}$, resulting from the Fourier expansion in Eq. (7.2), give the full basis set of states:

$$\{|\rho_i, n\rangle\} = \{|\rho_i\rangle\} \otimes \{|n\rangle\} \quad (7.5)$$

In Fourier space, the Hamiltonian and density matrix can be written as [320]

$$\hat{\mathcal{H}}_R(t) = \sum_n H_n \hat{F}_n e^{in\omega t} = e^{i\omega \hat{N}t} \left\{ \sum_n H_n \hat{F}_n \right\} e^{-i\omega \hat{N}t} \quad (7.6a)$$

$$\rho_R(t) = \sum_n \rho_n(t) \hat{F}_n e^{in\omega t} = e^{i\omega \hat{N}t} \left\{ \sum_n \rho_n(t) \hat{F}_n \right\} e^{-i\omega \hat{N}t} \quad (7.6b)$$

where \hat{F}_n are ladder operators in the Fourier space; $\langle v + n | \hat{F}_n | v \rangle = \mathbb{1}$, and \hat{N} is the number operator; $\langle v | \hat{N} | v' \rangle = v \delta_{vv'} \mathbb{1}$, and their commutation is $[\hat{N}, \hat{F}_n] = n \hat{F}_n$.

The Liouville-von Neumann equation Eq. (2.56) in Fourier space is

$$\frac{d}{dt}\rho_R(t) = -i[\hat{\mathcal{H}}_R(t), \rho_R(t)] \quad (7.7)$$

Using this Fourier representation of the Liouville-von Neumann equation, the interaction Hamiltonian becomes time independent [320]. This time independent Hamiltonian in Floquet space, $\hat{\mathcal{H}}_F$ is

$$\hat{\mathcal{H}}_Q = \sum_n H_n \hat{F}_n + \omega \hat{N} \quad (7.8)$$

giving the Floquet representation of the Liouville-von Neumann equation. This defines the evolution of the density operator;

$$\frac{d}{dt}\rho_Q(t) = -i[\hat{\mathcal{H}}_Q, \rho_Q(t)] \quad (7.9)$$

The Floquet density matrix can be evaluated with propagation over a time t using

$$\rho_Q(t) = U_Q(t)\rho_Q(0)U_Q^{-1}(t), \quad U_Q(t) = e^{-i\hat{H}_Q t} \quad (7.10)$$

Floquet theory has been successfully applied to describe solid-state-NMR [11, 268, 270]. It should be noted that Kuprov pointed out in [107] that the auxiliary matrix exponential formalism in Eq. (5.21) can be used to find solutions to rotating frame transformations without the usual method of nested commutators.

§ 7.1.1 Powder average

Various methods exist to discretise a sphere emulating the crystalline orientations of a powder average [12, 64, 245]; a grid uniformly distributed over a unit sphere used in solid-state-NMR simulations:

- ① A spiral circling the z -axis of a unit sphere [213].
- ② Finite triangular elements partitioning a unit sphere [323] (SOPHE).
- ③ Simulating many *subspectra*, then averaged to give the powder pattern [315] (SUMS).

α	β	w
0.000000	1.570796	0.066667
3.141593	1.570796	0.066667
1.570796	1.570796	0.066667
-1.570796	1.570796	0.066667
1.570796	0.000000	0.066667
1.570796	3.141593	0.066667
0.785398	0.955317	0.075000
0.785398	2.186276	0.075000
-0.785398	0.955317	0.075000
-0.785398	2.186276	0.075000
2.356194	0.955317	0.075000
2.356194	2.186276	0.075000
-2.356194	0.955317	0.075000
-2.356194	2.186276	0.075000

FIGURE 7.1: Orientations, α and β (radians), and weights w of the Lebedev grid quadrature.

- ④ Iteratively moving particles on the surface of a sphere under the influence of a repulsive force from other particles on the surface of a sphere [12] (REPULSION).
- ⑤ The use of Gaussian spherical quadrature [181,182] to weight the distribution over the surface of the sphere [65]

The following work uses the Lebedev quadrature to weight a distribution over the spherical grid [64,65]; the Euler angles and weights are shown in FIG. 7.1.

§ 7.1.2 Optimal control of a static powder average

With the Newton-GRAPE developed in §§4, 5, it seems a good place to start to test the method on a large system. As set out above, an optimisation problem involving an average over large grid will take many Hessian matrix evaluations, magnifying computational efficiency.

The system set out to test this is similar to the one in §4.3.2, except that the system is averaged over crystalline orientations. Optimal control of the powder average

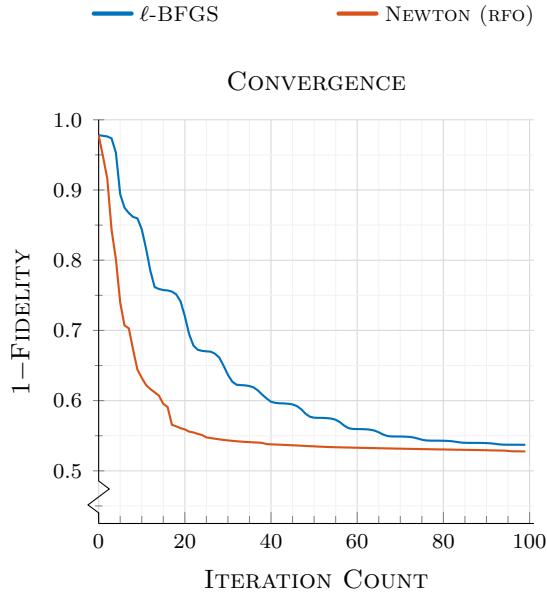


FIGURE 7.2: Comparison of optimisation methods for a ^{14}N quadrupolar overtone signal in static glycine powder. Two methods are compared: Newton-GRAPE method using a bracketing and sectioning line search; ℓ -BFGS-GRAPE using a bracketing and sectioning line search.

has a fidelity, gradients, and Hessians weighted over the crystalline orientations – rank 17 Lebedev grid with 110 points.

The optimal control problem is to excite ^{14}N from a state $\hat{T}_{1,0}$ to $\hat{T}_{2,2}$ (§A) through the nuclear quadrupolar interaction with appropriate control operators. The optimisation is performed over 400 time points for total pulse duration of 40 μs at a power level of 40 kHz.

At a particularly short total pulse duration, the optimal solution is not at a maximum fidelity = 1. The comparison of results for the convergence of ℓ -BFGS-GRAPE and Newton-GRAPE is shown in FIG. 7.2^{1}. Newton-GRAPE is quicker in the initial stages of the optimisation, as is expected compared to a quasi-Newton method, and has a similar convergence characteristic to the universal rotation pulses over a power miscalibration grid in §4.3.3. The terminal fidelity is also slightly better than that of the quasi-Newton method, again similar to the results of §4.3.3.

However, Newton-GRAPE took much longer (wall clock time) to reach a fidelity only a few percent better than the quasi-Newton approach – a couple of hours for

^{1}One convergence profile is shown – but convergence over the averaged grid gives very similar convergence from random starting pulses. It should be noted that the averaging has dictated a smooth convergence profile.

ℓ -BFGS-GRAPE to reach 100 iterations, and half a day for Newton-GRAPE to reach 100 iterations^{I}.

§ 7.1.3 Optimal control under magic angle spinning

Using Floquet space to describe the states of the solid-state-NMR system with a powder average, then to spin the system at the magic angle^{III}, is the test of optimal control set out in this section.

The powder average should be modelled by an appropriate Lebedev grid describing the crystalline orientations in the power. The *local fidelity* and the *local gradient* of the fidelity at each orientation of the grid to produce a global value of the fidelity and its gradient – the global measure is a weighted average over the crystalline grid. The local optimal control problem is performed with the ℓ -BFGS-GRAPE method.

Using the parameters set out in [136] for the overtone excitation in a glycine molecule, the optimal control problem here is a cross-polarisation experiment between ^{13}C and the quadrupolar ^{14}N . A spinning rate of frequency $f = 25 \times 10^3$ Hz is used to simulate the magic angle spinning experiment of [136], and the Fourier rank at which the Floquet matrix is truncated is 5 and the crystalline grid is averaged over a Lebedev grid rank 5 (α, β, γ) relative to the rotor frame (as in FIG. 7.1).

The initial state and final states of the system are the direct products with the identity matrix of size $(2 \times \text{max-rank}) + 1$, where $\text{max-rank} = 5$ in this case. This gives the optimal control boundaries as

$$\rho(0) = (\mathbb{1} \otimes L_z^{(\text{C})}) \quad (7.11\text{a})$$

$$\sigma = (\mathbb{1} \otimes L_z^{(\text{N})}) \quad (7.11\text{b})$$

where both states should be normalised. Control operators are in the Floquet formalism irradiating the resonant frequencies of ^{13}C and ^{14}N , again as a direct product with the identity matrix:

$$\hat{\mathcal{H}}_1 = \mathbb{1} \otimes \hat{\mathcal{H}}_x^{(\text{C})}, \quad \hat{\mathcal{H}}_2 = \mathbb{1} \otimes \hat{\mathcal{H}}_y^{(\text{C})}, \quad \hat{\mathcal{H}}_3 = \mathbb{1} \otimes \hat{\mathcal{H}}_x^{(\text{N})}, \quad \hat{\mathcal{H}}_4 = \mathbb{1} \otimes \hat{\mathcal{H}}_y^{(\text{N})} \quad (7.12)$$

^{I}Both methods used a 32 core computer parallelised calculations over the time steps.

^{III}the magic angle is $\theta_{\text{magic}} = \arccos \frac{1}{\sqrt{3}} = 54.7^\circ$

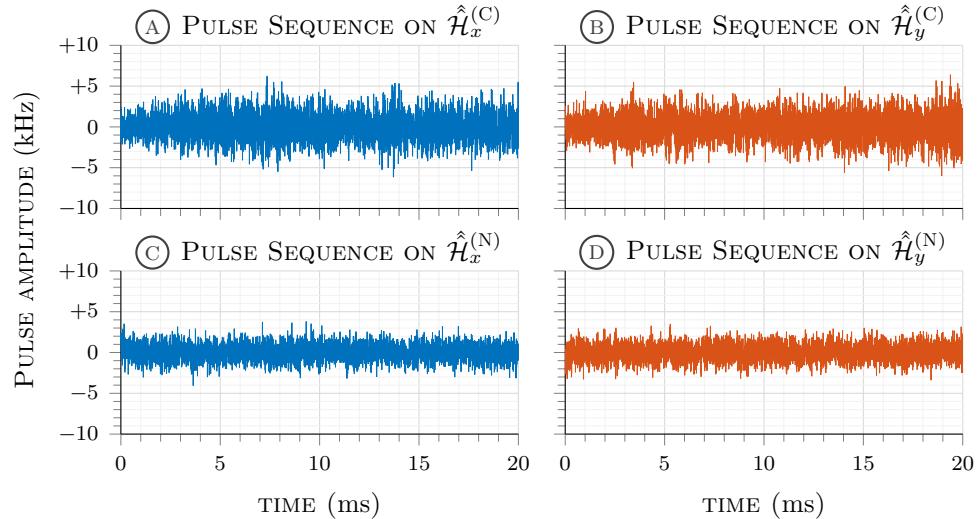


FIGURE 7.3: The four channel control pulse sequences producing the trajectories in FIG. 7.4. The pulses are a result of the ℓ -BFGS-GRAPE optimisation, which reached high fidelity, state-to-state transfer from ^{13}C to ^{14}N of glycine in a solid-state-NMR simulation.

with the nominal power of 10 kHz. Time steps are integer multiples of the characteristic time from the spinning rate, 5000 in this case, and the total pulse duration is 20 ms.

The ℓ -BFGS-GRAPE method is used; following from the long wall clock time of the Newton-GRAPE, the limited memory quasi-Newton method is more appropriate considering the large number of optimisation variables, 4×5000 . The optimisation was run for 256 iterations, although the optimisation had not fully converged at this point, a good set of results were obtained without waiting for super-linear convergence to find small gains.

The control pulses produced at the end of the 256 iterations are shown in FIG. 7.3 – looking like noise, apart from the ^{14}N pulses generally needing less power.

The trajectory analysis [167] of the pulses shows more useful information. As intended, the local population of the magnetisation is transferred from ^{13}C to ^{14}N in FIG. 7.4 (A), with a smooth transfer except for a small amplitude, high frequency oscillation on the ^{14}N trajectory.

The correlation orders in FIG. 7.4 (B) clearly show a 1-spin order giving way to the 2-spin order, to mediate the ^{13}C and ^{14}N coupling, then after about half of the pulse duration, the 2-spin order is transferred back to 1-spin order. There is an initial ramp in the 2-spin order and then a linear decay to zero at time = T .

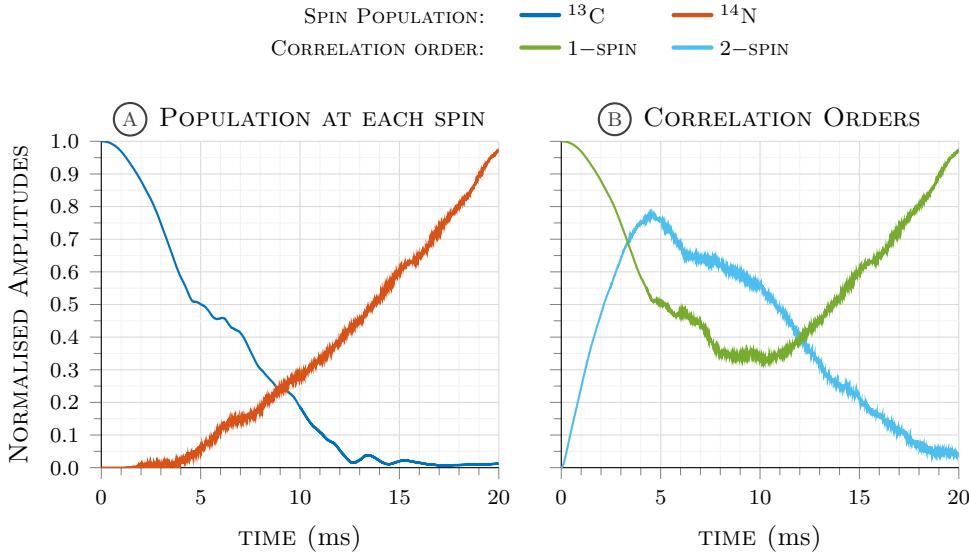


FIGURE 7.4: Trajectories analysis for a state-to-state transfer from ^{13}C to ^{14}N of glycine in a solid-state-NMR simulation. Averaged over the crystalline orientations using a Lebedev quadrature. 10 kHz power level of control pulses and 14.1 MHz magnet. (A) Local spin population. (B) Spin correlation order amplitude.

Again, the correlation orders develop a small amplitude, high frequency oscillation in the trajectories, this time after a maximum 2-spin order has been reached.

§ 7.2 The Fokker-Planck formalism

Kuprov [168] set out a review of the Fokker-Planck [83, 241] formalism used in magnetic resonance simulations, finding the very general formulation that should be preferred to the Liouville-von Neumann equation. For this reason, the optimal control methods set out in this thesis should be constructed to be compatible with the Fokker-Planck equation of motion. As pointed out in [168], there is a close relation between the Fokker-Planck equation and Floquet theory of the previous section [188] – the latter is a special case of the former.

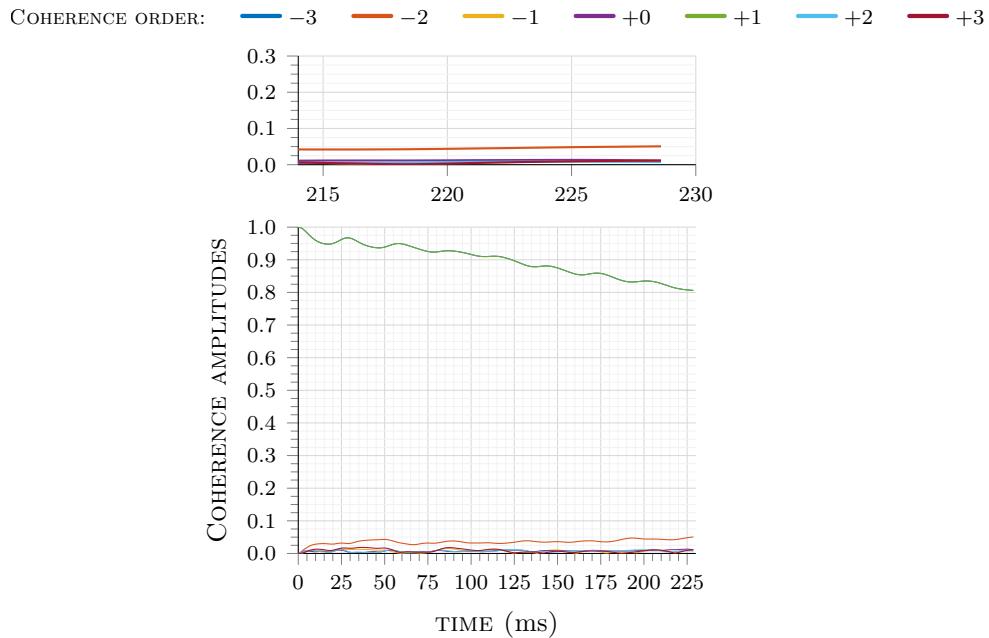


FIGURE 7.5: Trajectory analysis of an HMQC-type experiment to indirectly excite the overtone transition in glycine with hard pulses. Coherence orders are represented as the effect of a constant amplitude and zero phase. Magnified section of the plot shows higher coherence orders populated near the end of the control pulse trajectory (*upper*)

§ 7.2.1 Optimal Control in HMQC-type experiment

The testing set out in this section is an HMQC-type experiment (Heteronuclear Multiple Quantum Coherence) to indirectly excite the overtone transition in glycine [118] (results from hard pulses alone shown in FIG. 7.5). GRAPE calculations of optimal control in the Fokker-Planck formalism are used to find efficient pulses for an excitation of the +2 coherence order of ^{14}N in the HMQC-type experiment. Optimal pulses are designed to be resilient to radiofrequency pulse miscalibration of $\pm 2.5\%$. The total pulse length of the optimal set is investigated with respect to numerical convergence, maximum fidelity after 200 iterations, and coherence order trajectories.

The aim is to find optimal control pulses that spy on ^{14}N through echoes on a connected ^1H by exciting high coherence orders of ^{14}N . The optimal control pulses form part of a larger pulse sequence, shown in FIG. 7.6, where the second section of the optimal control pulses is the time reversed version of the first section.

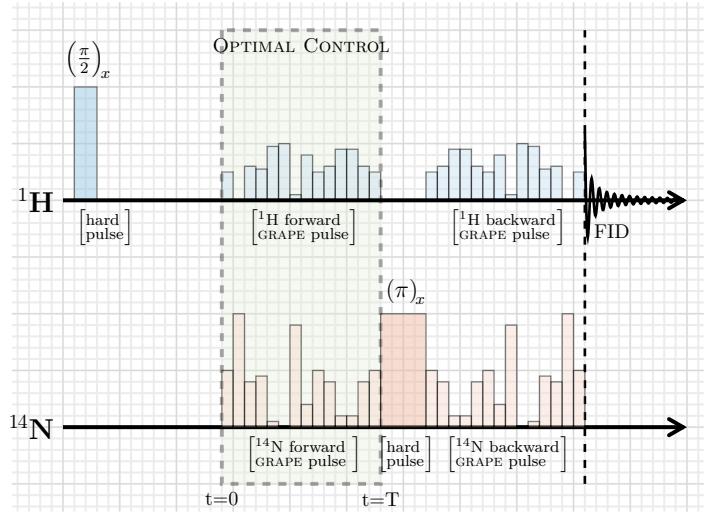


FIGURE 7.6: Pulse set including optimal control pulse. Pulses vary in phase only, and the amplitude should be considered constant. The second section of optimal pulses is the time-reversed version of the first section.

The spinning rate of the system is 35 kHz at the magic angle, in a 14.1 tesla magnet. Relaxation is included in the form of a diagonal approximation with a rate of 300 Hz, and the pulse duration is varied with different integer values of the rotor period. Optimal control of the initial state to a target state is performed as a sum of states picking out the +2 coherence excitation on the ^{14}N . The initial to target states are set out as

$$L_+^{(\text{H})} \xrightarrow{\text{GRAPE}} L_+^{(\text{H})} I_+^{(\text{N})} + L_-^{(\text{H})} I_+^{(\text{N})}$$

Control operators are defined in the Cartesian representation, allowing pulses on both ^{14}N and ^1H to work in the rotor frame at each of the weighted crystalline orientations in the rank-5 powder average:

$$\hat{\mathcal{H}}_x^{(\text{N})} = \mathbb{1} \otimes \left[\cos(\theta) \hat{\mathcal{H}}_z^{(\text{N})} + \sin(\theta) \left(\frac{\hat{\mathcal{H}}_+^{(\text{N})} + \hat{\mathcal{H}}_-^{(\text{N})}}{2} \right) \right] \quad (7.13\text{a})$$

$$\hat{\mathcal{H}}_y^{(\text{N})} = \mathbb{1} \otimes \left[\cos(\theta) \hat{\mathcal{H}}_z^{(\text{N})} + \sin(\theta) \left(\frac{\hat{\mathcal{H}}_+^{(\text{N})} - \hat{\mathcal{H}}_-^{(\text{N})}}{2i} \right) \right] \quad (7.13\text{b})$$

$$\hat{\mathcal{H}}_x^{(\text{H})} = \mathbb{1} \otimes \left[\cos(\theta) \hat{\mathcal{H}}_z^{(\text{H})} + \sin(\theta) \left(\frac{\hat{\mathcal{H}}_+^{(\text{H})} + \hat{\mathcal{H}}_-^{(\text{H})}}{2} \right) \right] \quad (7.13\text{c})$$

$$\hat{\mathcal{H}}_y^{(\text{H})} = \mathbb{1} \otimes \left[\cos(\theta) \hat{\mathcal{H}}_z^{(\text{H})} + \sin(\theta) \left(\frac{\hat{\mathcal{H}}_+^{(\text{H})} - \hat{\mathcal{H}}_-^{(\text{H})}}{2i} \right) \right] \quad (7.13\text{d})$$

Amplitude (polar) (kHz)	total time (μ s)	pulses (#)	∇ calc. (s)	% state transfer initial state = $\{L_+^{(H)}\}$ targets = $\{L_+^{(H)}L_+^{(N)}, L_-^{(H)}L_+^{(N)}\}$
spin rate = 35 kHz				
$\sqrt{2} \times 50$	57.143	200	47	7.89 %
$\sqrt{2} \times 50$	114.286	400	92	15.86 %
$\sqrt{2} \times 50$	171.429	600	140	18.67 %
$\sqrt{2} \times 50$	228.571	800	189	21.57 %

FIGURE 7.7: Simulation performance of each of four test optimisations: 2, 4, 6, and 8 rotor periods, each with 100 time slices per rotor period. Note: results are not converged and a greater fidelity would be achieved if the optimisation were allowed to run for more than 200 iterations, particularly for the longer pulse sets (see FIG. 7.10)

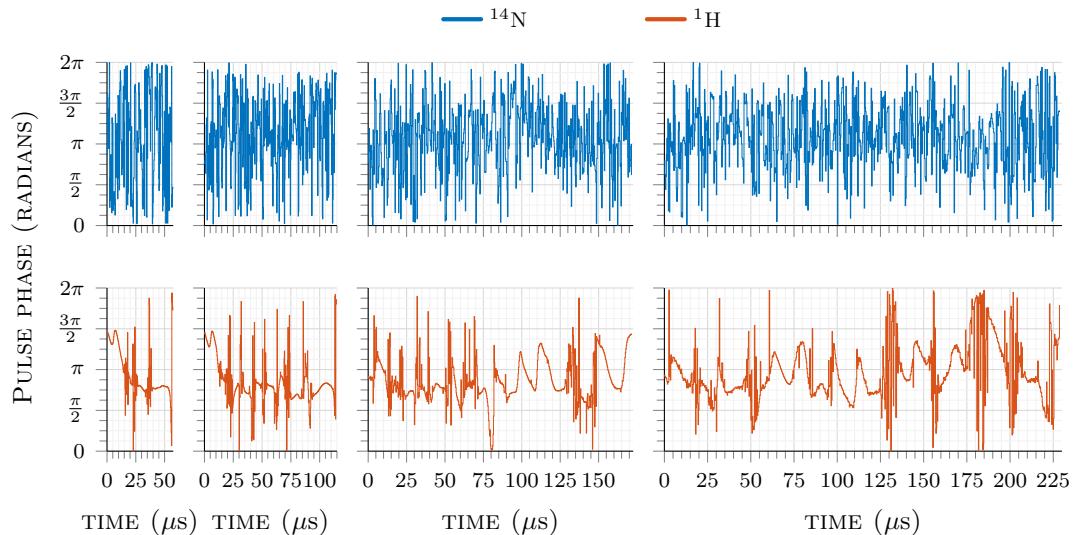


FIGURE 7.8: Phase profile resulting from the ℓ -BFGS-GRAPE method with 2, 4, 6, and 8 rotor period for total pulse duration (*left to right*). The amplitude profile is constant in all cases, $A = \sqrt{2} \cdot 50$ kHz, and the phase pulses on ^{14}N and ^1H (*upper and middle*) should be tolerant to a $\pm 2.5\%$ miscalibration on the pulse amplitudes (when translated to a Cartesian representation). The total density of magnetisation at each spin is shown as an effect of the pulses (*lower*).

where θ is the magic angle of the rotor spinning axis and $\mathbb{1}$ is the identity operator of size d , being the size of the spatial dimension of the Fokker-Planck formalism.

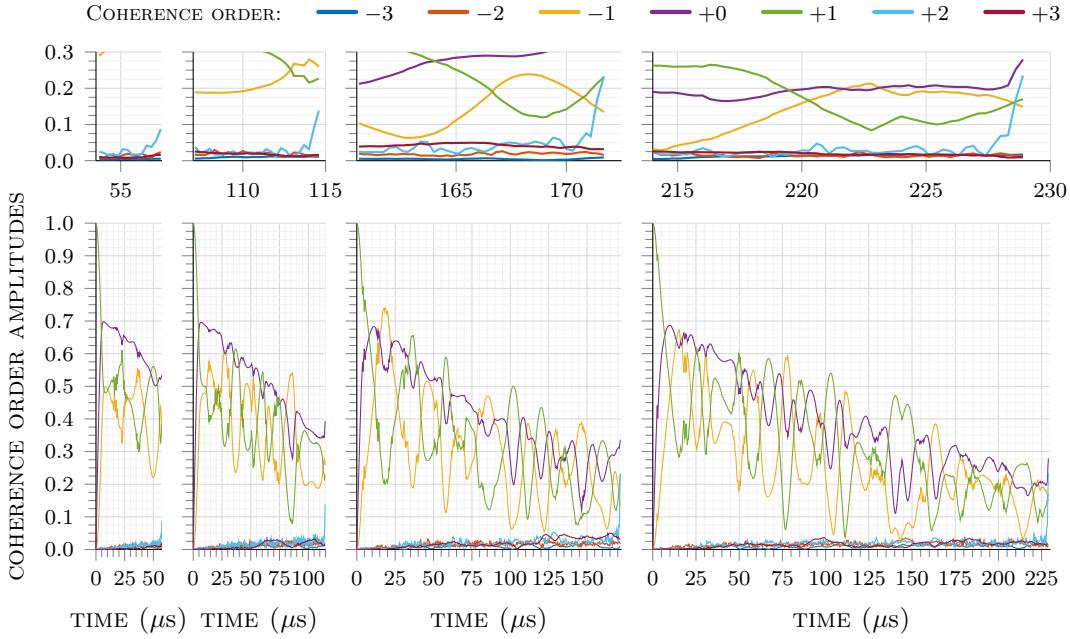


FIGURE 7.9: Trajectory analysis of an l-BFGS optimisation with 2, 4, 6, and 8 rotor periods for the total pulse duration (*left to right*). Coherence orders are represented as the effect of phase profiles shown in FIG. 7.8. Magnified section of the plot shows higher coherence orders populated near the end of the control pulse trajectory (*upper*). NOTE: Trajectories are the last trajectory calculated i.e. that at +2.5% pulse miscalibration and for the target state $L_-^{(H)} L_+^{(N)}$.

Normalised initial states and target states are similarly defined via the Kronecker product:

$$\rho(0) = \frac{1}{\sqrt{d}} (\mathbb{1} \otimes L_+^{(H)}) \quad (7.14a)$$

$$\rho_1(T) = \frac{1}{\sqrt{d}} (\mathbb{1} \otimes L_+^{(H)} L_+^{(N)}) \quad (7.14b)$$

$$\rho_2(T) = \frac{1}{\sqrt{d}} (\mathbb{1} \otimes L_-^{(H)} L_+^{(N)}) \quad (7.14c)$$

where $\mathbb{1}$ a unity vector of length d , $\rho(0)$ is the initial state at time $t = 0$, $\rho_1(T)$ and $\rho_2(T)$ are the two target states at the time $t = T$. The ℓ -BFGS-GRAPE method is used, and performs two fidelity and gradient calculations from the initial state to the two final states, then to average the two fidelities and two gradients. The full experimental state transfer problem can be envisaged as the pulse set being part of the whole experimental pulse sequence in FIG. 7.6.

Four optimisation tests were performed with the only difference being the total pulse time allowed for the control pulses. The time step remained equal in all

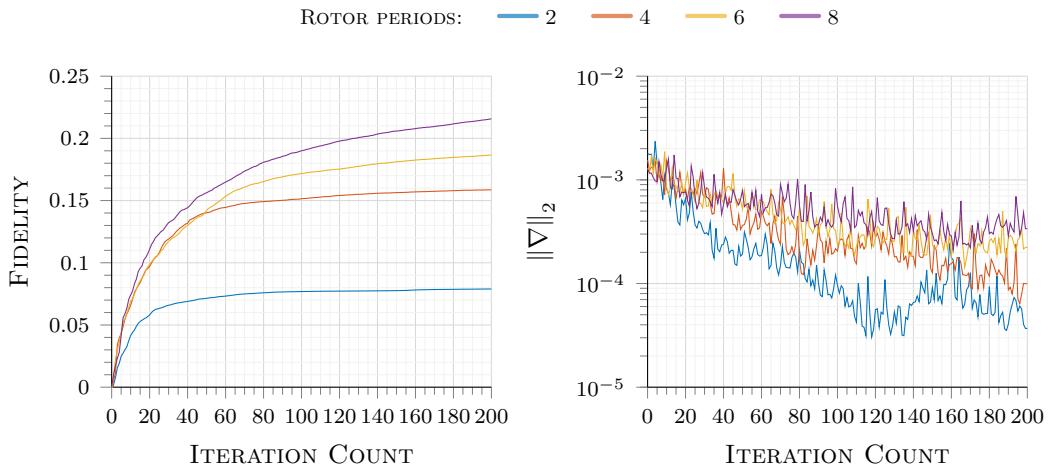


FIGURE 7.10: Convergence characteristics. Note that the optimisations have not yet achieved convergence to a maximum (apart from the optimisation with 2 rotor periods).

cases, having 100 time steps per rotor period (35 kHz spinning rate). The resulting fidelity from each test is shown in the FIG. 7.7 (final column).

The polar amplitude was kept to a constant, $\sqrt{2} \cdot 50$ kHz ^{I}, and only the phase profiles of pulses on each spin were varied. The phase profiles resulting from the optimisation are shown in FIG. 7.8. In addition, the coherence pathways for these pulses are shown in FIG. 7.9 – this is the main indication of the success of the simulations in comparison to the hard pulses of FIG. 7.5.

The four tests were run to 200 iterations, rather than a converged set of solutions. FIG. 7.10 shows how far each of the test optimisations converged. In addition to showing the final fidelity reached at 200 iterations, the norm of the gradient, $\|\nabla\|_2$, is shown at each iteration – the norm of the gradient is a measure of how optimal the solution is ^{II}.

^{I}corresponding to ± 50 kHz maximum/minimum pulse amplitude in the Cartesian representation

^{II}a maximum or minimum proper would have $\nabla = 0$ and hence $\|\nabla\|_2 = 0$ – the main point is the smaller this number, the more optimal the solution

§ 8

Feedback Control

“Most men will not swim before they are able to. Is that not witty? Naturally, they won’t swim! They are born for the solid earth, not for the water. And naturally they won’t think. They are made for life, not for thought. Yes, and he who thinks, what’s more, he who makes thought his business, he may go far in it, but he has bartered the solid earth for the water all the same, and one day he will drown.”

– Hermann Hesse, STEPPENWOLF

A significant current problem in high-field electron spin resonance (ESR) spectroscopy is the difficulty of achieving uniform and quantitative signal excitation using microwave pulses [33,279]. The greatest instrumentally available microwave power, in terms of the electron spin nutation frequency, is about 50 MHz; the shortest realistic $\pi/2$ pulse is therefore 5 ns long [44], corresponding to the excitation bandwidth of 200 MHz – enough to affect a significant portion of many solid state ESR signals, but insufficient to excite such signals uniformly and quantitatively. The consequences of partial excitation of the spectrum include useful orientation selection effects [19, 242, 265], but also reduced sensitivity and diminished modulation depth in two-electron dipolar spectroscopy [258, 299, 305].

The time resolution of the best available microwave pulse shaping equipment is of the order of 30 ps ^{II}. This work uses Bruker SpinJet AWG (Arbitrary Waveform Generator) with 0.625 ns time resolution §B.1 – in combination with non-resonant cavities it enables generation of shaped pulses with the bandwidth of about 1 GHz [44] and allows many broadband excitation schemes originally developed for nuclear magnetic resonance (NMR) spectroscopy [331] to be used with

^{II}Keysight M8196A 92 GSa/s Arbitrary Waveform Generator

only minor modifications [57, 58, 137, 272, 292]. Numerically designed “optimal control” microwave pulses [86, 108, 154] are also possible [145, 293], but a complication specific to ESR is that the waveforms received by the sample are very different from those sent by the AWG – the response function of the ESR instrument cannot be ignored [293].

One way around this is to construct a transfer matrix or a response function that connects, under the linear response approximation, the ideal pulse emitted by the computer to the real pulse seen by the sample. The transfer matrix may be measured either by adding an antenna to the resonator [293], or by using a sample with a narrow ESR line to pick up the intensity of each spectral component [145]. Quasi-linear responses, such as the phase variation across the excitation bandwidth in nutation frequency experiments, can be described with additional transfer matrices [58].

The transfer matrix approach is not perfect – different samples alter the dynamical properties of the resonator in different ways, as does the antenna – but the linear and quasi-linear models work well in practice. The standard procedure is to take the desired pulse, reverse-distort it through the transfer matrices, send the result out of the waveform generator and hope that a good rendering of the intended pulse shape arrives at the sample point. It usually does [145, 293], but the logistical overhead of measuring the transfer matrix is significant. Accurate measurement of the instrument response function in the ways described above is time-consuming (hours), and ESR resonators, particularly at high frequencies, tend to have strongly sample-dependent response functions.

In this chapter, a different microwave pulse shape refinement strategy is explored, which does not require explicit knowledge of the transfer matrix §8.1. It relies instead on the possibility of repeating an ESR experiment hundreds of times per second, and recognises the fact, discussed in detail below, that microwave pulses in ESR need very few discretisation points due to the significant width of the instrument response function. Further details of this work are in §B, including software code and experimental results.

Numerically optimised microwave pulses are used to increase excitation efficiency and modulation depth in electron spin resonance experiments performed on a spectrometer equipped with an arbitrary waveform generator (§8.1.1). The optimisation procedure is sample-specific and reminiscent of the magnet shimming process used in the early days of nuclear magnetic resonance – an objective function (for

example, echo integral in a spin echo experiment) is defined and optimised numerically as a function of the pulse waveform vector using noise-resilient gradient-free methods (§8.1.2). The resulting shaped microwave pulses achieve higher excitation bandwidth and better echo modulation depth than hard pulses. Although the method is theoretically less sophisticated than quantum optimal control techniques, the rapid electron relaxation means that the optimisation takes only a few seconds, and the knowledge of either the instrument response function or spin system ensemble parameters is not required. This makes the procedure fast, convenient, and easy to use. Improvements in broadband excitation efficiency, spin echo intensity [117] in §8.2.1 and modulation depth are demonstrated with optimal conditions of OOP-ESEEM [306] in §8.2.2, at the instrument time cost not exceeding the time it used to take to auto-shim an NMR magnet.

The method is known as “feedback control” [60, 87, 290]; in its electron spin resonance adaptation it is similar in principle to the well known (in the NMR circles) task of maximising a deuterium lock signal during the magnet shimming process [52] – a target variable is chosen and maximised, using a noise-resilient optimisation algorithm, with respect to the variables of interest. In the ESR case, these variables are amplitudes of the microwave field at each waveform discretisation point.

§ 8.1 Feedback Control Optimisation

One simple way to iteratively improve the result of experimental measurements is to use feedback control [60, 87, 290]. The idea is to modify inputs to the experiment according to the result of an experimental measurement: controlling the experiment “on the fly”. The Bruker SpinJet arbitrary waveform generator (AWG) has the ability to shape pulses of the ESR experiment – giving a flexible control method. However, distortions to the pulse shape are expected in two forms – constant distortions that are the same for repeated measurements, such as shape pulse interpolation, and those associated with non-linear effects such as noise from electronic components and sample specific miscalibration. FIG. 8.1 sets out a block diagram of a closed-loop feedback control using an AWG.

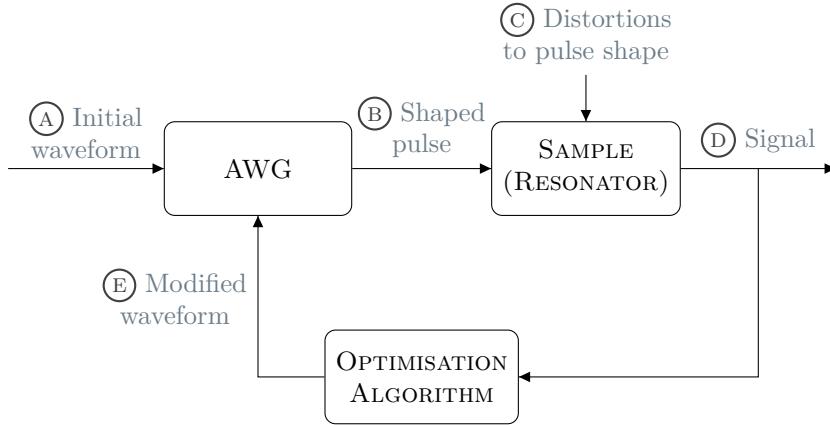


FIGURE 8.1: Block diagram of AWG closed-loop feedback control. (A) An initial, random, pulse set sent to the AWG. (B) Pulse-set is processed and interpolated, then sent to the sample (resonator). (C) The pulse-set arriving at the sample experiences shape distortions from system noise and hardware limitations. (D) The effect of pulse-set induced excitations on the sample is measured as a signal. (E) The optimisation algorithm calculates a new, modified set of pulses that are sent back to the AWG.

§ 8.1.1 Hardware and Software

The Bruker SpinJet AWG is a two channel arbitrary waveform generator, allowing the input of in-phase and quadrature components of shaped pulses to a spectrometer (FIG. B.1). The Bruker SpinJet AWG used in this work has a time resolution of 0.625 ns, 14-bit amplitude resolution, 1.6 GS/s sampling rate, and ± 400 MHz bandwidth around the carrier frequency. The Bruker EleXSys E580 ESR spectrometer has a 2 ns time base and, in combination with the AWG, resolves the time resolution mismatch by downsampling the pulse waveform onto a 1 ns increment time grid.

The software used in this work was written in-house, and has a flow of communication between *Spinach* [130] and XEPR Python libraries, shown in FIG. 8.2. The master process runs in *Matlab* and calls XEPR Python functions as necessary to control the instrument. Experimental data is written by XEPR into plain text files and subsequently parsed by *Matlab*. Optimisation restart capability is implemented using an MD5 hash table of the previously submitted experimental settings and outcomes [108] – an interrupted optimisation can therefore retrace its steps quickly without re-running previously executed experiments.

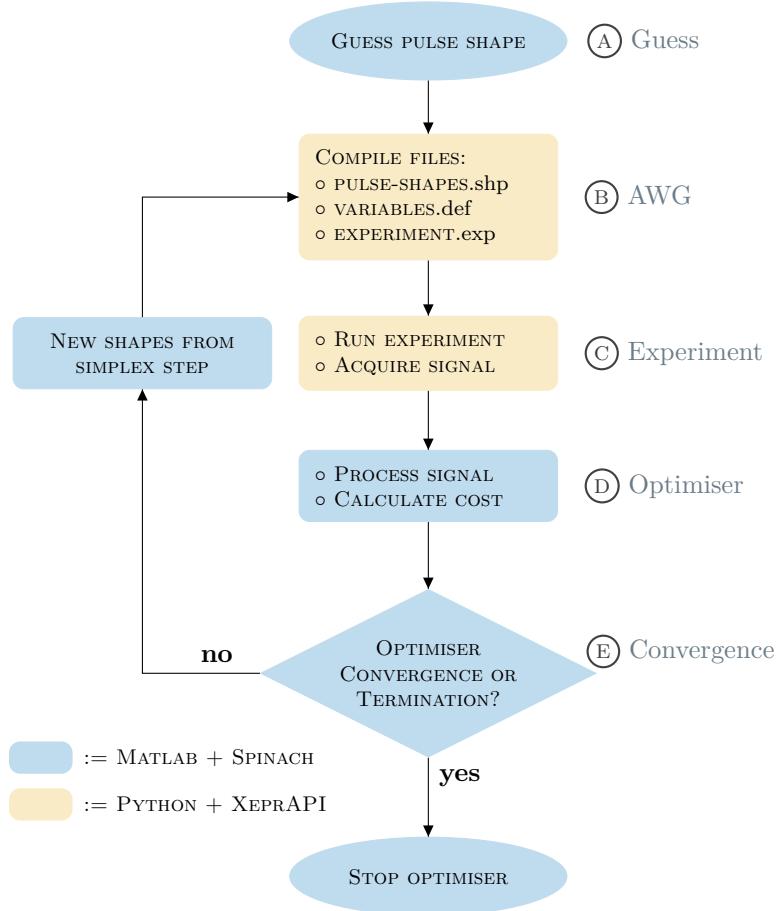


FIGURE 8.2: Software flow diagram. (A) Set initial pulses. A good guess of an ideal pulse set based on prior knowledge of the sample. (B) 1) Send this pulse-set to the AWG using PulseSPEL: Load, show, and compile shape file. 2) Modify PulseSPEL variables and compile. 3) Validate and compile PulseSPEL program. (C) Run experiment and wait for experiment completion, then read signal data. (D) Interpret data in optimiser. (E) If the optimisation algorithm hasn't converged, calculate a new, modified set of pulses that are sent back to the AWG. Python code is detailed in §B.2.

The simplex optimiser allows both Nelder-Mead [171, 220] and multidirectional search [50, 308] methods, with the option of evaluating independent sets of objective variables in parallel [125].

Many shapes can be stored within the AWG in the form of a plain-text shape-file, limited to 262,144 bytes of physical disk size. A limit of 115 consecutive shape-file compile events must be enforced to avoid overload of the AWG/console shared memory. Considering this last point, and that there may be many independent shapes involved in performing an optimisation iteration – it will become advantageous to perform a 2D experiment with these pre-calculated, shapes.

Two primary sources of waveform distortions may be identified: the static ones, introduced by the instrument electronics (pulse shape interpolation, hardware response function, *etc.*), and those associated with transient effects, *e.g.* electrical noise and sample-specific magnetic susceptibility effects. The presence and the transient nature of these distortions makes common ESR objective functions (signal integral, modulation depth, *etc.*) impossible to differentiate numerically and necessitates the use of gradient-free noise-resilient optimisation methods.

§ 8.1.2 Optimisation Method

An optimisation problem usually attempts to find extrema of an objective function [79, 96, 229] – in financial terms these extrema could be the minimum of a cost function, or a maximum of a profit function. This function should map a set of variables to a scalar number designed as a convenient metric characterising the “performance” of a solution formed from those variables.

The intensity of an echo makes a good echo experiment [33, 279] and the integral of the in-phase part of the echo signal makes a convenient metric to maximise. This is defined as the 1-norm of the signal plot vector [60]:

$$\|\vec{s}\|_1 := \int_0^T |s(t)| dt \quad (8.1)$$

Any optimisation should be properly scaled [79], and scaling for these experiments simply involves normalising the objective metric with the signal produced by a reference experiment; in this case, an experiment with conventional hard pulses. This gives the metric of the quality of an echo as

$$J(\mathbf{c}) = \frac{\|\vec{s}\|_1}{\|\vec{r}\|_1} \quad (8.2)$$

where $J(\mathbf{c})$ is the objective measure to maximise, \vec{s} is the signal measured within an objective function and \vec{r} is the reference signal measured from an experiment with hard pulses. $J(\mathbf{c}) > 1$ is defined as a good echo, better than that from hard pulses alone, and $0 < J(\mathbf{c}) < 1$ is defined as a bad echo, worse than that from hard pulses.

There is no obvious gradient with respect to the control parameters when considering an experimental measurement. Gradient-free optimisation strategies [26, 79,

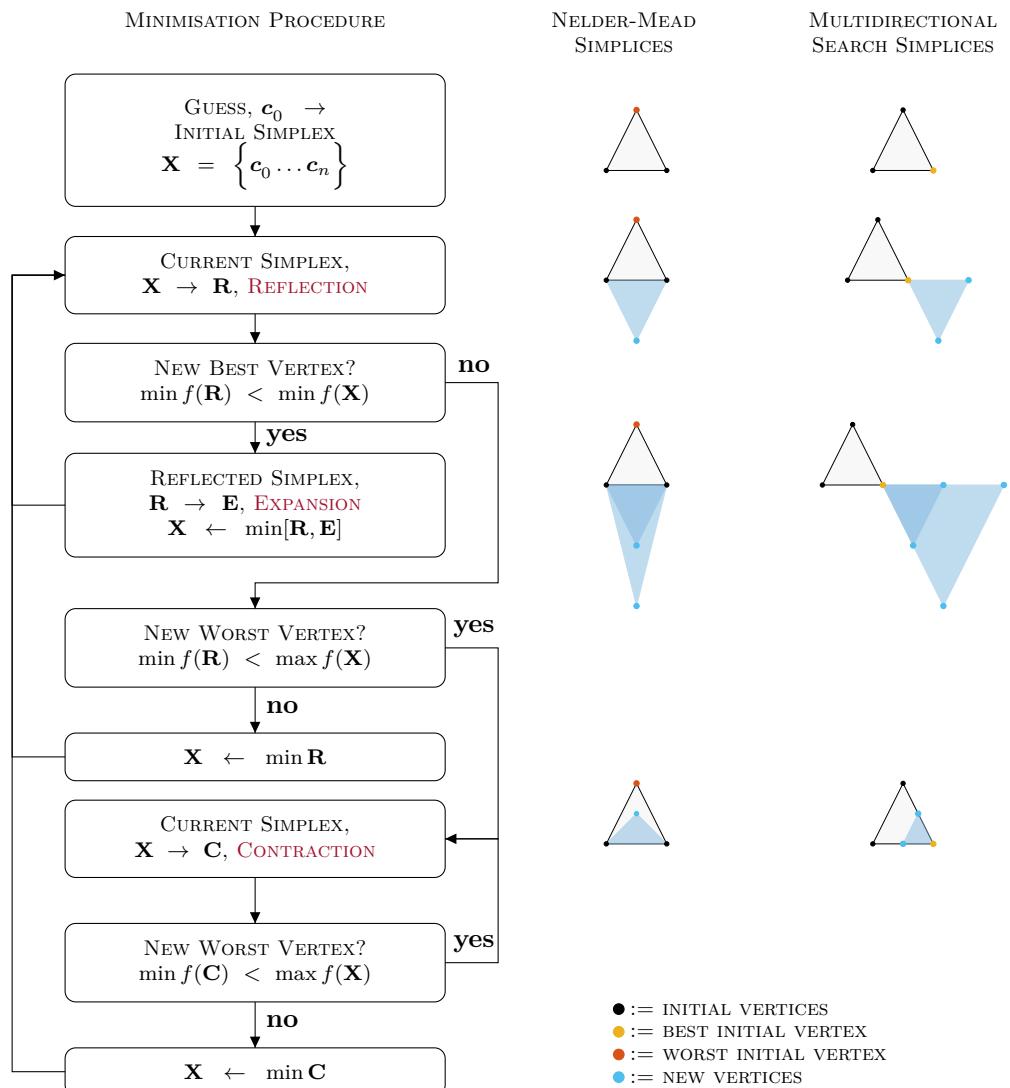


FIGURE 8.3: Different simplex steps for the Nelder-Mead and multidirectional search algorithms, each simplex has $n + 1$ vertices where n is the number of variables. Faces of the initial simplex are scaled to $|c_0|_\infty$, and form an equilateral simplex. Expansion and contraction factors set to $\beta = 2$ and $\gamma = 0.5$ respectively. The best/worst vertex and the centroid $c_c = \sum_0^n c_j$ of the simplex are used when reflecting, expanding, and contracting.

96, 229, 297] are the only choice for this real-time optimisation. A simple Nelder-Mead algorithm [171, 220] (a member of a family of direct search methods termed simplex methods [297], polytope methods [96] and ad-hoc methods [79]) has the benefit of a relatively small number of experimental evaluations for a single iteration compared to other popular algorithms e.g. genetic algorithms [28, 142] or simulated annealing [210]. However, convergence is not guaranteed and is linear at best [51]. A desirable benefit of the Nelder-Mead algorithm is its tolerance to random noise (on a smooth function) [22, 97]. In addition; a *restart* step has been shown to alleviate stagnation of simplex algorithms [148].

The shape file characterising the optimisation variables has the constraint that it can only contain values in the range $[-1, +1]$ and these constraints can complicate the formulation considered in the optimisation problem. However, gradients are not considered in for the optimisation problem stated in this work, and a transform can easily classify the bounded pulse-shape variables [79]: a repeating, mirrored transform can allow all real numbers to be evaluated by the objective function in a continuous manner. This waveform constraint transform can be classified as in Eq. (6.2):

$$\tilde{\mathbf{c}} = \left| \mathbf{c} - 4 \left\lfloor \frac{\mathbf{c} - 1}{4} \right\rfloor - 3 \right| - 1$$

where \mathbf{c} is the unbounded waveform and $\mathbf{c} - n \lfloor \mathbf{c}/n \rfloor$ is the remainder from dividing \mathbf{c} by n .

A modified Nelder-Mead simplex algorithm, the multidirectional search simplex method [50, 125, 308], has an efficient characteristic in that it can be used as a partially parallel simplex method, essentially creating a search grid of simplices. The added computational cost is that many more function evaluations must be made per simplex, depicted in FIG. 8.3. This multidirectional search algorithm has a comparable performance to finite-difference gradient algorithms and is designed to be tolerant to problems with many local minima [97]. A further useful property of the multidirectional search algorithm is that it has guaranteed convergence [307], similar to gradient following algorithms. The common algorithmic steps of the Nelder-Mead and multidirectional search algorithms are shown in FIG. 8.3.

To improve stability of the optimisation process, the pulse shape variables are stored in a database using an MD5 hash function during the optimisation iterations [108] – in the event of the optimisation algorithm halting mid-algorithm due to unexpected hardware incidents, the optimisation can effectively be resumed from the same stage after hardware problems have been resolved.

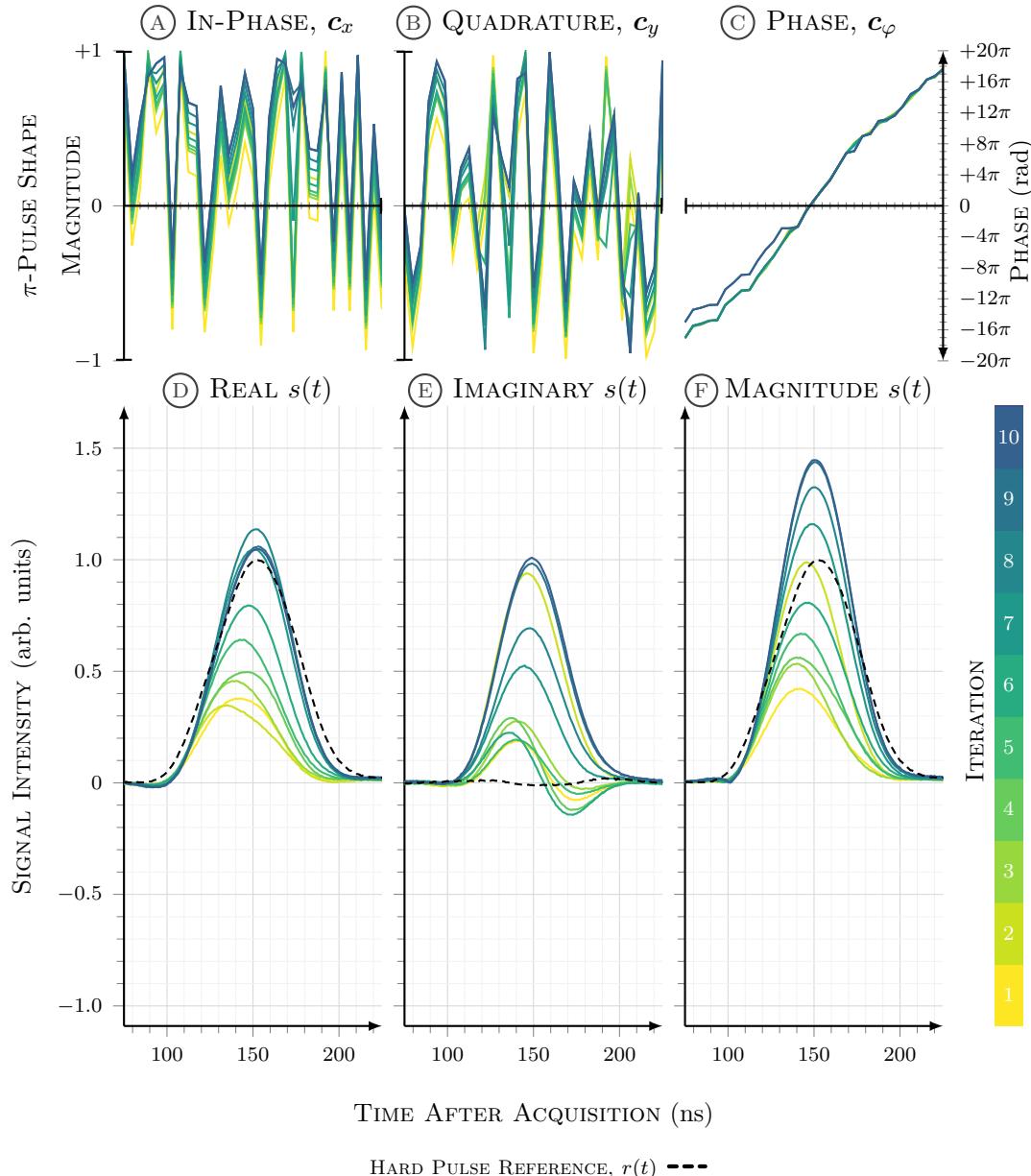


FIGURE 8.4: (A) In-phase, (B) quadrature, and (C) phase parts of the pulse shape at each iteration. The spin echo experiment performed on a nitroxide radical and the optimisation variables are 33 discrete shape points of a 32 ns π -pulse. (D) Real, (E) imaginary, and (F) magnitude of the measured signal at each iteration.

§ 8.2 Experimental Application

§ 8.2.1 Feedback Optimised Hahn Echo

A two-pulse Hahn echo [117,254] is a good test for this closed-loop feedback control, using the 1-norm of the echo signal as a figure of merit, as in Eq. (8.1). An optimised 32 ns π -pulse is the objective of the test, using discretised in-phase and quadrature components of a shaped waveform. In emulating the time resolution of the spectrometer, a shape with 33 discrete points should be used as the finest grid; one point at the start of every time point plus one point at the end of the last time point.

The optimisation starts at a random guess within the allowed bounds, and proceeds to search for shapes that produce better echoes. This first test of the feedback control methods uses the Nelder-Mead simplex algorithm, with termination conditions of a simplex size less than 0.05, or more than 100 consecutive contraction steps without finding a better solution (indicative of stagnation of the algorithm). The echoes produced at each of the optimisation iterations are shown in FIG. 8.4.

The results of the feedback control optimisation are not encouraging for these pulse shapes §B.3. The most obvious failure is that the integral of the in-phase part of the echo does not improve on that from hard pulses. The optimisation is considered converged to a local maximum, evident in FIGS. B.2, B.3, the latter part of the optimisation makes many functional evaluations with very little gain. This is typical of a simplex optimisation which tries many contraction steps before an improved solution can be found (see FIG. 8.3). There are two other characteristics of the optimisation:

- ① The quadrature part of the spin echo indicates an echo without the correct phase calibration.
- ② The pulse shape does not change by an appreciable amount during the optimisation.

The first of these characteristics is not so much of a concern: a phase calibration can be performed after the optimisation and can only give a larger in-phase part of the echo integral than that shown in FIG. 8.4 ①. The second characteristic

may have two possible reasons: either the variables for the optimisation are not normalised for an effective optimisation, making too small steps at each iteration, or there are just too many variables for the optimiser to effectively handle. The Nelder-Mead algorithm is known to be effective only for low dimensional problems [171] – there is no gradient to direct the algorithm and it can only be a sophisticated ad-hoc search [79]. Directed by the limits of the Nelder-Mead algorithm, further tests of lower dimension are required.

Two further tests are performed, one with approximately $\frac{2}{3}$ of the number of time points and one with approximately $\frac{1}{3}$ of the time points. In both cases the total pulse duration remains the same. The first of these is a 21 point shape and the results at each iteration of the Nelder-Mead algorithm are shown in FIG. 8.5.

Results are more encouraging for a pulse shape using 21 discrete points §B.3 – achieving an echo integral 27% better than that of hard pulses (see FIG. B.3). However, this optimisation makes many function evaluations giving a total optimisation time of more than 20 minutes (much longer than an experienced spectroscopist would take to find a good echo signal). The majority of the function evaluations are from inner contraction steps of the simplex algorithm – making smaller and smaller simplices with little gain. In fact, this optimisation terminates because the relative simplex size was less than the defined numerical tolerance. This optimisation can be considered stagnated within the noise level of the signal.

The final feedback controlled spin echo experiment shows a further insight into a hidden sophistication of the optimisation algorithm, when used within its limits. Again reducing the amount of dimensions of the optimisation problem, to approximately $\frac{1}{3}$ of the time points, a shape with 11 discrete points is used as the variables to the Nelder-Mead algorithm.

These results, shown in FIG. 8.6 and FIG. B.3, produce a good echo integral in a reasonable amount of time (6-7 minutes). They are considered converged for the same reason as the 22 point pulse shape – indicating that this number of variables is low enough for the Nelder-Mead algorithm to function effectively.

A final note, which leads indirectly to the next section of results, is that the variation of the quadrature part of the echo in FIG. 8.6 covers all phase from positive to negative offset, finishing at a zero phase offset. Investigating further – it is evident from the large variation in the quadrature channel of the pulse shape that the feedback control optimisation is trying to compensate for the ineffective initial phase calibration (calibrated to that for hard pulses without a quadrature

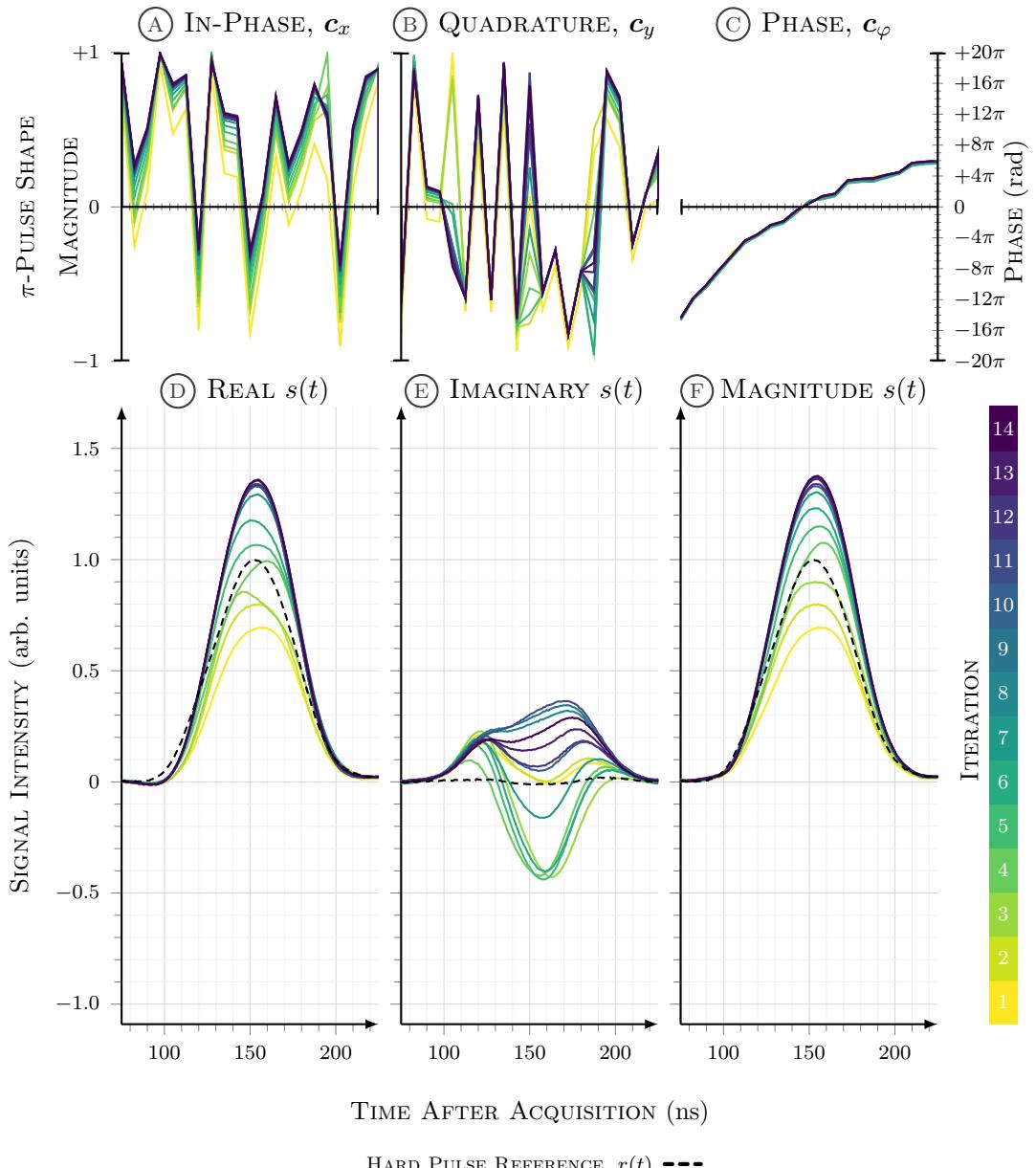


FIGURE 8.5: (A) In-phase, (B) quadrature, and (C) phase parts of the pulse shape at each iteration. The spin echo experiment performed on a nitroxide radical and the optimisation variables are 21 discrete shape points of a 32 ns π pulse. (D) Real, (E) imaginary, and (F) magnitude of the measured signal at each iteration.

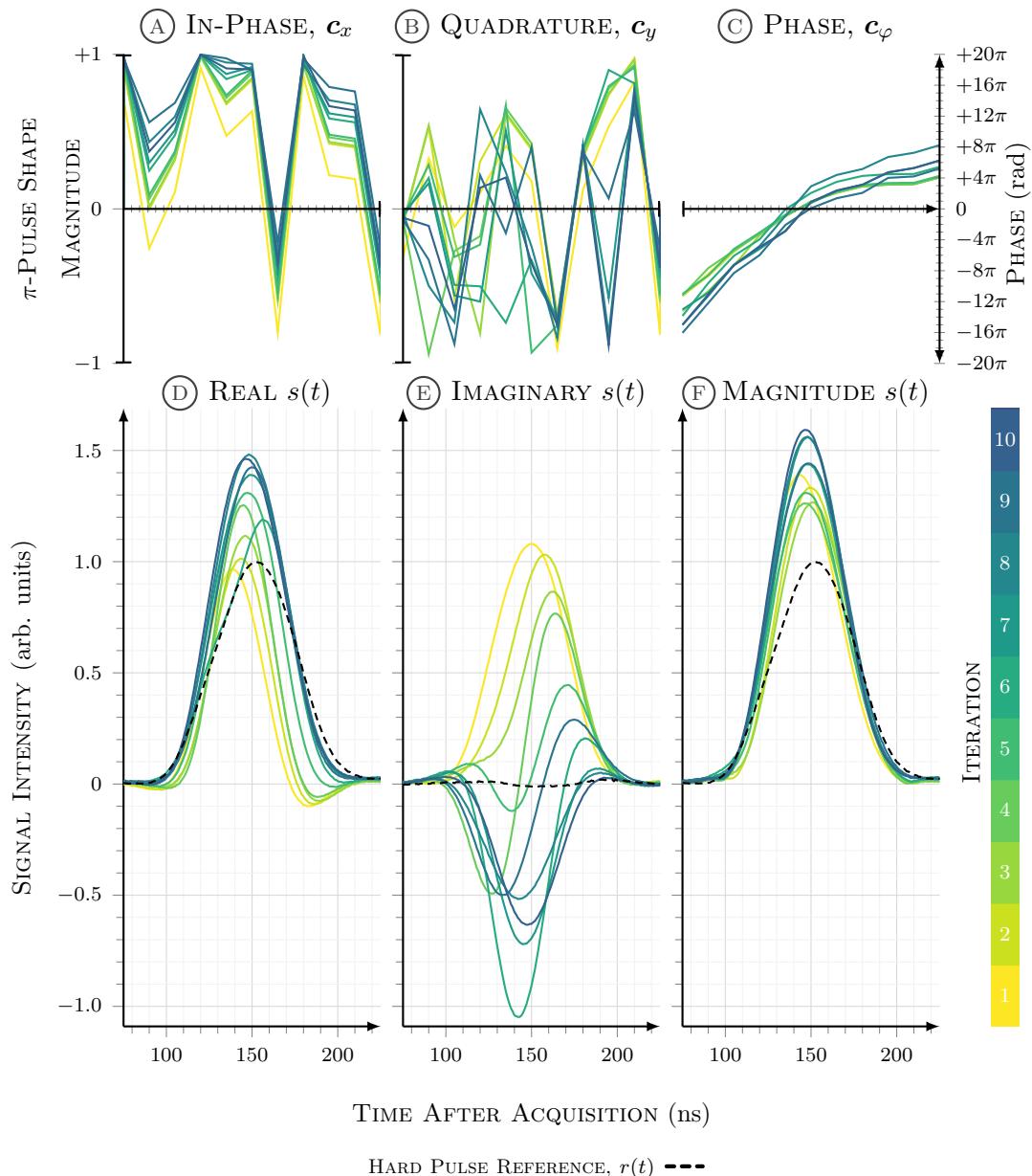


FIGURE 8.6: (A) In-phase, (B) quadrature, and (C) phase parts of the pulse shape at each iteration. The spin echo experiment performed on a nitroxide radical and the optimisation variables are 11 discrete shape points of a 32 ns π pulse. (D) Real, (E) imaginary, and (F) magnitude of the measured signal at each iteration.

element). This is also evident to a lesser degree from the 22 point pulse shape in FIG. 8.5.

In effect, these feedback control optimisations are giving better echo signals by being able to utilise the quadrature pulse channel at the same time as the in-phase channel, and the only real optimisation is to find the correct signal phase offset to give the largest echo integral. An important conclusion is that the development of the waveform shaping equipment in ESR spectroscopy must at this point focus on improving the instrument response function – there is little to gain by seeking faster arbitrary waveform generators until the width of the convolution kernel is reduced. The convolution of the 11 point optimum pulse from FIG. 8.6 with the experimentally measured response function for our EleXSys II E580 spectrometer is shown in FIG. 8.7.

§ 8.2.2 Feedback Optimised OOP-ESEEM

As an application of optimised spin echoes the OOP-ESEEM experiment, shown in FIG. 8.8, would need to optimise two echoes, one being a maximum echo at a delay time in the expected maximum modulation, and the other a minimum at the first minimum of the modulated signal. The objective metric to optimise would reflect that of a full OOP-ESEEM experiment, but only at these two delays:

$$J(\mathbf{c}) = \frac{\|\vec{s}_{d_1}\|_1^{(g)}}{\|\vec{r}_{d_1}\|_1^{(g)}} - \frac{\|\vec{s}_{d_2}\|_1^{(g)}}{\|\vec{r}_{d_2}\|_1^{(g)}} \quad (8.3)$$

where \vec{s}_{d_1} is the echo with a delay d_1 , corresponding to the first modulation maximum, \vec{s}_{d_2} is the echo with a delay d_2 , corresponding to the first modulation minimum, and similarly for the hard pulse reference \vec{r} . The first maximum is placed at 16 ns and the second at 144 ns (approximate values are found from an OOP-ESEEM experiment with hard pulses, shown in FIG. 8.9). The superscript (g) on the 1-norm indicates that the integral is only performed over the integrator gate width (as would be done for a full OOP-ESEEM experiment, set to 32 ns for these experiments). The time stepping of the OOP-ESEEM experiments is set to a coarse 6 ns, and only 22 shots-per-point/transient averages are used. In these feedback controlled optimisations, the initial $\pi/2$ -pulse is optimised and the π -pulse is left as a hard pulse. The reason for investigating the initial pulse character is that it has previously been investigated [306] and found to be important for OOP-ESEEM experiments.

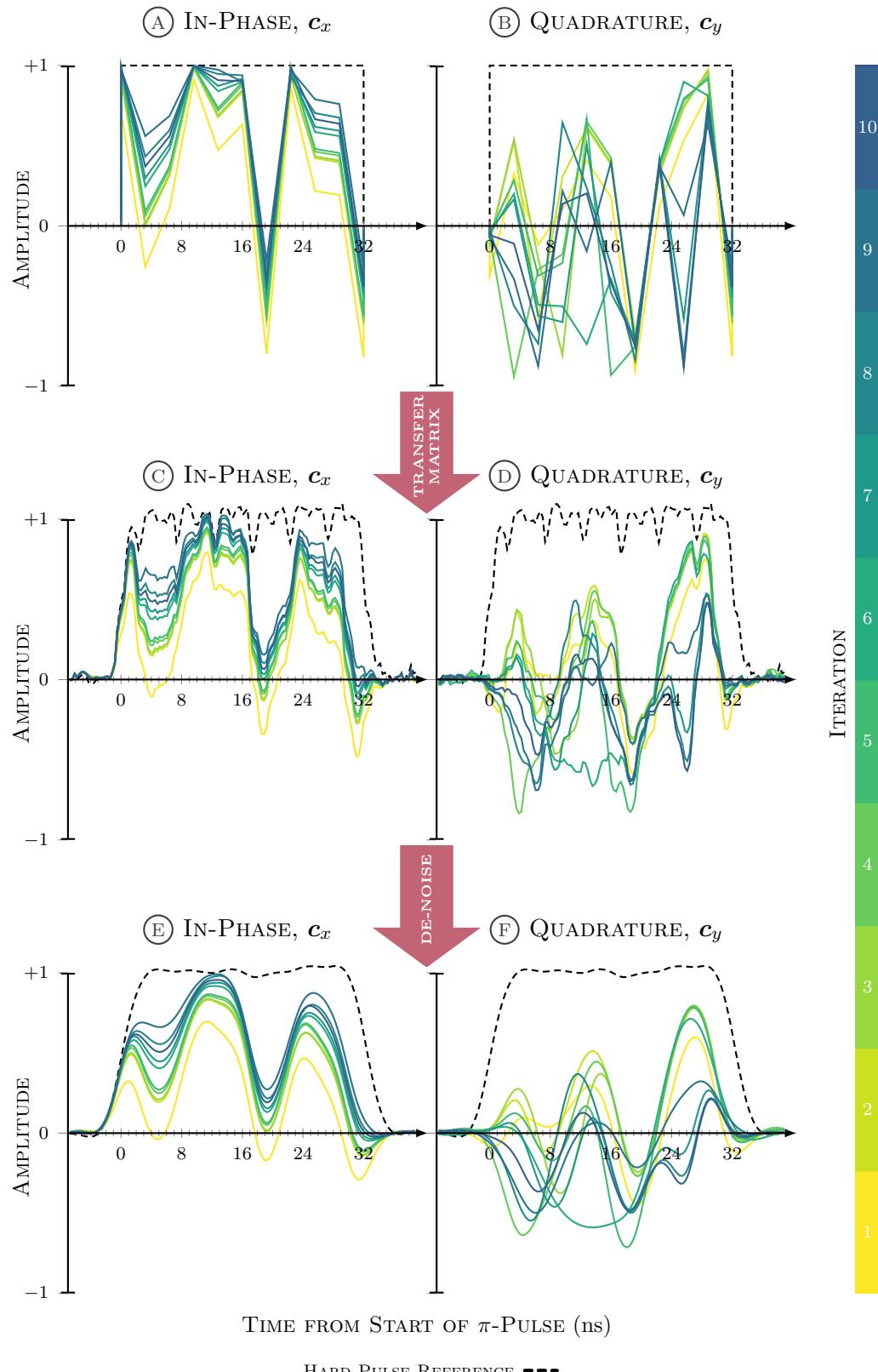


FIGURE 8.7: In-phase and quadrature parts of pulse-shapes from FIG. 8.6 (A)-(B), transformed using a 2 ns resolution transfer matrix in (C)-(D), followed by de-noising with a regularised least squares filter in (E)-(F).

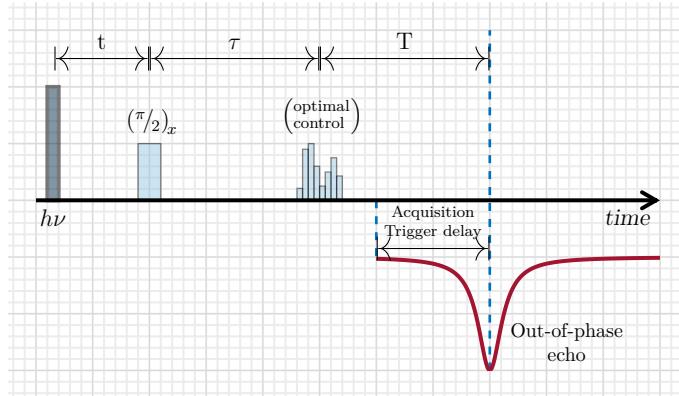


FIGURE 8.8: Out-of-phase ESEEM pulse sequence diagram, showing feedback control pulse (adapted from [33])

Considering the observation that results from the spin echo optimisation are finding an appropriate signal phase within their quadrature pulse channel, and that the in-phase pulse channel does not change appreciably from its initial shape; this should be defined explicitly as part of the optimisation variables.

Chirped pulses [57, 58, 151] and WURST pulse [292, 293] have an easily controllable phase and will be used as the basis of pulse shapes in this section of work. A WURST-type chirped pulse can be characterised by its amplitude and phase as

$$\begin{aligned} \mathbf{c}_r &= 2\pi\sqrt{\frac{\omega_{\text{bw}}}{T}} \left(1 - \left| \sin^a \left(\frac{\pi t}{T} \right) \right| \right) \\ \mathbf{c}_\varphi &= \pi \frac{\omega_{\text{bw}}}{T} t^2 \end{aligned} \quad (8.4)$$

where T is the total pulse duration, ω_{bw} is the pulse excitation bandwidth, a is the sine power used for the amplitude envelope (set to 25% in these experiments), and t is a time grid from $-\frac{T}{2}$ to $+\frac{T}{2}$. The variables of these shapes used for the objective function are the pulse excitation bandwidth, ω_{bw} , the total pulse duration, T , the phase offset of the pulse shape, $\mathbf{c}_{\varphi 0}$, and the nominal pulse amplitude of the waveform shape, $\mathbf{c}_{r\text{max}}$ (defined as a percentage of the maximum pulse amplitude in the spectrometer configuration). These shapes are characterised by few variables, which should enable simplex methods to perform effectively after appropriate scaling.

A first test of this optimisation using the Nelder-Mead algorithm proved unstable – the discontinuity of the objective variables forced the optimiser into local maxima, either contracting onto the pulse shape, the pulse length or the phase offset. This should be expected with discontinuous variables in the Nelder-Mead algorithm [79].

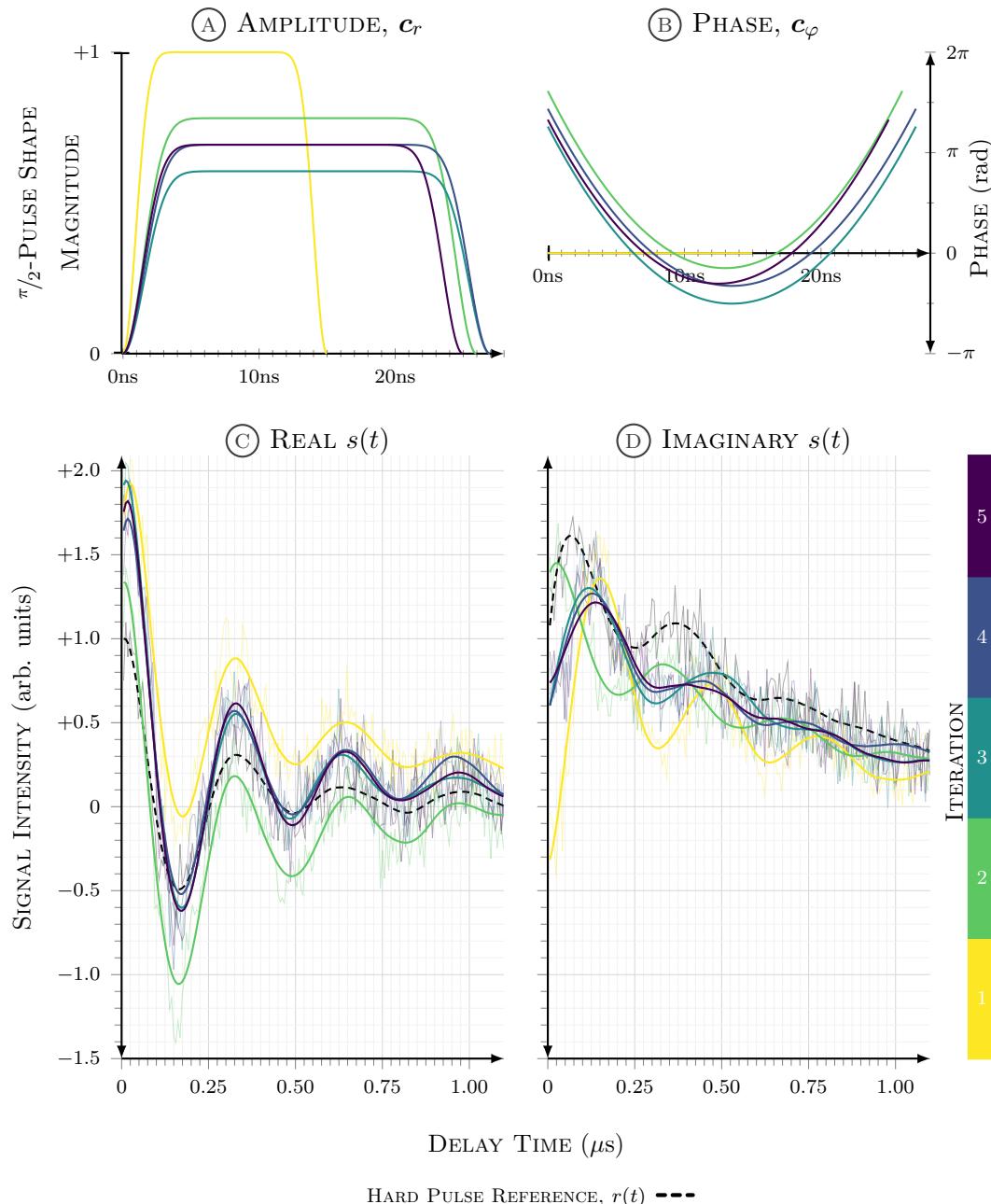


FIGURE 8.9: (A) Amplitude, and (B) phase parts of the WURST shape at each iteration. (C) Real, and (D) imaginary parts of the measured OOP-ESEEM signal at each iteration, normalised to the peak intensity of the real part of the reference signal from hard pulses. Opaque lines show the raw data and bold lines show data smoothed with a multi-pass Savitzky-Golay filter [261] (1st pass: 25-point cubic; 2nd pass: 75-point quintic).

Now that the dimensionality of the problem has reduced substantially by using chirped pulses, the multidirectional search algorithm can now become less expensive. Furthermore, the pattern-search type nature of the algorithm allows it to deal effectively with the discontinuous variables of this optimisation. The results of using this optimisation method are shown in FIG. 8.9.

The use of a phase offset with chirped-WURST pulses as variables of a multidirectional search optimisation can be considered quite successful in increasing the modulation depth of the OOP-ESEEM experiment compared to a similar experiment with hard pulses alone. The number of iterations is very low and utilising a partially parallel version of the simplex method reduces the total time taken to less than 2 minutes §B.3.

The optimisation can be considered converged because the simplex size reduces to less than the defined tolerance, which is a feature of the multidirectional search convergence [307]. These solutions should not be considered unique: the optimisation was started from a chirped pulse with a very small bandwidth, almost a hard pulse. When the optimisation is started from other random starts, it will find other solutions that give a good modulation depth – these optimisation methods are not global searches.

It should be noted that the results shown in FIG. 8.9 are explicit OOP-ESEEM experiments performed at each improved solution. This is for display in this thesis rather than a need of the optimisation – this only needs two of the 256 echo experiments needed for an OOP-ESEEM experiment. This extra time to perform the OOP-ESEEM experiment is not included in the timing of FIG. B.2. Furthermore, only 22 transient averages were used in the echo signal - because this method takes a very short amount of time, a much larger number of transient averages could be used, removing much of the noise in the raw data of FIG. 8.9 at very little cost compared to increasing the shots-per-point by a similar amount in the OOP-ESEEM experiment.

Using a simple feedback control with a gradient-free optimisation algorithm of a shaped pulse can improve the signal intensity of a spin echo ESR experiment. The simplicity, almost naïvety, of this feedback controlled optimisation is that experiment and sample specific pulse distortions need not be classified, which could be time-consuming and require rare expertise. Furthermore, this blind method of finding broadband pulses does not need to be time-consuming and can become a useful tool, similar to the convenience of an auto-shim.

This work may be considered an extra tool in the ESR optimisation toolbox. Code to produce examples within this manuscript are a combination of Matlab functions of the *Spinach* libraries [130] to parse data, optimise an objective function, and invoke Python scripts to communicate with the AWG using the XEPRAPI libraries. The function of Matlab in this context is a triviality, and code is not shown here. However, using Python scripts to communicate with the AWG in ways set out in this manuscript is not well documented. These communications scripts are bundled with *Spinach*. A user may implement these methods with relative ease and very little specialist knowledge of the hardware electronics or optimisation methods. Furthermore, this feedback control can be used in conjunction with transfer function methods as a fine-tuning of the numerical method [69, 123].

§ 9

Summary

Finally, from so little sleeping and so much reading, his brain dried up and he went completely out of his mind.

– Miguel de Cervantes Saavedra, DON QUIXOTE

A Newton-GRAPE method was developed using efficient calculation of analytical second directional derivatives. The method was developed to scale with the same complexity as methods that use only first directional derivatives. Algorithms to ensure a well-conditioned and positive definite matrix of second directional derivatives are used so the sufficient conditions of gradient-based numerical optimisation are met.

§ A

Irreducible Spherical Tensor Operators

	$d_{m,2}^{(2)}$	$d_{m,1}^{(2)}$	$d_{m,0}^{(2)}$	$d_{m,-1}^{(2)}$	$d_{m,-2}^{(2)}$
$d_{2,n}^{(2)}$	$\frac{(1+\cos\beta)^2}{4}$	$-\frac{(1+\cos\beta)\sin\beta}{2}$	$\sqrt{\frac{3}{8}}\sin^2\beta$	$-\frac{(1-\cos\beta)\sin\beta}{2}$	$\frac{(1-\cos\beta)^2}{4}$
$d_{1,n}^{(2)}$	$\frac{(1+\cos\beta)\sin\beta}{2}$	$\frac{\cos\beta-1}{2} + \cos^2\beta$	$-\sqrt{\frac{3}{8}}\sin 2\beta$	$\frac{\cos\beta+1}{2} - \cos^2\beta$	$-\frac{(1-\cos\beta)\sin\beta}{2}$
$d_{0,n}^{(2)}$	$\sqrt{\frac{3}{8}}\sin^2\beta$	$\sqrt{\frac{3}{8}}\sin 2\beta$	$\frac{3\cos^2\beta-1}{2}$	$-\sqrt{\frac{3}{8}}\sin 2\beta$	$\sqrt{\frac{3}{8}}\sin^2\beta$
$d_{-1,n}^{(2)}$	$\frac{(1-\cos\beta)\sin\beta}{2}$	$\frac{\cos\beta+1}{2} - \cos^2\beta$	$\sqrt{\frac{3}{8}}\sin 2\beta$	$\frac{\cos\beta-1}{2} + \cos^2\beta$	$-\frac{(1+\cos\beta)\sin\beta}{2}$
$d_{-2,n}^{(2)}$	$\frac{(1-\cos\beta)^2}{4}$	$\frac{(1-\cos\beta)\sin\beta}{2}$	$\sqrt{\frac{3}{8}}\sin^2\beta$	$\frac{(1+\cos\beta)\sin\beta}{2}$	$\frac{(1+\cos\beta)^2}{4}$

FIGURE A.1: Second-rank Wigner functions are defined in terms of reduced functions as $\mathfrak{D}_{m,n}^{(2)}(\alpha, \beta, \gamma) = e^{-im\alpha} d_{m,n}^{(2)}(\beta) e^{-in\gamma}$

ℓ	m	$Y_\ell^m(\theta, \varphi)$	$Y_\ell^m(x, y, z)$	$\hat{T}_{\ell m}(\hat{L})$
0	0	$\sqrt{\frac{1}{4\pi}}$	$\sqrt{\frac{1}{4\pi}}$	$\mathbb{1}$
1	1	$-\sqrt{\frac{3}{8\pi}} e^{i\varphi} \sin \theta$	$-\sqrt{\frac{3}{8\pi}} \frac{x+iy}{r}$	$-\sqrt{\frac{1}{2}} \hat{L}_+$
1	0	$\sqrt{\frac{3}{4\pi}} \cos \theta$	$\sqrt{\frac{3}{4\pi}} \frac{z}{r}$	\hat{L}_z
1	-1	$\sqrt{\frac{3}{8\pi}} e^{-i\varphi} \sin \theta$	$\sqrt{\frac{3}{8\pi}} \frac{x-iy}{r}$	$\sqrt{\frac{1}{2}} \hat{L}_-$
2	2	$\sqrt{\frac{15}{32\pi}} e^{2i\varphi} \sin^2 \theta$	$\sqrt{\frac{15}{32\pi}} \frac{(x+iy)^2}{r^2}$	$\frac{1}{2} \hat{L}_+^2$
2	1	$-\sqrt{\frac{15}{8\pi}} e^{i\varphi} \sin \theta \cos \theta$	$-\sqrt{\frac{15}{8\pi}} \frac{(x+iy)z}{r^2}$	$-\frac{1}{2} (\hat{L}_z \hat{L}_+ + \hat{L}_+ \hat{L}_z)$
2	0	$\sqrt{\frac{5}{16\pi}} (3 \cos^2 \theta - 1)$	$\sqrt{\frac{5}{16\pi}} \frac{2z^2 - x^2 - y^2}{r^2}$	$\sqrt{\frac{2}{3}} (\hat{L}_z^2 - \frac{1}{4} (\hat{L}_+ \hat{L}_- + \hat{L}_- \hat{L}_+))$
2	-1	$\sqrt{\frac{15}{8\pi}} e^{-i\varphi} \sin \theta \cos \theta$	$\sqrt{\frac{15}{8\pi}} \frac{(x-iy)z}{r^2}$	$\frac{1}{2} (\hat{L}_z \hat{L}_- + \hat{L}_- \hat{L}_z)$
2	-2	$\sqrt{\frac{15}{32\pi}} e^{-2i\varphi} \sin^2 \theta$	$\sqrt{\frac{15}{32\pi}} \frac{(x-iy)^2}{r^2}$	$\frac{1}{2} \hat{L}_-^2$

FIGURE A.2: Spherical harmonics may be written in terms of Cartesian coordinates, then to be replaced by corresponding Cartesian operators, to give rank-1 irreducible spherical tensor operators, for a single spin.

ℓ	m	$a_m^{(\ell)}$	$\hat{T}_{\ell m}(\hat{\vec{L}}, \hat{\vec{S}})$
0	0	$-\frac{1}{\sqrt{3}}(a_{xx} + a_{yy} + a_{zz})$	$-\frac{1}{\sqrt{3}}\hat{\vec{L}} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \hat{\vec{S}}$
1	1	$-\frac{1}{2}(a_{zx} - a_{xz} - i(a_{zy} - a_{yz}))$	$-\frac{1}{2}\hat{\vec{L}} \cdot \begin{pmatrix} 0 & 0 & -1 \\ 0 & 0 & -i \\ 1 & i & 0 \end{pmatrix} \cdot \hat{\vec{S}}$
1	0	$-\frac{i}{\sqrt{2}}(a_{xy} - a_{yx})$	$-\frac{1}{\sqrt{2}}\hat{\vec{L}} \cdot \begin{pmatrix} 0 & -i & 0 \\ i & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \hat{\vec{S}}$
1	-1	$-\frac{1}{2}(a_{zx} - a_{xz} + i(a_{zy} - a_{yz}))$	$-\frac{1}{2}\hat{\vec{L}} \cdot \begin{pmatrix} 0 & 0 & -1 \\ 0 & 0 & i \\ 1 & -i & 0 \end{pmatrix} \cdot \hat{\vec{S}}$
2	2	$+\frac{1}{2}(a_{xx} - a_{yy} - i(a_{xy} + a_{yx}))$	$+\frac{1}{2}\hat{\vec{L}} \cdot \begin{pmatrix} 1 & i & 0 \\ i & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \hat{\vec{S}}$
2	1	$-\frac{1}{2}(a_{xz} + a_{zx} - i(a_{yz} + a_{zy}))$	$-\frac{1}{2}\hat{\vec{L}} \cdot \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & i \\ 1 & i & 0 \end{pmatrix} \cdot \hat{\vec{S}}$
2	0	$+\frac{i}{\sqrt{6}}(2a_{zz} - (a_{xx} + a_{yy}))$	$+\frac{1}{\sqrt{6}}\hat{\vec{L}} \cdot \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 2 \end{pmatrix} \cdot \hat{\vec{S}}$
2	-1	$+\frac{1}{2}(a_{xz} + a_{zx} + i(a_{yz} + a_{zy}))$	$+\frac{1}{2}\hat{\vec{L}} \cdot \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & -i \\ 1 & -i & 0 \end{pmatrix} \cdot \hat{\vec{S}}$
2	-2	$+\frac{1}{2}(a_{xx} - a_{yy} + i(a_{xy} + a_{yx}))$	$+\frac{1}{2}\hat{\vec{L}} \cdot \begin{pmatrix} 1 & -i & 0 \\ -i & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \hat{\vec{S}}$

FIGURE A.3: The coefficients required for the transformation between the standard 3×3 matrix representation and rank-2 irreducible spherical tensors operators, for 2 spins.

§ B

The Arbitrary Waveform Generator in Optimisation

§ B.1 AWG Schematics

A schematic block diagram of an AWG is shown in FIG. B.1 and has the following function:

- ① A set of pulses, in the form of a specified code, is sent to the AWG through an Ethernet port, processed by on-board computation hardware.

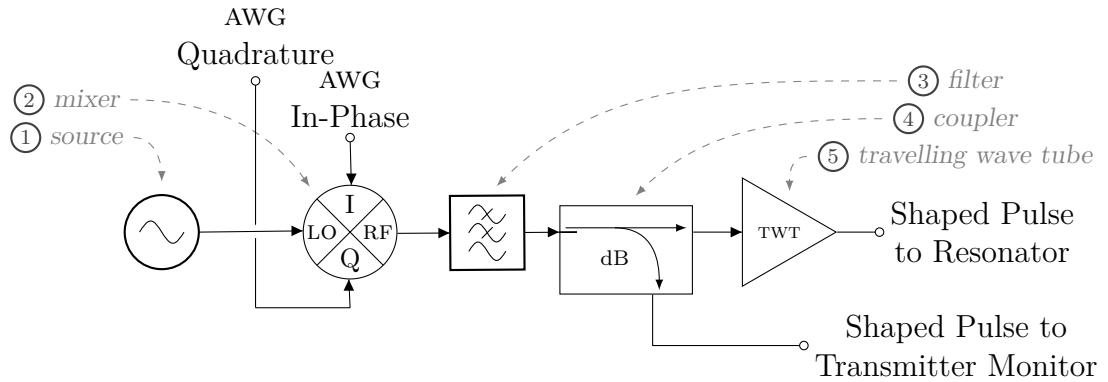


FIGURE B.1: The Bruker SpinJet Schematics.

② Pulses are formed on an in-phase channel (I channel) and an 90° out-of-phase, quadrature channel (Q channel).

$$\text{AWG}_I = \cos \omega_a t$$

$$\text{AWG}_Q = \sin \omega_a t$$

③ Two AWG channels are connected to the I and Q ports of an IQ-mixer.

④ The IQ-mixer takes the amplified signal from a microwave source, then splitting this into an in-phase channel and its quadrature channel

$$LO_I = \cos \omega_o t$$

$$LO_Q = \sin \omega_o t$$

where ω_o is the angular frequency of the carrier, microwave signal (local oscillator).

⑤ Separately for the I and Q channels, the AWG channel and corresponding microwave channel are mixed.

$$I = \cos(\omega_o - \omega_a)t + \cos(\omega_o + \omega_a)t,$$

$$Q = \sin(\omega_o - \omega_a)t - \sin(\omega_o + \omega_a)t$$

⑥ These two mixed signals are then recombined at the final stage of the IQ-mixer.

§ B.2 Xepr Python code

To programme a feedback loop on to the combination of AWG and spectrometer, Python interpreted XEPR API scripting can be used for operation of the XEPR application by Bruker BioSpin. XEPR API can be easily combined with the language ProDeL^{11} used in PulseSPEL files as a back-end to the XEPR console software.

^{11}PROcedure DEscription Language embedded within the XEPR software

Load current experiment

A header script must be invoked at the start of every XEPR Python script, to import libraries needed and to locate and load the XEPR API module. After the header, `XexprAPI.Xexpr()` creates the XEPR object and connects to the XEPR software, the API has to be enabled within the XEPR GUI menu^{I} API. Also common to XEPR scripts is the command to create/access an experiment object of the XEPR application, invoked with `cur_exp = Xexpr.XexprExperiment()`, to access the current XEPR experiment or `config = Xexpr.XexprExperiment("AcqHidden")` to access XEPR's hidden experiment object.

The current experiment object must be imported to the current workspace, assuming there is a current experiment on the spectrometer. The hidden experiment object will become useful in modifying some parts of the experiment.

```

1 # load the Xexpr API module into Xexpr structure
2 Xexpr = XexprAPI.Xexpr()
3
4 # define the current and hidden experiments
5 currentExp = Xexpr.XexprExperiment()
6 hiddenExp = Xexpr.XexprExperiment("AcqHidden")

```

Acquire signal data

Measure the 1D or 2D signal as an array of real numbers, corresponding to relative intensity of the measured signal. The current experiment object should already be imported to the workspace. The signal data should be stored in a plain text file to be used later, within Matlab.

Having established a connection to XEPR through the XEPRAPI module, and accessed a current experiment within the XEPR GUI^{II}, the next task is to modify values of variables that will become the variables of the objective function.

```

1 # run compiled experiment, wait until finished
2 currentExp.aqExpRunAndWait()
3
4 # acquire current signal dataset
5 dset = Xexpr.XexprDataset()

```

^{I}Processing → XexprAPI → Enable Xexpr

^{II}An initial experiment should be set up within the XEPR GUI before these scripts are invoked; the intention is to modify values of variables already set within the XEPR GUI.

```

6
7 # separate data into x, real(y), and imag(y)
8 ordinate = dset.0
9 x_signal = dset.X
10 rY_signal = ordinate.real
11 iy_signal = ordinate.imag

```

Modify PulseSPEL variables

PulseSPEL definitions from the definition file can be modified by initially parsing all PulseSPEL parameters contained in the experiment and then matching them with inputs to the Python script, the matched parameters can then be modified as instructed. The first input to the Python script is the variable name and the second is the new value.

```

1 # get current PulseSPEL variables
2 Defs = currentExp.getParam("PlsSPELGlbTxt").value
3 Defs = Defs.split("\n")
4
5 # replace matched PulseSPEL variables
6 for value in Defs:
7     if str(sys.argv[1]) in value:
8         new = str(sys.argv[1])+" = "+str(sys.argv[2])
9         currentExp["ftEPR.PlsSPELSetVar"].value = new

```

The script will read comma-separated variable names stored in the first line of the text file `data_varsfile.txt`, which can be created from within Matlab, then match them to similarly ordered comma-separated variable values in the second line of the text file. Although it is not ideal to pass variables from one script to another in this way, it is a simple way which works given a waiting time between commands. The script also contains an error handling for variables defined in `data_varsfile.txt` which have not been previously defined in the XEPR experiment. The command `cur_exp["ftESR.PlsSPELSetVar"].value = cmdStr` sets the current experiment's PulseSPEL variables to those scraped from the externally created `data_varsfile.txt` file to the string `cmdStr`.

Load shape file

Assuming a plain text shape file has already been created, the code to send this shape to the experiment involves loading this shape file to the workspace, then

compiling it. The argument of the Python script is the name of the shape file. When a maximum number of uploads is reached, 115, the current experiment should be reset (not shown here).

```

1 # Load shapefile into PulseSPEL
2 Xepr.XeprCmds.aqPgShpLoad(str(sys.argv[1]))
3
4 # Show shapefile in PulseSPEL window
5 Xepr.XeprCmds.aqPgShowShp()
6
7 # Compile shapefile into experiment
8 Xepr.XeprCmds.aqPgCompile()

```

The final step is to run the current experiment with the modified variables with the command `cur_exp.aqExpRunAndWait()` and wait for the experiment to finish.

Within an objective function, a Matlab optimisation should modify the values of the variables in the file `data_varsfile.txt` at each iteration, then to call the Python script to send these modified values to the spectrometer. The result of the modified experiment should pass a metric to be maximised/minimised back to Matlab's optimisation; the objective of the objective function. A simple metric to use would be that which gives the best echo signal i.e. maximising the echo.

The echo signal data must be read from the XEPR software, again, this is achieved with the XEPR API. Script that acquires the XEPR data with the command `dset = Xepr.XeprDataset()`, then to sort and store relevant data in three text files to be read by Matlab's objective function an interpreted as an echo. A simple echo signal maximisation would maximise the integral of the echo shape.

The optimal control strategy set out so far is no more than an automatic calibration, which a moderately experienced experimenter can achieve in very little time through the XEPR GUI. In fact, with error handling and waiting times included in the Python scripts outlined so far, each experimental modification will take a number of seconds: a simplex method would take a long time to search through this variable space to its optimal experimental settings. However, this is just a simple example to text the two-way communication between an optimisation algorithm and its “hardware objective function”.

Experiment:	OOP-ESEEM		
Pulse sequence:	$h\nu \rightarrow \frac{\pi}{2} \rightarrow \pi \rightarrow s(t)$		
Optimised Pulse:	$\frac{\pi}{2}$ -pulse		
$\frac{\pi}{2}$ pulse length:	$2\text{ns} \leq \text{variable} \leq 160\text{ns}$		
π pulse length:	32ns		
Phase cycling:	16-step		
Transient averages:	22		
Algorithm:	NM		
Pulse shape:	wurst		
Phase offset:	none		
	$-\frac{\pi}{2} < \text{variable} \leq +\frac{\pi}{2}$		
Iteration	X	# $f(\mathbf{c})$	$J(\mathbf{c})$
1:	I	4	0.20
2:	E	2	0.39
3:	C	2	0.56
4:	R	2	0.79
5:	C	2	1.03
6:	S	7	1.20
7:	S	7	1.20
8:	S	7	1.20
9:	C	2	1.20
Total experiments:	2×42		
Total time:	218s		
Average f(x) time:	5.2s		
MDS	2×37		
	99s		
	2.7s		

FIGURE B.2: Results of the feedback control optimisation of OOP-ESEEM experiments using Nelder-Mead (NM) and multidirectional search (MDS) algorithms. $J(\mathbf{c})$ is the quality of the echo signal. $\#f(\mathbf{c})$ is the number of functional evaluations (2 experiments for OOP-ESEEM). The step is the type of simplex \mathbf{X} = at that iteration – **I** = initial, **R** = reflect, **E** = expand, **C** = contract, **S** = shrink. The termination condition of the algorithms was a simplex size < 0.05 or no progress was made after 150 experiment evaluations were made during the contraction steps. In each case the objective value was normalised to the same measure from the same experiments using conventional hard pulses.

§ B.3 Feedback Control Results Tables

Figs. B.2, B.3 outline the results obtained in §8.2 for the feedback control of 2-pulse Hahn echo experiments and OOP-ESEEM experiments. The variables of optimisation are the piecewise triangular shape increments of a π -pulse in the echo experiments, and the WURST pulse shape parameters (pulse length, amplitude, phase offset, and pulse excitation bandwidth from Eq. (8.4)) in the OOP-ESEEM experiments.

Experiment:				2-pulse echo					
Pulse sequence:				$\frac{\pi}{2} \rightarrow \pi \rightarrow s(t)$					
Optimised Pulse:				π -pulse					
$\frac{\pi}{2}$ pulse length:				16ns					
π pulse length:				32ns					
Phase cycling:				none					
Transient averages:				200					
Algorithm:		NM			NM			NM	
Pulse shape:		33 point			21 point			11 point	
Phase offset:		none			none			none	
Iteration	X	# $f(\mathbf{c})$	$J(\mathbf{c})$	X	# $f(\mathbf{c})$	$J(\mathbf{c})$	X	# $f(\mathbf{c})$	$J(\mathbf{c})$
1:	I	67	0.43	I	43	0.75	I	23	0.71
2:	R	2	0.44	E	24	0.83	E	4	0.71
3:	E	20	0.47	E	16	0.89	E	2	0.80
4:	E	20	0.52	E	18	1.09	E	6	0.84
5:	E	16	0.59	E	40	1.09	E	2	1.02
6:	E	26	0.72	E	58	1.16	E	10	1.03
7:	E	60	0.93	E	58	1.21	E	2	1.23
8:	E	104	0.98	E	20	1.21	E	28	1.27
9:	R	24	0.98	R	2	1.22	R	14	1.28
10:	R	2	0.98	E	36	1.24	C	26	1.28
11:				R	14	1.24			
12:				E	16	1.24			
13:				R	18	1.25			
14:				R	16	1.27			
Total experiments:		341			389			119	
Total time:		1030s			1273s			380s	
Average f(x) time:		3.0s			3.3s			3.2s	

FIGURE B.3: Results of the feedback control optimisation of 2-pulse echo experiments using the Nelder-Mead (NM) algorithm. A number of contraction steps are omitted from the 2-pulse echo experiments, where no objective improvement was made. $J(\mathbf{c})$ is the quality of the echo signal. $\#f(\mathbf{c})$ is the number of functional evaluations (1 experiment for the 2-pulse echoes). The step is the type of simplex \mathbf{X} = at that iteration – **I** = initial, **R** = reflect, **E** = expand, **C** = contract, **S** = shrink. The termination condition of the algorithms was a simplex size < 0.05 or no progress was made after 150 experiment evaluations were made during the contraction steps. In each case the objective value was normalised to the same measure from the same experiments using conventional hard pulses.

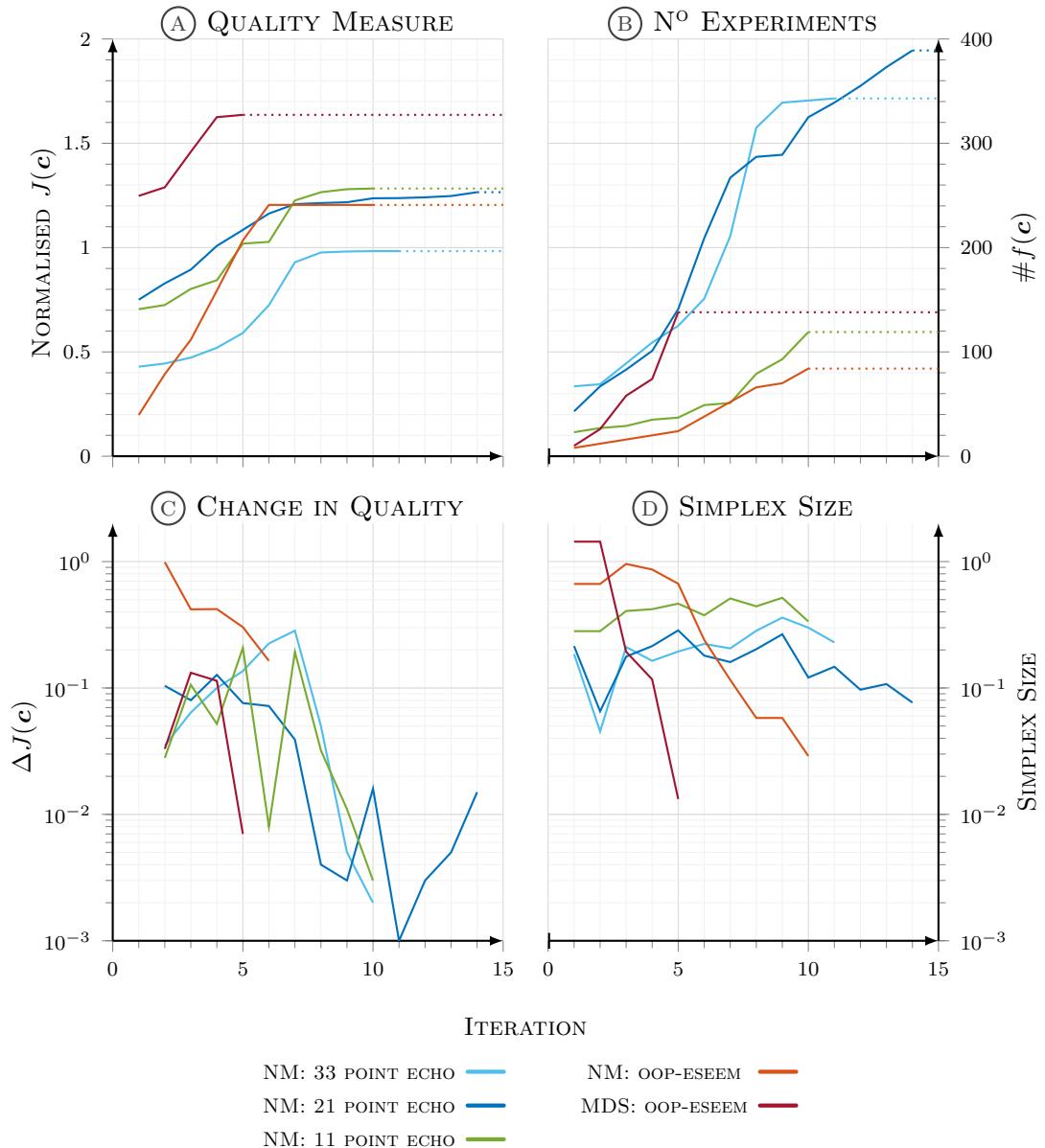


FIGURE B.4: Convergence of Nelder-Mead (NM) and multidirectional search (MDS) algorithms. (A) shows the quality of the echo signal and (C) is its change. (B) shows the number of functional evaluations and (D) is a measure of simplex size.

Echo experiments all start from a random π -pulse shape and OOP-ESEEM experiments start with random parameters of the WURST pulse at 32 ns, seeded from the same random number. All WURST pulse shapes have a amplitude envelope of 25% rise/fall time. Convergence characteristics of these feedback control experiments are shown in FIG. B.4.

Bibliography

- [1] A. Abragam. *The Principles of Nuclear Magnetism* (Clarendon Press, 1961).
ISBN: 978-0-19-852014-6
- [2] K. Aizu. *Parameter differentiation of quantum-mechanical linear operators*, Journal of Mathematical Physics **4**, 6 (1963), pp. 762–775. <https://doi.org/10.1063/1.1724318>
- [3] A. H. Al-Mohy, N. J. Higham. *A new scaling and squaring algorithm for the matrix exponential*, SIAM Journal on Matrix Analysis and Applications **31**, 3 (2009), pp. 970–989. <https://doi.org/10.1137/09074721x>
- [4] G. M. Amdahl. *Proceedings of the April 18-20, 1967, spring joint computer conference on - AFIPS '67 (Spring)* (1967), pp. 483–485. <https://doi.org/10.1145/1465482.1465560>
- [5] C. K. Anand, A. D. Bain, A. T. Curtis, Z. Nie. *Designing optimal universal pulses using second-order, large-scale, non-linear optimization*, Journal of Magnetic Resonance **219** (2012), pp. 61–74. <https://doi.org/10.1016/j.jmr.2012.04.004>
- [6] C. Arenz, G. Gualdi, D. Burgarth. *Control of open quantum systems: Case study of the central spin model*, New Journal of Physics **16** (2014), p. 065023. <https://doi.org/10.1088/1367-2630/16/6/065023>
- [7] G. B. Arfken, H. J. Weber. *Mathematical Methods for Physicists*, 5th edition (Academic Press, 2001). ISBN: 978-0-12-059826-7
- [8] L. Armijo. *Minimization of functions having Lipschitz continuous first partial derivatives*, Pacific Journal of Mathematics **16**, 1 (1966), pp. 1–3. <https://doi.org/10.2140/pjm.1966.16.1>

- [9] W. E. Arnoldi. *The principle of minimized iterations in the solution of the matrix eigenvalue problem*, Quarterly of Applied Mathematics **9**, 1 (1951), pp. 17–29. <https://doi.org/10.1090/qam/42792>
- [10] T. Auckenthaler, M. Bader, T. Huckle, A. Spörl, K. Waldherr. *Matrix exponentials and parallel prefix computation in a quantum control problem*, Parallel Computing **36**, 5-6 (2010), pp. 359–369. <https://doi.org/10.1016/j.parco.2010.01.006>
- [11] A. D. Bain, R. S. Dumont. *Introduction to Floquet theory: The calculation of spinning sideband intensities in magic-angle spinning NMR*, Concepts in Magnetic Resonance **13**, 3 (2001), pp. 159–170. <https://doi.org/10.1002/cmr.1006>
- [12] M. Bak, N. C. Nielsen. *REPULSION, a novel approach to efficient powder averaging in solid-state NMR*, Journal of Magnetic Resonance **125**, 1 (1997), pp. 132–139. <https://doi.org/10.1006/jmre.1996.1087>
- [13] M. Bak, J. T. Rasmussen, N. C. Nielsen. *SIMPSON: A general simulation program for solid-state NMR spectroscopy*, Journal of Magnetic Resonance **147** (2000), pp. 296–330. <https://doi.org/10.1006/jmre.2000.2179>
- [14] J. Baker. *An algorithm for the location of transition states*, Journal of Computational Chemistry **7**, 4 (1986), pp. 385–395. <https://doi.org/10.1002/jcc.540070402>
- [15] A. Banerjee, N. Adams, J. Simons, R. Shepard. *Search for stationary points on surfaces*, The Journal of Physical Chemistry **89**, 1 (1985), pp. 52–57. <https://doi.org/10.1021/j100247a015>
- [16] A. Banerjee, F. Grein. *Convergence behavior of some multiconfiguration methods*, International Journal of Quantum Chemistry **10**, 1 (1976), pp. 123–134. <https://doi.org/10.1002/qua.560100110>
- [17] D. P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods* (Athena Scientific, 1996). ISBN: 978-1-88-652904-5
- [18] M. Bjerring, S. Jain, B. Paaske, J. M. Vinther, N. C. Nielsen. *Designing dipolar recoupling and decoupling experiments for biological solid-state NMR using interleaved continuous wave and rf pulse irradiation*, Accounts of Chemical Research **46**, 9 (2013), pp. 2098–2107. <https://doi.org/10.1021/ar300329g>

[19] B. E. Bode, J. Plackmeyer, T. F. Prisner, O. Schiemann. *PELDOR measurements on a Nitroxide-labeled Cu(II) Porphyrin: Orientation selection, spin-density distribution, and conformational flexibility*, The Journal of Physical Chemistry A **112**, 23 (2008), pp. 5064–5073. <https://doi.org/10.1021/jp710504k>

[20] T. W. Borneman, D. G. Cory. *Bandwidth-limited control and ringdown suppression in high-Q resonators*, Journal of Magnetic Resonance **225** (2012), pp. 120–129. <https://doi.org/10.1016/j.jmr.2012.10.011>

[21] T. W. Borneman, M. D. Hürlimann, D. G. Cory. *Application of optimal control to CPMG refocusing pulse design*, Journal of Magnetic Resonance **207**, 2 (2010), pp. 220–233. <https://doi.org/10.1016/j.jmr.2010.09.003>

[22] D. M. Bortz, C. T. Kelley. *Computational Methods for Optimal Design and Control*, J. Borggaard, J. Burns, E. Cliff, S. Schreck, eds. (Birkhäuser Boston, 1998), vol. 24 of *Progress in Systems and Control Theory*. ISBN: 978-1-46-127279-3. https://doi.org/10.1007/978-1-4612-1780-0_5

[23] S. Boyd, L. Vandenberghe. *Convex Optimization* (Cambridge University Press, 2004). ISBN: 978-0-52-183378-3

[24] M. Braun, S. J. Glaser. *Cooperative pulses*, Journal of Magnetic Resonance **207**, 1 (2010), pp. 114–123. <https://doi.org/10.1016/j.jmr.2010.08.013>

[25] M. Braun, S. J. Glaser. *Concurrently optimized cooperative pulses in robust quantum control: Application to broadband Ramsey-type pulse sequence elements*, New Journal of Physics **16** (2014), p. 115002. <https://doi.org/10.1088/1367-2630/16/11/115002>

[26] R. P. Brent. *Algorithms for Minimization without Derivatives* (Dover Publications, 2013). ISBN: 978-0-48-641998-5

[27] C. Bretschneider, A. Karabanov, N. C. Nielsen, W. Köckenberger. *Conversion of parahydrogen induced longitudinal two-spin order to evenly distributed single spin polarisation by optimal control pulse sequences*, Journal of Chemical Physics **136**, 9 (2012), p. 094201. <https://doi.org/10.1063/1.3691193>

[28] C. Brif, R. Chakrabarti, H. Rabitz. *Control of quantum phenomena: past, present and future*, New Journal of Physics **12**, 7 (2010), p. 075008. <https://doi.org/10.1088/1367-2630/12/7/075008>

[29] L. de Broglie. *Recherches sur la théorie des quanta*, Annales de Physique **10**, 3 (1925), pp. 22–128. <https://doi.org/10.1051/anphys/192510030022>

[30] C. G. Broyden. *Quasi-Newton methods and their application to function minimisation*, Mathematics of Computation **21**, 99 (1967), pp. 368–381. <https://doi.org/10.2307/2003239>

[31] C. G. Broyden. *The convergence of a class of double-rank minimization algorithms 1. General considerations*, IMA Journal of Applied Mathematics **6**, 1 (1970), pp. 76–90. <https://doi.org/10.1093/imamat/6.1.76>

[32] C. G. Broyden. *The convergence of a class of double-rank minimization algorithms: 2. The new algorithm*, IMA Journal of Applied Mathematics **6**, 3 (1970), pp. 222–231. <https://doi.org/10.1093/imamat/6.3.222>

[33] M. Brustolon. *Electron Paramagnetic Resonance: A Practitioner’s Toolkit* (Wiley, 2009). ISBN: 978-0-47-025882-8

[34] R. H. Byrd, J. Nocedal, R. B. Schnabel. *Representations of quasi-Newton matrices and their use in limited memory methods*, Mathematical Programming **63**, 1–3 (1994), pp. 129–156. <https://doi.org/10.1007/bf01582063>

[35] T. Caneva, T. Calarco, S. Montangero. *Chopped random-basis quantum optimization*, Physical Review A **84**, 2 (2011), p. 022326. <https://doi.org/10.1103/physreva.84.022326>

[36] T. Caneva, M. Murphy, T. Calarco, R. Fazio, S. Montangero, V. Giovannetti, G. E. Santoro. *Optimal control at the quantum speed limit*, Physical Review Letters **103**, 24 (2009), p. 240501. <https://doi.org/10.1103/physrevlett.103.240501>

[37] F. Carbonell, J. C. Jímenez, L. M. Pedroso. *Computing multiple integrals involving matrix exponentials*, Journal of Computational and Applied Mathematics **213**, 1 (2008), pp. 300–305. <https://doi.org/10.1016/j.cam.2007.01.007>

[38] A. J. Carpenter, A. Ruttan, R. S. Varga. *Rational Approximation and Interpolation*, P. R. Graves-Morris, E. B. Saff, R. S. Varga, eds. (Springer, 1984), vol. 1105 of *Lecture Notes in Mathematics*. ISBN: 978-3-54-013899-0. <https://doi.org/10.1007/bfb0072427>

- [39] H. Y. Carr, E. M. Purcell. *Effects of diffusion on free precession in nuclear magnetic resonance experiments*, Physical Review **94**, 3 (1954), pp. 630–638. <https://doi.org/10.1103/physrev.94.630>
- [40] C. J. Cerjan, W. H. Miller. *On finding transition states*, The Journal of Chemical Physics **75**, 6 (1981), pp. 2800–2806. <https://doi.org/10.1063/1.442352>
- [41] G. Ciaramella, A. Borzì, G. Dirr, D. Wachsmuth. *Newton methods for the optimal control of closed quantum spin systems*, SIAM Journal on Scientific Computing **37**, 1 (2015), pp. A319–A346. <https://doi.org/10.1137/140966988>
- [42] W. Cody, G. Meinardus, R. Varga. *Chebyshev rational approximations to $\exp(-x)$ in $[0, +\infty]$ and applications to heat-conduction problems*, Journal of Approximation Theory **2**, 1 (1969), pp. 50–65. [https://doi.org/10.1016/0021-9045\(69\)90030-6](https://doi.org/10.1016/0021-9045(69)90030-6)
- [43] S. Conolly, D. Nishimura, A. Macovski. *Optimal control solutions to the magnetic resonance selective excitation problem*, IEEE Transactions on Medical Imaging **5**, 2 (1986), pp. 106–115. <https://doi.org/10.1109/tmi.1986.4307754>
- [44] P. A. S. Cruickshank, D. R. Bolton, D. A. Robertson, R. I. Hunter, R. J. Wylde, G. M. Smith. *A kilowatt pulsed 94 GHz electron paramagnetic resonance spectrometer with high concentration sensitivity, high instantaneous bandwidth, and low dead time*, Review of Scientific Instruments **80**, 10 (2009), p. 103102. <https://doi.org/10.1063/1.3239402>
- [45] D. D'Alessandro. *Introduction to Quantum Control and Dynamics* (Chapman and Hall/CRC, 2007). ISBN: 978-1-58-488884-0
- [46] W. C. Davidon. *Variable metric method for minimization*. Tech. rep., Argonne National Lab., IL, USA (1959). <https://doi.org/10.2172/4222000>
- [47] W. C. Davidon. *Variance algorithm for minimization*, The Computer Journal **10**, 4 (1968), pp. 406–410. <https://doi.org/10.1093/comjnl/10.4.406>
- [48] W. C. Davidon. *Variable metric method for minimization*, SIAM Journal on Optimization **1**, 1 (1991), pp. 1–17. <https://doi.org/10.1137/0801001>
- [49] M. F. Dempsey, B. Condon, D. M. Hadley. *MRI safety review*, Seminars in Ultrasound, CT and MRI **23**, 5 (2002), pp. 392–401. [https://doi.org/10.1016/S0887-2171\(02\)90010-7](https://doi.org/10.1016/S0887-2171(02)90010-7)

- [50] J. E. Dennis Jr., V. Torczon. *Direct search methods on parallel machines*, SIAM Journal on Optimization **1**, 4 (1991), pp. 448–474. <https://doi.org/10.1137/0801027>
- [51] J. E. Dennis Jr, D. J. Woods. *Optimization on microcomputers: The Nelder-Mead simplex algorithm*. Tech. rep., Rice University, Houston, TX, USA (1987). <https://doi.org/10.21236/ada453814>
- [52] A. E. Derome. *Modern NMR techniques for chemistry research* (Pergamon, 1987). ISBN: 978-0-08-032513-2
- [53] S. J. DeVience, R. L. Walsworth, M. S. Rosen. *Preparation of nuclear spin singlet states using spin-lock induced crossing*, Physical Review Letters **111**, 17 (2013), p. 173002. <https://doi.org/10.1103/physrevlett.111.173002>
- [54] L. Dieci, A. Papini. *Padé approximation for the exponential of a block triangular matrix*, Linear Algebra and its Applications **308**, 1–3 (2000), pp. 183–202. [https://doi.org/10.1016/s0024-3795\(00\)00042-2](https://doi.org/10.1016/s0024-3795(00)00042-2)
- [55] P. A. M. Dirac. *The Principles of Quantum Mechanics* (Oxford University Press, 1930). ISBN: 978-0-19-852011-5
- [56] F. Dolde, V. Bergholm, Y. Wang, I. Jakobi, B. Naydenov, S. Pezzagna, J. Meijer, F. Jelezko, P. Neumann, T. Schulte-Herbrüggen, J. Biamonte, J. Wrachtrup. *High-fidelity spin entanglement using optimal control*, Nature Communications **5** (2014), p. 3371. <https://doi.org/10.1038/ncomms4371>
- [57] A. Doll, G. Jeschke. *Fourier-transform electron spin resonance with bandwidth-compensated chirp pulses*, Journal of Magnetic Resonance **246** (2014), pp. 18–26. <https://doi.org/10.1016/j.jmr.2014.06.016>
- [58] A. Doll, S. Pribitzer, R. Tschaggelar, G. Jeschke. *Adiabatic and fast passage ultra-wideband inversion in pulsed EPR*, Journal of Magnetic Resonance **230** (2013), pp. 27–39. <https://doi.org/10.1016/j.jmr.2013.01.002>
- [59] P. Doria, T. Calarco, S. Montangero. *Optimal control technique for many-body quantum dynamics*, Physical Review Letters **106**, 19 (2011), p. 190501. <https://doi.org/10.1103/physrevlett.106.190501>
- [60] J. C. Doyle, B. A. Francis, A. R. Tannenbaum. *Feedback Control Theory* (Dover Publications, 2009). ISBN: 978-0-48-646933-1

- [61] R. S. Dumont, S. Jain, A. Bain. *Simulation of many-spin system dynamics via sparse matrix methodology*, Journal of Chemical Physics **106**, 14 (1997), pp. 5928–5936. <https://doi.org/10.1063/1.473258>
- [62] F. J. Dyson. *The radiation theories of Tomonaga, Schwinger, and Feynman*, Physical Review **75** (1949), pp. 486–502. <https://doi.org/10.1103/physrev.75.486>
- [63] F. J. Dyson. *The S matrix in quantum electrodynamics*, Physical Review **75** (1949), pp. 1736–1755. <https://doi.org/10.1103/physrev.75.1736>
- [64] M. Edén. *Computer simulations in solid-state NMR. III. Powder averaging*, Concepts in Magnetic Resonance Part A **18A**, 1 (2003), pp. 24–55. <https://doi.org/10.1002/cmr.a.10065>
- [65] M. Edén, M. H. Levitt. *Computation of orientational averages in solid-state NMR by gaussian spherical quadrature*, Journal of Magnetic Resonance **132**, 2 (1998), pp. 220–239. <https://doi.org/10.1006/jmre.1998.1427>
- [66] L. J. Edwards. *Simulation of coherence selection by pulsed field gradients in liquid-state NMR using an auxiliary matrix formalism*, Journal of Magnetic Resonance **240** (2014), pp. 95–101. <https://doi.org/10.1016/j.jmr.2014.01.011>
- [67] L. J. Edwards, I. Kuprov. *Parallel density matrix propagation in spin dynamics simulations*, Journal of Chemical Physics **136**, 4 (2012), p. 044108. <https://doi.org/10.1063/1.3679656>
- [68] L. J. Edwards, D. V. Savostyanov, A. A. Nevezorov, M. Concistre, G. Pileio, I. Kuprov. *Grid-free powder averages: On the applications of the Fokker–Planck equation to solid state NMR*, Journal of Magnetic Resonance **235** (2013), pp. 121–129. <https://doi.org/10.1016/j.jmr.2013.07.011>
- [69] D. J. Egger, F. K. Wilhelm. *Adaptive hybrid optimal quantum control for imprecisely characterized systems*, Physical Review Letters **112**, 24 (2014), p. 240503. <https://doi.org/10.1103/physrevlett.112.240503>
- [70] S. Ehni, B. Luy. *BEBE^{tr} and BUBI: J-compensated concurrent shaped pulses for ¹H–¹³C experiments*, Journal of Magnetic Resonance **232** (2013), pp. 7–17. <https://doi.org/10.1016/j.jmr.2013.04.007>

[71] A. Einstein, B. Podolsky, N. Rosen. *Can quantum-mechanical description of physical reality be considered complete?*, Physical Review **47**, 10 (1935), pp. 777–780. <https://doi.org/10.1103/physrev.47.777>

[72] R. Eitan, M. Mundt, D. J. Tannor. *Optimal control with accelerated convergence: Combining the Krotov and quasi-Newton methods*, Physical Review A **83**, 5 (2011), p. 053426. <https://doi.org/10.1103/physreva.83.053426>

[73] R. R. Ernst, G. Bodenhausen, A. Wokaun. *Principles of Nuclear Magnetic Resonance in One and Two Dimensions* (Clarendon Press, 1991). ISBN: 978-0-19-855647-3

[74] R. P. Feynman. *An operator calculus having applications in quantum electrodynamics*, Physical Review **84**, 1 (1951), pp. 108–128. <https://doi.org/10.1103/physrev.84.108>

[75] R. Fisher, F. Helmer, S. J. Glaser, F. Marquardt, T. Schulte-Herbrüggen. *Optimal control of circuit quantum electrodynamics in one and two dimensions*, Physical Review B **81**, 8 (2010), p. 085328. <https://doi.org/10.1103/physrevb.81.085328>

[76] R. Fisher, H. Yuan, A. Spörl, S. J. Glaser. *Time-optimal generation of cluster states*, Physical Review A **79**, 4 (2009), p. 042304. <https://doi.org/10.1103/physreva.79.042304>

[77] R. M. Fisher. *Optimal Control of Multi-Level Quantum Systems*. Ph.D. thesis, Technische Universität München (2010).

[78] R. Fletcher. *A new approach to variable metric algorithms*, The Computer Journal **13**, 3 (1970), pp. 317–322. <https://doi.org/10.1093/comjnl/13.3.317>

[79] R. Fletcher. *Practical Methods of Optimization*, 2nd edition (Wiley, 2000). ISBN: 978-0-47-149463-8

[80] R. Fletcher, M. J. D. Powell. *A rapidly convergent descent method for minimization*, The Computer Journal **6**, 2 (1963), pp. 163–168. <https://doi.org/10.1093/comjnl/6.2.163>

[81] F. F. Floether, P. de Fouquières, S. G. Schirmer. *Robust quantum gates for open systems via optimal control: Markovian versus non-Markovian dynamics*, New Journal of Physics **14**, 7 (2012), p. 073023. <https://doi.org/10.1088/1367-2630/14/7/073023>

[82] G. Floquet. *Sur les équations différentielles linéaires à coefficients périodiques*, Annales Scientifiques de l'École Normale Supérieure **12**, 1883 (1883), pp. 47–88. <https://doi.org/10.24033/asens.220>

[83] A. D. Fokker. *Die mittlere energie rotierender elektrischer dipole im strahlungsfeld*, Annalen der Physik **348**, 5 (1914), pp. 810–820. <https://doi.org/10.1002/andp.19143480507>

[84] P. de Fouquières. *Implementing quantum gates by optimal control with doubly exponential convergence*, Physical Review Letters **108**, 11 (2012), p. 110504. <https://doi.org/10.1103/physrevlett.108.110504>

[85] P. de Fouquières, S. G. Schirmer. *A closer look at quantum control landscapes and their implication for control optimization*, Infinite Dimensional Analysis, Quantum Probability and Related Topics **16**, 3 (2013), p. 1350021. <https://doi.org/10.1142/s0219025713500215>

[86] P. de Fouquières, S. G. Schirmer, S. J. Glaser, I. Kuprov. *Second order gradient ascent pulse engineering*, Journal of Magnetic Resonance **212**, 2 (2011), pp. 412–417. <https://doi.org/10.1016/j.jmr.2011.07.023>

[87] G. F. Franklin, J. D. Powell, A. Emami-Naeini. *Feedback Control of Dynamic Systems* (Addison-Wesley, 1994). ISBN: 978-0-20-152747-6

[88] R. A. Frazer, W. J. Duncan, A. R. Collar. *Elementary Matrices* (Cambridge University Press, 1947). ISBN: 978-0-52-109155-8

[89] R. Freeman. *Shaped radiofrequency pulses in high resolution NMR*, Progress in Nuclear Magnetic Resonance Spectroscopy **32**, 1 (1998), pp. 59–106. [https://doi.org/10.1016/s0079-6565\(97\)00024-1](https://doi.org/10.1016/s0079-6565(97)00024-1)

[90] L. Frydman, J. S. Harwood. *Isotropic spectra of half-integer quadrupolar spins from bidimensional magic-angle spinning NMR*, Journal of the American Chemical Society **117**, 19 (1995), pp. 5367–5368. <https://doi.org/10.1021/ja00124a023>

[91] E. Gallopoulos, Y. Saad. *Efficient solution of parabolic equations by Krylov approximation methods*, SIAM Journal on Scientific and Statistical Computing **13**, 5 (1992), pp. 1236–1264. <https://doi.org/10.1137/0913071>

[92] R. V. Gamkrelidze. *Principles of Optimal Control Theory* (Springer, 1978).

[93] W. Gerlach, O. Stern. *Der experimentelle nachweis des magnetischen moments des silberatoms*, Zeitschrift für Physik **8**, 1 (1922), pp. 110–111. <https://doi.org/10.1007/bf01329580>

[94] N. I. Gershenzon, K. Kobzar, B. Luy, S. J. Glaser, T. E. Skinner. *Optimal control design of excitation pulses that accommodate relaxation*, Journal of Magnetic Resonance **188**, 2 (2007), pp. 330–336. <https://doi.org/10.1016/j.jmr.2007.08.007>

[95] P. E. Gill, W. Murray. *Newton-type methods for unconstrained and linearly constrained optimization*, Mathematical Programming **7**, 1 (1974), pp. 311–350. <https://doi.org/10.1007/bf01585529>

[96] P. E. Gill, W. Murray, M. H. Wright. *Practical Optimization* (Emerald Group Publishing Limited, 1982). ISBN: 978-0-12-283952-8

[97] P. Gilmore, C. T. Kelley. *An implicit filtering algorithm for optimization of functions with many local minima*, SIAM Journal on Optimization **5**, 2 (1995), pp. 269–285. <https://doi.org/10.1137/0805015>

[98] S. J. Glaser, U. Boscain, T. Calarco, C. P. Koch, W. Köckenberger, R. Kosloff, I. Kuprov, B. Luy, S. G. Schirmer, T. Schulte-Herbrüggen, D. Sugny, F. K. Wilhelm. *Training Schrödinger’s cat: Quantum optimal control*, European Physical Journal D **69**, 12 (2015), p. 279. <https://doi.org/10.1140/epjd/e2015-60464-1>

[99] S. J. Glaser, T. Schulte-Herbrüggen, M. Sieveking, O. Schedletzky, N. C. Nielsen, O. W. Sørensen, C. Griesinger. *Unitary control in quantum ensembles: Maximizing signal intensity in coherent spectroscopy*, Science **280**, 5362 (1998), pp. 421–424. <https://doi.org/10.1126/science.280.5362.421>

[100] D. Goldfarb. *A family of variable-metric methods derived by variational means*, Mathematics of Computation **24**, 109 (1970), pp. 23–26. <https://doi.org/10.2307/2004873>

[101] D. Goldfarb. *Curvilinear path steplength algorithms for minimization which use directions of negative curvature*, Mathematical Programming **18**, 1 (1980), pp. 31–40. <https://doi.org/10.1007/bf01588294>

[102] S. M. Goldfeld, R. E. Quandt, H. F. Trotter. *Maximization by quadratic hill-climbing*, Econometrica **34**, 3 (1966), pp. 541–551. <https://doi.org/10.2307/1909768>

[103] M. Goldman. *Formal theory of spin-lattice relaxation*, Journal of Magnetic Resonance **149**, 2 (2001), pp. 160–187. <https://doi.org/10.1006/jmre.2000.2239>

[104] A. A. Goldstein. *On steepest descent*, Journal of the SIAM Series A Control **3**, 1 (1965), pp. 147–151. <https://doi.org/10.1137/0303013>

[105] A. A. Goldstein, J. F. Price. *An effective algorithm for minimization*, Numerische Mathematik **10**, 3 (1967), pp. 184–189. <https://doi.org/10.1007/bf02162162>

[106] G. H. Golub, C. F. Van Loan. *Matrix Computations*, 4th edition (John Hopkins University Press, 2013). ISBN: 978-1-42-140794-4

[107] D. L. Goodwin, I. Kuprov. *Auxiliary matrix formalism for interaction representation transformations, optimal control, and spin relaxation theories*, The Journal of Chemical Physics **143**, 8 (2015), p. 084113. <https://doi.org/10.1063/1.4928978>

[108] D. L. Goodwin, I. Kuprov. *Modified Newton-Raphson GRAPE methods for optimal control of spin systems*, The Journal of Chemical Physics **144**, 20 (2016), p. 204107. <https://doi.org/10.1063/1.4949534>

[109] D. L. Goodwin, W. K. Myers, C. R. Timmel, I. Kuprov. *Feedback control optimisation of ESR experiments*. arXiv:1707.05184 [physics.chem-ph] (2017).

[110] P. L. Gor'kov, E. Y. Chekmenev, C. Li, M. Cotten, J. J. Buffy, N. J. Traaaseth, G. Veglia, W. W. Brey. *Using low- E resonators to reduce RF heating in biological samples for static solid-state NMR up to 900MHz*, Journal of Magnetic Resonance **185**, 1 (2007), pp. 77–93. <https://doi.org/10.1016/j.jmr.2006.11.008>

[111] T. Gradl, A. Spörl, T. Huckle, S. J. Glaser, T. Schulte-Herbrüggen. *Euro-Par 2006 Parallel Processing*, W. E. Nagel, W. V. Walter, W. Lehner, eds., Lecture Notes in Computer Science (Springer, 2006). ISBN: 978-3-54-037783-2. https://doi.org/10.1007/11823285_78

[112] J. Greenstadt. *On the relative efficiencies of gradient methods*, Mathematics of Computation **21**, 99 (1967), pp. 360–367. <https://doi.org/10.2307/2003238>

[113] C. Griesinger, M. Bennati, H. M. Vieth, C. Luchinat, G. Parigi, P. Höfer, F. Engelke, S. J. Glaser, V. Denysenkov, T. F. Prisner. *Dynamic nuclear*

polarization at high magnetic fields in liquids, Progress in Nuclear Magnetic Resonance Spectroscopy **64** (2012), pp. 4–28. <https://doi.org/10.1016/j.pnmrs.2011.10.002>

[114] P. Groszkowski, A. G. Fowler, F. Motzoi, F. K. Wilhelm. *Tunable coupling between three qubits as a building block for a superconducting quantum computer*, Physical Review B **84**, 14 (2011), p. 144516. <https://doi.org/10.1103/physrevb.84.144516>

[115] R. Haberkorn. *Density matrix description of spin-selective radical pair reactions*, Molecular Physics **32**, 5 (1976), pp. 1491–1493. <https://doi.org/10.1080/00268977600102851>

[116] U. Haeberlen, J. Waugh. *Coherent averaging effects in magnetic resonance*, Physical Review **175**, 2 (1968), pp. 453–467. <https://doi.org/10.1103/physrev.175.453>

[117] E. L. Hahn. *Spin echoes*, Physical Review **80**, 4 (1950), pp. 580–594. <https://doi.org/10.1103/physrev.80.580>

[118] I. M. Haies, J. A. Jarvis, H. Bentley, I. Heinmaa, I. Kuprov, P. T. F. Williamson, M. Carravetta. *¹⁴N overtone NMR under MAS: signal enhancement using symmetry-based sequences and novel simulation strategies*, Physical Chemistry Chemical Physics **17**, 9 (2015), pp. 6577–6587. <https://doi.org/10.1039/c4cp03994g>

[119] J. O. Hansen, C. Kehlet, M. Bjerring, T. Vosegaard, S. J. Glaser, N. Khaneja, N. C. Nielsen. *Optimal control based design of composite dipolar recoupling experiments by analogy to single-spin inversion pulses*, Chemical Physics Letters **447**, 1–3 (2007), pp. 154–161. <https://doi.org/10.1016/j.cplett.2007.08.072>

[120] F. Hausdorff. *Die symbolische exponentialformel in der gruppentheorie*, Berichte der Sächsischen Akademie der Wissenschaften (Math. Phys. Klasse) **58** (1906), pp. 19–48.

[121] M. D. Hebden. *An Algorithm for Minimization using Exact Second Derivatives* (Technical Report, T.P. 515, AERA Theoretical Physics Division, Harwell, Oxfordshire, UK, 1973).

[122] W. Heisenberg. *Über den anschaulichen inhalt der quantentheoretischen kinematik und mechanik*, Zeitschrift für Physik **43**, 3–4 (1927), pp. 172–198. <https://doi.org/10.1007/bf01397280>

[123] M. Hellgren, E. Räsänen, E. K. U. Gross. *Optimal control of strong-field ionization with time-dependent density-functional theory*, Physical Review A **88**, 1 (2013), p. 013414. <https://doi.org/10.1103/physreva.88.013414>

[124] M. R. Hestenes, E. Stiefel. *Methods of conjugate gradients for solving linear systems*, Journal of Research of the National Bureau of Standards **49**, 6 (1952), p. 409. <https://doi.org/10.6028/jres.049.044>

[125] N. J. Higham. *Optimization by direct search in matrix computations*, SIAM Journal on Matrix Analysis and Applications **14**, 2 (1993), pp. 317–333. <https://doi.org/10.1137/0614023>

[126] N. J. Higham. *The scaling and squaring method for the matrix exponential revisited*, SIAM Journal on Matrix Analysis and Applications **26**, 4 (2005), pp. 1179–1193. <https://doi.org/10.1137/04061101x>

[127] N. J. Higham. *Functions of Matrices: Theory and Computation* (SIAM, 2008). ISBN: 978-0-89-871646-7

[128] M. Hochbruck, C. Lubich. *On Krylov subspace approximations to the matrix exponential operator*, SIAM Journal on Numerical Analysis **34**, 5 (1997), pp. 1911–1925. <https://doi.org/10.1137/s0036142995280572>

[129] J. S. Hodges, J. C. Yang, C. Ramanathan, D. G. Cory. *Universal control of nuclear spins via anisotropic hyperfine interactions*, Physical Review A **78**, 1 (2008), p. 010303. <https://doi.org/10.1103/physreva.78.010303>

[130] H. J. Hogben, M. Krzystyniak, G. T. P. Charnock, P. J. Hore, I. Kuprov. *Spinach - A software library for simulation of spin dynamics in large spin systems*, Journal of Magnetic Resonance **208**, 2 (2011), pp. 179–194. <https://doi.org/10.1016/j.jmr.2010.11.008>

[131] P. Hore. *Nuclear Magnetic Resonance*, 2nd edition (Oxford University Press, 1995). ISBN: 978-0-19-870341-9

[132] W. Huisenga, L. Pesce, R. Kosloff, P. Saalfrank. *Faber and Newton polynomial integrators for open-system density matrix propagation*, Journal of Chemical Physics **110**, 12 (1999), pp. 5538–5547. <https://doi.org/10.1063/1.478451>

[133] Y. Iwasa. *HTS and NMR/MRI magnets: Unique features, opportunities, and challenges*, Physica C: Superconductivity and its Applications **445–448** (2006), pp. 1088–1094. <https://doi.org/10.1016/j.physc.2006.05.040>

[134] M. A. Janich, M. A. McLean, R. Noeske, S. J. Glaser, R. F. Schulte. *Slice-selective broadband refocusing pulses for the robust generation of crushed spin-echoes*, Journal of Magnetic Resonance **223** (2012), pp. 129–137. <https://doi.org/10.1016/j.jmr.2012.08.003>

[135] M. A. Janich, R. F. Schulte, M. Schwaiger, S. J. Glaser. *Robust slice-selective broadband refocusing pulses*, Journal of Magnetic Resonance **213**, 1 (2011), pp. 126–135. <https://doi.org/10.1016/j.jmr.2011.09.025>

[136] J. A. Jarvis, I. M. Haies, P. T. F. Williamson, M. Caravetta. *An efficient NMR method for the characterisation of ^{14}N sites through indirect ^{13}C detection*, Physical Chemistry Chemical Physics **15**, 20 (2013), p. 7613. <https://doi.org/10.1039/c3cp50787d>

[137] G. Jeschke, S. Pribitzer, A. Doll. *Coherence transfer by passage pulses in electron paramagnetic resonance spectroscopy*, The Journal of Physical Chemistry B **119**, 43 (2015), pp. 13570–13582. <https://doi.org/10.1021/acs.jpcb.5b02964>

[138] J. R. Johansson, P. D. Nation, F. Nori. *QuTiP: An open-source Python framework for the dynamics of open quantum systems*, Computer Physics Communications **183**, 8 (2012), pp. 1760–1772. <https://doi.org/10.1016/j.cpc.2012.02.021>

[139] J. R. Johansson, P. D. Nation, F. Nori. *QuTiP 2: A Python framework for the dynamics of open quantum systems*, Computer Physics Communications **184**, 4 (2013), pp. 1234–1240. <https://doi.org/10.1016/j.cpc.2012.11.019>

[140] J. A. Jones, P. J. Hore. *Spin-selective reactions of radical pairs act as quantum measurements*, Chemical Physics Letters **488**, 1–3 (2010), pp. 90–93. <https://doi.org/10.1016/j.cplett.2010.01.063>

[141] R. S. Judson, K. K. Lehmann, H. Rabitz, W. S. Warren. *Optimal design of external fields for controlling molecular motion: Application to rotation*, Journal of Molecular Structure **223** (1990), pp. 425–456. [https://doi.org/10.1016/0022-2860\(90\)80485-3](https://doi.org/10.1016/0022-2860(90)80485-3)

[142] R. S. Judson, H. Rabitz. *Teaching lasers to control molecules*, Physical Review Letters **68**, 10 (1992), pp. 1500–1503. <https://doi.org/10.1103/physrevlett.68.1500>

[143] C. G. Källström. *Computing $\exp(A)$ and $\int \exp(As)ds$* . Tech. rep., Lund Institute of Technology, Division of Automatic Control (1973).

[144] R. Karplus, J. Schwinger. *A note on saturation in microwave spectroscopy*, Physical Review **73**, 9 (1948), pp. 1020–1026. <https://doi.org/10.1103/physrev.73.1020>

[145] T. Kaufmann, T. J. Keller, J. M. Franck, R. P. Barnes, S. J. Glaser, J. M. Martinis, S. Han. *DAC-board based X-band EPR spectrometer with arbitrary waveform control*, Journal of Magnetic Resonance **235** (2013), pp. 95–108. <https://doi.org/10.1016/j.jmr.2013.07.015>

[146] C. Kehlet, M. Bjerring, A. C. Sivertsen, T. Kristensen, J. J. Enghild, S. J. Glaser, N. Khaneja, N. C. Nielsen. *Optimal control based NCO and NCA experiments for spectral assignment in biological solid-state NMR spectroscopy*, Journal of Magnetic Resonance **188**, 2 (2007), pp. 216–230. <https://doi.org/10.1016/j.jmr.2007.06.011>

[147] C. Kehlet, T. Vosegaard, N. Khaneja, S. J. Glaser, N. C. Nielsen. *Low-power homonuclear dipolar recoupling in solid-state NMR developed using optimal control theory*, Chemical Physics Letters **414**, 1–3 (2005), pp. 204–209. <https://doi.org/10.1016/j.cplett.2005.08.061>

[148] C. T. Kelley. *Detection and Remediation of Stagnation in the Nelder–Mead Algorithm Using a Sufficient Decrease Condition*, SIAM Journal on Optimization **10**, 1 (1999), pp. 43–55. <https://doi.org/10.1137/s1052623497315203>

[149] C. S. Kenney, A. J. Laub. *A Schur–Fréchet algorithm for computing the logarithm and exponential of a matrix*, SIAM Journal on Matrix Analysis and Applications **19**, 3 (1998), pp. 640–663. <https://doi.org/10.1137/s0895479896300334>

[150] N. Khaneja. *Switched control of electron nuclear spin systems*, Physical Review A **76**, 3 (2007), p. 032326. <https://doi.org/10.1103/physreva.76.032326>

[151] N. Khaneja. *Chirp excitation*, Journal of Magnetic Resonance **282** (2017), pp. 32–36. <https://doi.org/10.1016/j.jmr.2017.07.003>

[152] N. Khaneja, R. Brockett, S. J. Glaser. *Time optimal control in spin systems*, Physical Review A **63**, 3 (2001), p. 032308. <https://doi.org/10.1103/physreva.63.032308>

[153] N. Khaneja, S. J. Glaser, R. Brockett. *Sub-Riemannian geometry and time optimal control of three spin systems: Quantum gates and coherence transfer*, Physical Review A **65**, 3 (2002), p. 032301. <https://doi.org/10.1103/physreva.65.032301>

[154] N. Khaneja, T. O. Reiss, C. Kehlet, T. Schulte-Herbrüggen, S. J. Glaser. *Optimal control of coupled spin dynamics: design of NMR pulse sequences by gradient ascent algorithms*, Journal of Magnetic Resonance **172**, 2 (2005), pp. 296–305. <https://doi.org/10.1016/j.jmr.2004.11.004>

[155] B. Khani, J. M. Gambetta, F. Motzoi, F. K. Wilhelm. *Optimal generation of Fock states in a weakly nonlinear oscillator*, Physica Scripta **T137** (2009), p. 014021. <https://doi.org/10.1088/0031-8949/2009/t137/014021>

[156] B. Khani, S. T. Merkel, F. Motzoi, J. M. Gambetta, F. K. Wilhelm. *High-fidelity quantum gates in the presence of dispersion*, Physical Review A **85**, 2 (2012), p. 022306. <https://doi.org/10.1103/physreva.85.022306>

[157] K. Kobzar, S. Ehni, T. E. Skinner, S. J. Glaser, B. Luy. *Exploring the limits of broadband 90° and 180° universal rotation pulses*, Journal of Magnetic Resonance **225** (2012), pp. 142–160. <https://doi.org/10.1016/j.jmr.2012.09.013>

[158] K. Kobzar, B. Luy, N. Khaneja, S. J. Glaser. *Pattern pulses: Design of arbitrary excitation profiles as a function of pulse amplitude and offset*, Journal of Magnetic Resonance **173**, 2 (2005), pp. 229–235. <https://doi.org/10.1016/j.jmr.2004.12.005>

[159] K. Kobzar, T. E. Skinner, N. Khaneja, S. J. Glaser, B. Luy. *Exploring the limits of broadband excitation and inversion pulses*, Journal of Magnetic Resonance **170**, 2 (2004), pp. 236–243. <https://doi.org/10.1016/j.jmr.2004.06.017>

[160] K. Kobzar, T. E. Skinner, N. Khaneja, S. J. Glaser, B. Luy. *Exploring the limits of broadband excitation and inversion: II. Rf-power optimized pulses*, Journal of Magnetic Resonance **194**, 1 (2008), pp. 58–66. <https://doi.org/10.1016/j.jmr.2008.05.023>

[161] S. S. Köcher, T. Heydenreich, S. J. Glaser. *Visualization and analysis of modulated pulses in magnetic resonance by joint time-frequency representations*, Journal of Magnetic Resonance **249** (2014), pp. 63–71. <https://doi.org/10.1016/j.jmr.2014.10.004>

[162] M. R. M. Koos, H. Feyrer, B. Luy. *Broadband excitation pulses with variable RF amplitude-dependent flip angle (RADFA)*, Magnetic Resonance in Chemistry **53**, 11 (2015), pp. 886–893. <https://doi.org/10.1002/mrc.4297>

[163] V. F. Krotov. *Global Methods in Optimal Control Theory* (CRC Press, 1995). ISBN: 978-0-82479329-6

[164] R. Kubo. *Generalized cumulant expansion method*, Journal of the Physical Society of Japan **17**, 7 (1962), pp. 1100–1120. <https://doi.org/10.1143/jpsj.17.1100>

[165] I. Kuprov. *Polynomially scaling spin dynamics II: further state-space compression using Krylov subspace techniques and zero track elimination*, Journal of Magnetic Resonance **195**, 1 (2008), pp. 45–51. <https://doi.org/10.1016/j.jmr.2008.08.008>

[166] I. Kuprov. *Diagonalization-free implementation of spin relaxation theory for large spin systems*, Journal of Magnetic Resonance **209**, 1 (2011), pp. 31–38. <https://doi.org/10.1016/j.jmr.2010.12.004>

[167] I. Kuprov. *Spin system trajectory analysis under optimal control pulses*, Journal of Magnetic Resonance **233** (2013), pp. 107–112. <https://doi.org/10.1016/j.jmr.2013.02.012>

[168] I. Kuprov. *Fokker-Planck formalism in magnetic resonance simulations*, Journal of Magnetic Resonance **270** (2016), pp. 124–135. <https://doi.org/10.1016/j.jmr.2016.07.005>

[169] I. Kuprov, C. T. Rodgers. *Derivatives of spin dynamics simulations*, Journal of Chemical Physics **131**, 23 (2009), p. 234108. <https://doi.org/10.1063/1.3267086>

[170] I. Kuprov, N. Wagner-Rundell, P. J. Hore. *Polynomially scaling spin dynamics simulation algorithm based on adaptive state-space restriction*, Journal of Magnetic Resonance **189**, 2 (2007), pp. 241–250. <https://doi.org/10.1016/j.jmr.2007.09.014>

[171] J. C. Lagarias, J. A. Reeds, M. H. Wright, P. E. Wright. *Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions*, SIAM Journal on Optimization **9**, 1 (1998), pp. 112–147. <https://doi.org/10.1137/s1052623496303470>

[172] E. N. Laguerre. *Le calcul des systèmes linéaires, extrait d'une lettre adressé à M. Hermite*, Journal de l'École polytechnique **25** (1867), pp. 215–264.

[173] C. Lanczos. *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, Journal of Research of the National Bureau of Standards **45**, 4 (1950), p. 255. <https://doi.org/10.6028/jres.045.026>

[174] L. D. Landau, E. M. Lifshitz. *Mechanics*, 3rd edition (Butterworth-Heinemann, 1976). ISBN: 978-0-75-062896-9

[175] L. D. Landau, E. M. Lifshitz. *Quantum Mechanics: Non-Relativistic Theory*, 3rd edition (Butterworth-Heinemann, 1981). ISBN: 978-0-08-029140-6

[176] M. Lapert, Y. Zhang, M. Braun, S. J. Glaser, D. Sugny. *Singular extremals for the time-optimal control of dissipative spin-1/2 particles*, Physical Review Letters **104**, 8 (2010), p. 083001. <https://doi.org/10.1103/physrevlett.104.083001>

[177] L. S. Lasdon, S. K. Mitter, A. D. Waren. *The conjugate gradient method for optimal control problems*, IEEE Transactions on Automatic Control **12**, 2 (1967), pp. 132–138. <https://doi.org/10.1109/tac.1967.1098538>

[178] C. Laustsen, S. Bowen, M. S. Vinding, N. C. Nielsen, J. H. Ardenkjær-Larsen. *Storage of magnetization as singlet order by optimal control designed pulses*, Magnetic Resonance in Medicine **71**, 3 (2014), pp. 921–926. <https://doi.org/10.1002/mrm.24733>

[179] P. C. Lauterbur. *Image formation by induced local interactions: Examples employing nuclear magnetic resonance*, Nature **242**, 5394 (1973), pp. 190–191. <https://doi.org/10.1038/242190a0>

[180] J. D. Lawson. *Generalized Runge-Kutta processes for stable systems with large Lipschitz constants*, SIAM Journal on Numerical Analysis **4**, 3 (1967), pp. 372–380. <https://doi.org/10.1137/0704033>

[181] V. I. Lebedev. *Values of the nodes and weights of ninth to seventeenth order Gauss-Markov quadrature formulae invariant under the octahedron group with inversion*, USSR Computational Mathematics and Mathematical Physics **15**, 1 (1975), pp. 44–51. [https://doi.org/10.1016/0041-5553\(75\)90133-0](https://doi.org/10.1016/0041-5553(75)90133-0)

[182] V. I. Lebedev. *Quadratures on a sphere*, USSR Computational Mathematics and Mathematical Physics **16**, 2 (1976), pp. 10–24. [https://doi.org/10.1016/0041-5553\(76\)90100-2](https://doi.org/10.1016/0041-5553(76)90100-2)

[183] J. S. Lee, R. R. Regatte, A. Jerschow. *Optimal nuclear magnetic resonance excitation schemes for the central transition of a spin 3/2 in the presence of residual quadrupolar coupling*, Journal of Chemical Physics **129**, 22 (2008), p. 224510. <https://doi.org/10.1063/1.3036005>

[184] J. S. Lee, R. R. Regatte, A. Jerschow. *Selective detection of ordered sodium signals by a jump-and-return pulse sequence*, Journal of Magnetic Resonance **200**, 1 (2009), pp. 126–129. <https://doi.org/10.1016/j.jmr.2009.06.015>

[185] J.-S. Lee, R. R. Regatte, A. Jerschow. *Optimal control NMR differentiation between fast and slow sodium*, Chemical Physics Letters **494**, 4–6 (2010), pp. 331–336. <https://doi.org/10.1016/j.cplett.2010.06.019>

[186] J. S. Lee, D. Xia, G. Madelin, R. R. Regatte. *Sodium inversion recovery MRI on the knee joint at 7T with an optimal control pulse*, Journal of Magnetic Resonance **262** (2016), pp. 33–41. <https://doi.org/10.1016/j.jmr.2015.12.002>

[187] M. Leskes, P. K. Madhu, S. Vega. *Floquet theory in solid-state nuclear magnetic resonance*, Progress in Nuclear Magnetic Resonance Spectroscopy **57**, 4 (2010), pp. 345–380. <https://doi.org/10.1016/j.pnmrs.2010.06.002>

[188] T. O. Levante, M. Baldus, B. H. Meier, R. R. Ernst. *Formalized quantum mechanical Floquet theory and its application to sample spinning in nuclear magnetic resonance*, Molecular Physics **86**, 5 (1995), pp. 1195–1212. <https://doi.org/10.1080/00268979500102671>

[189] T. O. Levante, T. Bremi, R. R. Ernst. *Pulse-sequence optimization with analytical derivatives. Application to deuterium decoupling in oriented phases*, Journal of Magnetic Resonance, Series A **121**, 2 (1996), pp. 167–177. <https://doi.org/10.1006/jmra.1996.0157>

[190] M. H. Levitt. *Symmetry constraints on spin dynamics: Application to hyperpolarized NMR*, Journal of Magnetic Resonance **262** (2016), pp. 91–99. <https://doi.org/10.1016/j.jmr.2015.08.021>

[191] F. L. Lewis, D. L. Vrabie, V. L. Syrmos. *Optimal Control*, 3rd edition (Wiley, 2012). ISBN: 978-0-47-063349-6

[192] J. S. Li, N. Khaneja. *Control of inhomogeneous quantum ensembles*, Physical Review A **73**, 3 (2006), p. 030302. <https://doi.org/10.1103/physreva.73.030302>

[193] M. L. Liou. *A novel method of evaluating transient response*, Proceedings of the IEEE **54**, 1 (1966), pp. 20–23. <https://doi.org/10.1109/proc.1966.4569>

[194] D. C. Liu, J. Nocedal. *On the limited memory BFGS method for large scale optimization*, Mathematical Programming **45**, 1–3 (1989), pp. 503–528. <https://doi.org/10.1007/bf01589116>

[195] C. F. Van Loan. *The sensitivity of the matrix exponential*, SIAM Journal on Numerical Analysis **14**, 6 (1977), pp. 971–981. <https://doi.org/10.1137/0714065>

[196] C. F. Van Loan. *Computing integrals involving the matrix exponential*, IEEE Transactions on Automatic Control **23**, 3 (1978), pp. 395–404. <https://doi.org/10.1109/tac.1978.1101743>

[197] D. J. Lurie. *A systematic design procedure for selective pulses in NMR imaging*, Magnetic Resonance Imaging **3**, 3 (1985), pp. 235–243. [https://doi.org/10.1016/0730-725x\(85\)90352-2](https://doi.org/10.1016/0730-725x(85)90352-2)

[198] B. Luy, K. Kobzar, T. E. Skinner, N. Khaneja, S. J. Glaser. *Construction of universal rotations from point-to-point transformations*, Journal of Magnetic Resonance **176**, 2 (2005), pp. 179–186. <https://doi.org/10.1016/j.jmr.2005.06.002>

[199] S. Machnes, E. Assémat, D. J. Tannor, F. K. Wilhelm. *Tunable, flexible, and efficient optimization of control pulses for practical qubits*, Physical Review Letters **120**, 15 (2018), p. 150401. <https://doi.org/10.1103/physrevlett.120.150401>

[200] S. Machnes, U. Sander, S. J. Glaser, P. de Fouquières, A. Gruslys, S. G. Schirmer, T. Schulte-Herbrüggen. *Comparing, optimizing, and benchmarking quantum-control algorithms in a unifying programming framework*, Physical Review A **84**, 2 (2011), p. 022305. <https://doi.org/10.1103/physreva.84.022305>

[201] C. MacLaurin. *A Treatise on Fluxions* (Forgotten Books, 1742).
ISBN: 978-0-28-290169-1

[202] Y. Maday, G. Turinici. *New formulations of monotonically convergent quantum control algorithms*, The Journal of Chemical Physics **118**, 18 (2003), pp. 8191–8196. <https://doi.org/10.1063/1.1564043>

[203] W. Magnus. *On the exponential solution of differential equations for a linear operator*, Communications on Pure and Applied Mathematics **7**, 4 (1954), pp. 649–673. <https://doi.org/10.1002/cpa.3160070404>

[204] D. W. Marquardt. *An algorithm for least-squares estimation of nonlinear parameters*, Journal of the Society for Industrial and Applied Mathematics **11**, 2 (1963), pp. 431–441. <https://doi.org/10.1137/0111030>

[205] A. Massire, M. A. Cloos, A. Vignaud, D. Le Bihan, A. Amadon, N. Boulant. *Design of non-selective refocusing pulses with phase-free rotation axis by gradient ascent pulse engineering algorithm in parallel transmission at 7T*, Journal of Magnetic Resonance **230** (2013), pp. 76–83. <https://doi.org/10.1016/j.jmr.2013.01.005>

[206] R. Mathias. *Approximation of matrix-valued functions*, SIAM Journal on Matrix Analysis and Applications **14**, 4 (1993), pp. 1061–1063. <https://doi.org/10.1137/0614070>

[207] I. I. Maximov, Z. Tošner, N. C. Nielsen. *Optimal control design of NMR and dynamic nuclear polarization experiments using monotonically convergent algorithms*, The Journal of Chemical Physics **128**, 18 (2008), p. 184505. <https://doi.org/10.1063/1.2903458>

[208] S. Meiboom, D. Gill. *Modified spin-echo method for measuring nuclear relaxation times*, Review of Scientific Instruments **29**, 8 (1958), pp. 688–691. <https://doi.org/10.1063/1.1716296>

[209] E. Merzbacher. *Quantum Mechanics*, 3rd edition (Wiley, 1997). ISBN: 978-0-47-188702-7

[210] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, E. Teller. *Equation of state calculations by fast computing machines*, The Journal of Chemical Physics **21**, 6 (1953), pp. 1087–1092. <https://doi.org/10.1063/1.1699114>

[211] C. Moler, C. Van Loan. *Nineteen dubious ways to compute the exponential of a matrix*, SIAM Review **20**, 4 (1978), pp. 801–836. <https://doi.org/10.1137/1020098>

[212] C. Moler, C. Van Loan. *Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later*, SIAM Review **45**, 1 (2003), pp. 3–49. <https://doi.org/10.1137/s00361445024180>

[213] M. J. Mombourquette, J. A. Weil. *Simulation of magnetic resonance powder spectra*, Journal of Magnetic Resonance (1969) **99**, 1 (1992), pp. 37–44. [https://doi.org/10.1016/0022-2364\(92\)90153-x](https://doi.org/10.1016/0022-2364(92)90153-x)

[214] J. J. Moré, D. C. Sorensen. *Newton's method*. Tech. rep., Argonne National Lab., IL, USA (1982). <https://doi.org/10.2172/5326201>

[215] J. J. Moré, D. C. Sorensen. *Computing a trust region step*, SIAM Journal on Scientific and Statistical Computing **4**, 3 (1983), pp. 553–572. <https://doi.org/10.1137/0904038>

[216] F. Motzoi, J. M. Gambetta, S. T. Merkel, F. K. Wilhelm. *Optimal control methods for rapidly time-varying Hamiltonians*, Physical Review A **84**, 2 (2011), p. 022307. <https://doi.org/10.1103/physreva.84.022307>

[217] F. Motzoi, J. M. Gambetta, P. Rebentrost, F. K. Wilhelm. *Simple pulses for elimination of leakage in weakly nonlinear qubits*, Physical Review Letters **103**, 11 (2009), p. 110501. <https://doi.org/10.1103/physrevlett.103.110501>

[218] I. Najfeld, T. F. Havel. *Derivatives of the matrix exponential and their computation*, Advances in Applied Mathematics **16**, 3 (1995), pp. 321–375. <https://doi.org/10.1006/aama.1995.1017>

[219] M. Ndong, H. Tal-Ezer, R. Kosloff, C. P. Koch. *A Chebychev propagator with iterative time ordering for explicitly time-dependent Hamiltonians*, Journal of Chemical Physics **132**, 6 (2010), p. 064105. <https://doi.org/10.1063/1.3312531>

[220] J. A. Nelder, R. Mead. *A simplex method for function minimization*, The Computer Journal **7**, 4 (1965), pp. 308–313. <https://doi.org/10.1093/comjnl/7.4.308>

[221] J. L. Neves, B. Heitmann, N. Khaneja, S. J. Glaser. *Heteronuclear decoupling by optimal tracking*, Journal of Magnetic Resonance **201**, 1 (2009), pp. 7–17. <https://doi.org/10.1016/j.jmr.2009.07.024>

[222] I. Newton. *Methodus Fluxionum et Serierum Infinitarum* (Cambridge University Press, 1671). ISBN: 978-0-52-104581-0

[223] A. B. Nielsen, M. Bjerring, J. T. Nielsen, N. C. Nielsen. *Symmetry-based dipolar recoupling by optimal control: Band-selective experiments for assignment of solid-state NMR spectra of proteins*, Journal of Chemical Physics **131**, 2 (2009), p. 025101. <https://doi.org/10.1063/1.3157737>

[224] M. A. Nielsen, I. L. Chuang. *Quantum Computation and Quantum Information* (Cambridge University Press, 2000). ISBN: 978-0-52-163503-5

[225] R. Nigmatullin, S. G. Schirmer. *Implementation of fault-tolerant quantum logic gates via optimal control*, New Journal of Physics **11** (2009), p. 105032. <https://doi.org/10.1088/1367-2630/11/10/105032>

[226] M. Nimbalkar, B. Luy, T. E. Skinner, J. L. Neves, N. I. Gershenzon, K. Kobzar, W. Bermel, S. J. Glaser. *The fantastic four: A plug 'n' play set of optimal control pulses for enhancing NMR spectroscopy*, Journal of Magnetic Resonance **228** (2013), pp. 16–31. <https://doi.org/10.1016/j.jmr.2012.12.007>

[227] M. Nimbalkar, R. Zeier, J. L. Neves, S. B. Elavarasi, H. Yuan, N. Khaneja, K. Dorai, S. J. Glaser. *Multiple-spin coherence transfer in linear Ising spin chains and beyond: Numerically optimized pulses and experiments*, Physical Review A **85**, 1 (2012), p. 012325. <https://doi.org/10.1103/physreva.85.012325>

[228] J. Nocedal. *Updating quasi-Newton matrices with limited storage*, Mathematics of Computation **35**, 151 (1980), pp. 773–782. <https://doi.org/10.2307/2006193>

[229] J. Nocedal, S. J. Wright. *Numerical Optimization*, 2nd edition (Springer, 2006). ISBN: 978-0-38-730303-1

[230] L. A. O'Dell, K. J. Harris, R. W. Schurko. *Optimized excitation pulses for the acquisition of static NMR powder patterns from half-integer quadrupolar nuclei*, Journal of Magnetic Resonance **203**, 1 (2010), pp. 156–166. <https://doi.org/10.1016/j.jmr.2009.12.016>

[231] L. A. O'Dell, C. I. Ratcliffe. *14N magic angle spinning overtone NMR spectra*, Chemical Physics Letters **514**, 1–3 (2011), pp. 168–173. <https://doi.org/10.1016/j.cplett.2011.08.030>

[232] L. A. O'Dell, C. I. Ratcliffe. *Crystal structure based design of signal enhancement schemes for solid-state NMR of insensitive half-integer quadrupolar nuclei*, Journal of Physical Chemistry A **115**, 5 (2011), pp. 747–752. <https://doi.org/10.1021/jp111531e>

[233] A. W. Overhauser. *Polarization of nuclei in metals*, Physical Review **92**, 2 (1953), pp. 411–415. <https://doi.org/10.1103/physrev.92.411>

[234] H. Padé. *Sur la représentation approchée d'une fonction par des fractions rationnelles*, Annales scientifiques de l'École Normale Supérieure **9** (1892), pp. 3–93. <https://doi.org/10.24033/asens.378>

[235] B. N. Parlett. *A recurrence among the elements of functions of triangular matrices*, Linear Algebra and its Applications **14**, 2 (1976), pp. 117–121. [https://doi.org/10.1016/0024-3795\(76\)90018-5](https://doi.org/10.1016/0024-3795(76)90018-5)

[236] W. Pauli Jr. *Zur quantenmechanik des magnetischen elektrons*, Zeitschrift für Physik **43**, 9–10 (1927), pp. 601–623. <https://doi.org/10.1007/bf01397326>

[237] A. N. Pechen, N. Il'In. *Trap-free manipulation in the Landau-Zener system*, Physical Review A **86**, 5 (2012), p. 052117. <https://doi.org/10.1103/physreva.86.052117>

[238] A. N. Pechen, D. J. Tannor. *Are there traps in quantum control landscapes?*, Physical Review Letters **106**, 12 (2011), p. 120402. <https://doi.org/10.1103/physrevlett.106.120402>

[239] A. N. Pechen, D. J. Tannor. *Quantum control landscape for a Λ -atom in the vicinity of second-order traps*, Israel Journal of Chemistry **52**, 5 (2012), pp. 467–472. <https://doi.org/10.1002/ijch.201100165>

[240] A. P. Peirce, M. A. Dahleh, H. Rabitz. *Optimal control of quantum-mechanical systems: Existence, numerical approximation, and applications*, Physical Review A **37**, 12 (1988), pp. 4950–4964. <https://doi.org/10.1103/physreva.37.4950>

[241] M. Planck. *Sitzungsberichte der Königlich Preussischen Akademie der Wissenschaften*, D. A. der Wissenschaften, ed. (Forgotten Books, 1917). ISBN: 978-0-42-849189-5

[242] Y. Polyhach, A. Godt, C. Bauer, G. Jeschke. *Spin pair geometry revealed by high-field DEER in the presence of conformational distributions*, Journal of

Magnetic Resonance **185**, 1 (2007), pp. 118–129. <https://doi.org/10.1016/j.jmr.2006.11.012>

[243] N. Pomplun, S. J. Glaser. *Exploring the limits of electron-nuclear polarization transfer efficiency in three-spin systems*, Physical Chemistry Chemical Physics **12**, 22 (2010), pp. 5791–5798. <https://doi.org/10.1039/c003751f>

[244] N. Pomplun, B. Heitmann, N. Khaneja, S. J. Glaser. *Optimization of electron-nuclear polarization transfer*, Applied Magnetic Resonance **34**, 3–4 (2008), pp. 331–346. <https://doi.org/10.1007/s00723-008-0124-6>

[245] A. Ponti. *Simulation of magnetic resonance static powder lineshapes: A quantitative assessment of spherical codes*, Journal of Magnetic Resonance **138**, 2 (1999), pp. 288–297. <https://doi.org/10.1006/jmre.1999.1758>

[246] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, E. F. Mishchenko. *The Mathematical Theory of Optimal Processes* (Pergamon Press, 1964). ISBN: 978-0-08-010176-7

[247] N. Rach, M. M. Müller, T. Calarco, S. Montangero. *Dressing the chopped-random-basis optimization: A bandwidth-limited access to the trap-free landscape*, Physical Review A **92**, 6 (2015), p. 062343. <https://doi.org/10.1103/physreva.92.062343>

[248] J. Raphson. *Analysis Aequationum Universalis* (Gale ECCO, Print Editions, 1690). ISBN: 978-1-17-102563-4

[249] P. Rebentrost, I. Serban, T. Schulte-Herbrüggen, F. K. Wilhelm. *Optimal control of a qubit coupled to a non-Markovian environment*, Physical Review Letters **102**, 9 (2009), p. 090401. <https://doi.org/10.1103/physrevlett.102.090401>

[250] A. G. Redfield. *On the theory of relaxation processes*, IBM Journal of Research and Development **1**, 1 (1957), pp. 19–31. <https://doi.org/10.1147/rd.11.0019>

[251] G. Riviello, C. Brif, R. Long, R. B. Wu, K. W. Moore Tibbetts, T. S. Ho, H. Rabitz. *Searching for quantum optimal control fields in the presence of singular critical points*, Physical Review A **90**, 1 (2014), p. 013404. <https://doi.org/10.1103/physreva.90.013404>

[252] H. H. Rosenbrock. *An automatic method for finding the greatest or least value of a function*, The Computer Journal **3**, 3 (1960), pp. 175–184. <https://doi.org/10.1093/comjnl/3.3.175>

[253] D. Rosenfeld, Y. Zur. *Design of adiabatic selective pulses using optimal control theory*, Magnetic Resonance in Medicine **36**, 3 (1996), pp. 401–409. <https://doi.org/10.1002/mrm.1910360311>

[254] L. G. Rowan, E. L. Hahn, W. B. Mims. *Electron-spin-echo envelope modulation*, Physical Review **137**, 1A (1965), pp. A61–A71. <https://doi.org/10.1103/physrev.137.a61>

[255] C. A. Ryan, C. Negrevergne, M. Laforest, E. Knill, R. Laflamme. *Liquid-state nuclear magnetic resonance as a testbed for developing quantum control methods*, Physical Review A **78**, 1 (2008), p. 012328. <https://doi.org/10.1103/physreva.78.012328>

[256] Y. Saad. *Analysis of some Krylov subspace approximations to the matrix exponential operator*, SIAM Journal on Numerical Analysis **29**, 1 (1992), pp. 209–228. <https://doi.org/10.1137/0729014>

[257] R. S. Said, J. Twamley. *Robust control of entanglement in a nitrogen-vacancy center coupled to a ¹³C nuclear spin in diamond*, Physical Review A **80**, 3 (2009), p. 032303. <https://doi.org/10.1103/physreva.80.032303>

[258] K. M. Salikhov, Y. E. Kand rashkin, A. K. Salikhov. *Peculiarities of free induction and primary spin echo signals for spin-correlated radical pairs*, Applied Magnetic Resonance **3**, 1 (1992), pp. 199–216. <https://doi.org/10.1007/BF03166790>

[259] B. C. Sanctuary. *Multipole operators for an arbitrary number of spins*, The Journal of Chemical Physics **64**, 11 (1976), pp. 4352–4361. <https://doi.org/10.1063/1.432104>

[260] S. A. Sarji, B. J. J. Abdullah, G. Kumar, A. H. Tan, P. Narayanan. *Failed magnetic resonance imaging examinations due to claustrophobia*, Australasian Radiology **42**, 4 (1998), pp. 293–295. <https://doi.org/10.1111/j.1440-1673.1998.tb00525.x>

[261] A. Savitzky, M. J. E. Golay. *Smoothing and differentiation of data by simplified least squares procedures*, Analytical Chemistry **36**, 8 (1964), pp. 1627–1639. <https://doi.org/10.1021/ac60214a047>

[262] D. V. Savostyanov, S. V. Dolgov, J. M. Werner, I. Kuprov. *Exact NMR simulation of protein-size spin systems using tensor train formalism*, Physical Review B **90**, 8 (2014), p. 085139. <https://doi.org/10.1103/physrevb.90.085139>

[263] J. C. Saywell, I. Kuprov, D. L. Goodwin, M. Carey, T. Freegarde. *Optimal control of mirror pulses for atom interferometry*. arXiv:1804.04625 [quant-ph] (2018).

[264] K. V. Schenker, D. Suter, A. Pines. *Broadband heteronuclear decoupling in the presence of homonuclear dipolar and quadrupolar interactions*, Journal of Magnetic Resonance (1969) **73**, 1 (1987), pp. 99–113. [https://doi.org/10.1016/0022-2364\(87\)90228-9](https://doi.org/10.1016/0022-2364(87)90228-9)

[265] O. Schiemann, P. Cekan, D. Margraf, T. F. Prisner, S. T. Sigurdsson. *Relative orientation of rigid Nitroxides by PELDOR: Beyond distance measurements in nucleic acids*, Angewandte Chemie International Edition **48**, 18 (2009), pp. 3292–3295. <https://doi.org/10.1002/anie.200805152>

[266] F. Schilling, S. J. Glaser. *Tailored real-time scaling of heteronuclear couplings*, Journal of Magnetic Resonance **223** (2012), pp. 207–218. <https://doi.org/10.1016/j.jmr.2012.07.024>

[267] S. G. Schirmer. *Implementation of quantum gates via optimal control*, Journal of Modern Optics **56**, 6 (2009), pp. 831–839. <https://doi.org/10.1080/09500340802344933>

[268] A. Schmidt, S. Vega. *The Floquet theory of nuclear magnetic resonance spectroscopy of single spins and dipolar coupled spin pairs in rotating solids*, The Journal of Chemical Physics **96**, 4 (1992), pp. 2655–2680. <https://doi.org/10.1063/1.462015>

[269] U. Schollwöck. *The density-matrix renormalization group*, Reviews of Modern Physics **77**, 1 (2005), pp. 259–315. <https://doi.org/10.1103/revmodphys.77.259>

[270] I. Scholz, J. D. Van Beek, M. Ernst. *Operator-based Floquet theory in solid-state NMR*, Solid State Nuclear Magnetic Resonance **37**, 3–4 (2010), pp. 39–59. <https://doi.org/10.1016/j.ssnmr.2010.04.003>

[271] J.-H. Schönfeldt, J. Twamley, S. Rebić. *Optimized control of Stark-shift-chirped rapid adiabatic passage in a Λ -type three-level system*, Physical Review A **80**, 4 (2009), p. 043401. <https://doi.org/10.1103/physreva.80.043401>

[272] P. Schöps, P. E. Spindler, A. Marko, T. F. Prisner. *Broadband spin echoes and broadband SIFTER in EPR*, Journal of Magnetic Resonance **250** (2015), pp. 55–62. <https://doi.org/10.1016/j.jmr.2014.10.017>

[273] E. Schrödinger. *Quantisierung als eigenwertproblem*, Annalen der Physik **385**, 13 (1926), pp. 437–490. <https://doi.org/10.1002/andp.19263851302>

[274] T. Schulte-Herbrüggen, S. J. Glaser, G. Dirr, U. Helmke. *Gradient flows for optimization in quantum information and quantum dynamics: foundations and applications*, Reviews in Mathematical Physics **22**, 06 (2010), pp. 597–667. <https://doi.org/10.1142/s0129055x10004053>

[275] T. Schulte-Herbrüggen, R. Marx, A. Fahmy, L. Kauffman, S. Lomonaco, N. Khaneja, S. J. Glaser. *Control aspects of quantum computing using pure and mixed states*, Philosophical Transactions of the Royal Society A **370**, 1976 (2012), pp. 4651–4670. <https://doi.org/10.1098/rsta.2011.0513>

[276] T. Schulte-Herbrüggen, A. Spörl, N. Khaneja, S. J. Glaser. *Optimal control-based efficient synthesis of building blocks of quantum algorithms: A perspective from network complexity towards time complexity*, Physical Review A **72**, 4 (2005), p. 042331. <https://doi.org/10.1103/physreva.72.042331>

[277] T. Schulte-Herbrüggen, A. Spörl, N. Khaneja, S. J. Glaser. *Optimal control for generating quantum gates in open dissipative systems*, Journal of Physics B **44**, 15 (2011), p. 154013. <https://doi.org/10.1088/0953-4075/44/15/154013>

[278] T. Schulte-Herbrüggen, A. Spörl, K. Waldherr, T. Gradl, S. J. Glaser, T. Huckle. *High Performance Computing in Science and Engineering, Garching/Munich 2007*, S. Wagner, M. Steinmetz, A. Bode, M. Brehm, eds. (Springer, 2009). ISBN: 978-3-54-069181-5. https://doi.org/10.1007/978-3-540-69182-2_41

[279] A. Schweiger, G. Jeschke. *Principles of Pulse Electron Paramagnetic Resonance* (Oxford University Press, 2001). ISBN: 978-0-19-850634-8

[280] D. F. Shanno. *Conditioning of quasi-Newton methods for function minimization*, Mathematics of Computation **24**, 111 (1970), pp. 647–656. <https://doi.org/10.2307/2004840>

[281] R. Shepard, I. Shavitt, J. Simons. *Comparison of the convergence characteristics of some iterative wave function optimization methods*, The Journal of Chemical Physics **76**, 1 (1982), pp. 543–557. <https://doi.org/10.1063/1.442701>

[282] J. H. Shirley. *Solution of the Schrödinger equation with a Hamiltonian periodic in time*, Physical Review **138**, 4B (1965), pp. B979–B987. <https://doi.org/10.1103/PhysRev.138.B979>

[283] M. S. Silver, R. I. Joseph, D. I. Hoult. *Highly selective $\pi/2$ and π pulse generation*, Journal of Magnetic Resonance (1969) **59**, 2 (1984), pp. 347–351. [https://doi.org/10.1016/0022-2364\(84\)90181-1](https://doi.org/10.1016/0022-2364(84)90181-1)

[284] T. E. Skinner, N. I. Gershenzon. *Optimal control design of pulse shapes as analytic functions*, Journal of Magnetic Resonance **204**, 2 (2010), pp. 248–255. <https://doi.org/10.1016/j.jmr.2010.03.002>

[285] T. E. Skinner, N. I. Gershenzon, M. Nimbalkar, W. Bermel, B. Luy, S. J. Glaser. *New strategies for designing robust universal rotation pulses: Application to broadband refocusing at low power*, Journal of Magnetic Resonance **216** (2012), pp. 78–87. <https://doi.org/10.1016/j.jmr.2012.01.005>

[286] T. E. Skinner, K. Kobzar, B. Luy, M. R. Bendall, W. Bermel, N. Khaneja, S. J. Glaser. *Optimal control design of constant amplitude phase-modulated pulses: Application to calibration-free broadband excitation*, Journal of Magnetic Resonance **179**, 2 (2006), pp. 241–249. <https://doi.org/10.1016/j.jmr.2005.12.010>

[287] T. E. Skinner, T. O. Reiss, B. Luy, N. Khaneja, S. J. Glaser. *Application of optimal control theory to the design of broadband excitation pulses for high-resolution NMR*, Journal of Magnetic Resonance **163**, 1 (2003), pp. 8–15. [https://doi.org/10.1016/s1090-7807\(03\)00153-8](https://doi.org/10.1016/s1090-7807(03)00153-8)

[288] T. E. Skinner, T. O. Reiss, B. Luy, N. Khaneja, S. J. Glaser. *Reducing the duration of broadband excitation pulses using optimal control with limited RF amplitude*, Journal of Magnetic Resonance **167**, 1 (2004), pp. 68–74. <https://doi.org/10.1016/j.jmr.2003.12.001>

[289] T. E. Skinner, T. O. Reiss, B. Luy, N. Khaneja, S. J. Glaser. *Tailoring the optimal control cost function to a desired output: Application to minimizing phase errors in short broadband excitation pulses*, Journal of Magnetic Resonance **172**, 1 (2005), pp. 17–23. <https://doi.org/10.1016/j.jmr.2004.09.011>

[290] S. Skogestad, I. Postlethwaite. *Multivariable Feedback Control: Analysis and Design*, 2nd edition (Wiley-Interscience, 2005). ISBN: 978-0-47-001168-3

[291] E. D. Sontag. *Mathematical Control Theory: Deterministic Finite Dimensional Systems*, 2nd edition (Springer, 1998). ISBN: 978-0-38-798489-6

[292] P. E. Spindler, S. J. Glaser, T. E. Skinner, T. F. Prisner. *Broadband inversion PELDOR spectroscopy with partially adiabatic shaped pulses*, Angewandte Chemie International Edition **52**, 12 (2013), pp. 3425–3429. <https://doi.org/10.1002/anie.201207777>

[293] P. E. Spindler, Y. Zhang, B. Endeward, N. Gershernzon, T. E. Skinner, S. J. Glaser, T. F. Prisner. *Shaped optimal control pulses for increased excitation bandwidth in EPR*, Journal of Magnetic Resonance **218** (2012), pp. 49–58. <https://doi.org/10.1016/j.jmr.2012.02.013>

[294] A. Spörl, T. Schulte-Herbrüggen, S. J. Glaser, V. Bergholm, M. J. Storcz, J. Ferber, F. K. Wilhelm. *Optimal control of coupled Josephson qubits*, Physical Review A **75**, 1 (2007), p. 012302. <https://doi.org/10.1103/physreva.75.012302>

[295] O. Stern. *Ein weg zur experimentellen prüfung der richtungsquantelung im magnetfeld*, Zeitschrift für Physik **7**, 1 (1921), pp. 249–253. <https://doi.org/10.1007/bf01332793>

[296] Q. Sun, I. Pelczer, G. Riviello, R. B. Wu, H. Rabitz. *Experimental exploration over a quantum control landscape through nuclear magnetic resonance*, Physical Review A **89**, 3 (2014), p. 033413. <https://doi.org/10.1103/physreva.89.033413>

[297] W. H. Swann. *Numerical Methods for Unconstrained Optimization*, W. Murray, ed., Institute of Mathematics and its Applications (Academic Press, 1972). ISBN: 978-0-12-512250-4

[298] D. Tabak, B. C. Kuo. *Optimal Control by Mathematical Programming* (Prentice Hall, 1971). ISBN: 978-0-13-638106-8

[299] J. Tan, M. C. Thurnauer, J. R. Norris. *Electron spin echo envelope modulation due to exchange and dipolar interactions in a spin-correlated radical pair*, Chemical Physics Letters **219**, 3–4 (1994), pp. 283–290. [https://doi.org/10.1016/0009-2614\(94\)87059-4](https://doi.org/10.1016/0009-2614(94)87059-4)

[300] D. J. Tannor. *Introduction to Quantum Mechanics* (University Science Books, 2007). ISBN: 978-1-89-138999-3

[301] D. J. Tannor, V. Kazakov, V. Orlov. *Time-Dependent Quantum Molecular Dynamics. Nato ASI Series (Series B: Physics)*, J. Broeckhove, L. Lathouwers, eds. (Springer, 1992). https://doi.org/10.1007/978-1-4899-2326-4_24

[302] B. Taylor. *Methodus Incrementorum Directa & Inversa* (Gale ECCO, Print Editions, 1715). ISBN: 978-1-17-005307-2

[303] A. J. Temps, C. F. Brewer. *Synthesis of arbitrary frequency domain transmitting pulses applicable to pulsed NMR instruments*, Journal of Magnetic Resonance (1969) **56**, 3 (1984), pp. 355–372. [https://doi.org/10.1016/0022-2364\(84\)90300-7](https://doi.org/10.1016/0022-2364(84)90300-7)

[304] T. Theis, Y. Feng, T. L. Wu, W. S. Warren. *Composite and shaped pulses for efficient and robust pumping of disconnected eigenstates in magnetic resonance*, The Journal of Chemical Physics **140**, 1 (2014), p. 014201. <https://doi.org/10.1063/1.4851337>

[305] M. C. Thurnaur, J. R. Norris. *An electron spin echo phase shift observed in photosynthetic algae: Possible evidence for dynamic radical pair interactions*, Chemical Physics Letters **76**, 3 (1980), pp. 557–561. [https://doi.org/10.1016/0009-2614\(80\)80667-1](https://doi.org/10.1016/0009-2614(80)80667-1)

[306] C. R. Timmel, C. E. Fursman, A. J. Hoff, P. J. Hore. *Spin-correlated radical pairs: microwave pulse effects on lifetimes, electron spin echo envelope modulations, and optimum conditions for detection by electron spin echo spectroscopy*, Chemical Physics **226**, 3 (1998), pp. 271–283. [https://doi.org/10.1016/s0301-0104\(97\)00283-8](https://doi.org/10.1016/s0301-0104(97)00283-8)

[307] V. Torczon. *On the convergence of the multidirectional search algorithm*, SIAM Journal on Optimization **1**, 1 (1991), pp. 123–145. <https://doi.org/10.1137/0801010>

[308] V. J. Torczon. *Multi-directional search: a direct search algorithm for parallel machines*. Ph.D. thesis, Rice University, Houston, TX, USA (1989).

[309] Z. Tošner, R. Andersen, B. Stevensson, M. Edén, N. C. Nielsen, T. Vosegaard. *Computer-intensive simulation of solid-state NMR experiments using SIMPSON*, Journal of Magnetic Resonance **246** (2014), pp. 79–93. <https://doi.org/10.1016/j.jmr.2014.07.002>

[310] Z. Tošner, S. J. Glaser, N. Khaneja, N. C. Nielsen. *Effective Hamiltonians by optimal control: Solid-state NMR double-quantum planar and isotropic dipolar recoupling*, Journal of Chemical Physics **125**, 18 (2006), p. 184502. <https://doi.org/10.1063/1.2366703>

[311] Z. Tošner, T. Vosegaard, C. Kehlet, N. Khaneja, S. J. Glaser, N. C. Nielsen. *Optimal control in NMR spectroscopy: Numerical implementation in SIMPSON*, Journal of Magnetic Resonance **197**, 2 (2009), pp. 120–134. <https://doi.org/10.1016/j.jmr.2008.11.020>

[312] G. E. Uhlenbeck, S. Goudsmit. *Ersetzung der hypothese vom unmechanischen zwang durch eine forderung bezüglich des inneren verhaltens jedes einzelnen elektrons*, Die Naturwissenschaften **13**, 47 (1925), pp. 953–954. <https://doi.org/10.1007/bf01558878>

[313] T. S. Untidt, S. J. Glaser, C. Griesinger, N. C. Nielsen. *Unitary bounds and controllability of quantum evolution in NMR spectroscopy*, Molecular Physics **96**, 12 (1999), pp. 1739–1744. <https://doi.org/10.1080/00268979909483117>

[314] T. S. Untidt, T. Schulte-Herbrüggen, B. Luy, S. J. Glaser, C. Griesinger, O. W. Sørensen, N. C. Nielsen. *Design of NMR pulse experiments with optimum sensitivity: Coherence-order-selective transfer in I_2S and I_3S spin systems*, Molecular Physics **95**, 5 (1998), pp. 787–796. <https://doi.org/10.1080/002689798166413>

[315] S. J. Varner, R. L. Vold, G. L. Hoatson. *An efficient method for calculating powder patterns*, Journal of Magnetic Resonance, Series A **123**, 1 (1996), pp. 72–80. <https://doi.org/10.1006/jmra.1996.0215>

[316] M. Veshtort, R. G. Griffin. *SPINEVOLUTION: A powerful tool for the simulation of solid and liquid state NMR experiments*, Journal of Magnetic Resonance **178**, 2 (2006), pp. 248–282. <https://doi.org/10.1016/j.jmr.2005.07.018>

[317] M. S. Vinding, D. Brenner, D. H. Y. Tse, S. Vellmer, T. Vosegaard, D. Suter, T. Stöcker, I. I. Maximov. *Application of the limited-memory quasi-Newton algorithm for multi-dimensional, large flip-angle RF pulses at 7T*, Magnetic Resonance Materials in Physics, Biology and Medicine **30**, 1 (2017), pp. 29–39. <https://doi.org/10.1007/s10334-016-0580-1>

[318] M. S. Vinding, B. Guérin, T. Vosegaard, N. C. Nielsen. *Local SAR, global SAR, and power-constrained large-flip-angle pulses with optimal control and virtual observation points*, Magnetic Resonance in Medicine **77**, 1 (2017), pp. 374–384. <https://doi.org/10.1002/mrm.26086>

[319] M. S. Vinding, I. I. Maximov, Z. Tošner, N. C. Nielsen. *Fast numerical design of spatial-selective rf pulses in MRI using Krotov and quasi-Newton*

based optimal control methods, Journal of Chemical Physics **137**, 5 (2012), p. 054203. <https://doi.org/10.1063/1.4739755>

[320] E. Vinogradov, P. K. Madhu, S. Vega. *New Techniques in Solid-State NMR*, J. Klinowski, ed. (Springer, 2005), vol. 246 of *Topics in Current Chemistry*. ISBN: 978-3-54-022168-5. <https://doi.org/10.1007/b98648>

[321] H. A. Van der Vorst. *An iterative solution method for solving $f(A)x = b$, using Krylov subspace information obtained for the symmetric positive definite matrix A*, Journal of Computational and Applied Mathematics **18**, 2 (1987), pp. 249–263. [https://doi.org/10.1016/0377-0427\(87\)90020-3](https://doi.org/10.1016/0377-0427(87)90020-3)

[322] T. Vosegaard, C. Kehlet, N. Khaneja, S. J. Glaser, N. C. Nielsen. *Improved excitation schemes for multiple-quantum magic-angle spinning for quadrupolar nuclei designed using optimal control theory*, Journal of the American Chemical Society **127**, 40 (2005), pp. 13768–13769. <https://doi.org/10.1021/ja054035g>

[323] D. Wang, G. R. Hanson. *A new method for simulating randomly oriented powder spectra in magnetic resonance: The Sydney Opera House (SOPHE) method*, Journal of Magnetic Resonance, Series A **117**, 1 (1995), pp. 1–8. <https://doi.org/10.1006/jmra.1995.9978>

[324] X. Wang, A. Bayat, S. G. Schirmer, S. Bose. *Robust entanglement in anti-ferromagnetic Heisenberg chains by single-spin optimal control*, Physical Review A **81**, 3 (2010), p. 032312. <https://doi.org/10.1103/physreva.81.032312>

[325] R. C. Ward. *Numerical computation of the matrix exponential With accuracy estimate*, SIAM Journal on Numerical Analysis **14**, 4 (1977), pp. 600–610. <https://doi.org/10.1137/0714039>

[326] W. S. Warren. *Effects of arbitrary laser or NMR pulse shapes on population inversion and coherence*, The Journal of Chemical Physics **81**, 12 (1984), pp. 5437–5448. <https://doi.org/10.1063/1.447644>

[327] W. S. Warren, H. Rabitz, M. Dahleh. *Coherent control of quantum dynamics: The dream is alive*, Science **259**, 5101 (1993), pp. 1581–1589. <https://doi.org/10.1126/science.259.5101.1581>

[328] D. Wei, U. Akbey, B. Paaske, H. Oschkinat, B. Reif, M. Bjerring, N. C. Nielsen. *Optimal 2H rf pulses and 2H – ^{13}C cross-polarization methods for solid-state 2H MAS NMR of perdeuterated proteins*, Journal of Physical

Chemistry Letters **2**, 11 (2011), pp. 1289–1294. <https://doi.org/10.1021/jz200511b>

[329] D. Wei, Y. Chang, S. J. Glaser, X. Yang. *Cooperative pulses for pseudo-pure state preparation*, Applied Physics Letters **104**, 24 (2014), p. 4884295. <https://doi.org/10.1063/1.4884295>

[330] H. Weyl. *Quantenmechanik und gruppentheorie*, Zeitschrift für Physik **46**, 1–2 (1927), pp. 1–46. <https://doi.org/10.1007/bf02055756>

[331] S. Wimperis. *Broadband, narrowband, and passband composite pulses for use in advanced NMR experiments*, Journal of Magnetic Resonance, Series A **109**, 2 (1994), pp. 221–231. <https://doi.org/10.1006/jmra.1994.1159>

[332] H. M. Wiseman, G. J. Milburn. *Quantum Measurement and Control* (Cambridge University Press, 2009). ISBN: 978-0-52-180442-4

[333] P. Wolfe. *Convergence conditions for ascent methods*, SIAM Review **11**, 2 (1969), pp. 226–235. <https://doi.org/10.1137/1011036>

[334] P. Wolfe. *Convergence conditions for ascent methods. II: Some corrections*, SIAM Review **13**, 2 (1971), pp. 185–188. <https://doi.org/10.1137/1013035>

[335] R.-B. Wu, H. Rabitz. *Control landscapes for open system quantum operations*, Journal of Physics A: Mathematical and Theoretical **45**, 48 (2012), p. 485303. <https://doi.org/10.1088/1751-8113/45/48/485303>

[336] R. Zeier, T. Schulte-Herbrüggen. *Symmetry principles in quantum systems theory*, Journal of Mathematical Physics **52**, 11 (2011), p. 113510. <https://doi.org/10.1063/1.3657939>

[337] G. Zhang, F. Schilling, S. J. Glaser, C. Hilty. *Chemical shift correlations from hyperpolarized NMR using a single SHOT*, Analytical Chemistry **85**, 5 (2013), pp. 2875–2881. <https://doi.org/10.1021/ac303313s>

[338] W. Zhu, H. Rabitz. *A rapid monotonically convergent iteration algorithm for quantum optimal control over the expectation value of a positive definite operator*, The Journal of Chemical Physics **109**, 2 (1998), pp. 385–391. <https://doi.org/10.1063/1.476575>