# Protocols and applications for imperfect quantum devices


Matthew L. Goh

Merton College

University of Oxford

A thesis submitted for the degree of

*Doctor of Philosophy*

Michaelmas 2024

# Acknowledgements


## Personal

Adequately thanking the many people whose support has enabled me to start and finish this doctorate is perhaps a more daunting task than the thesis itself. Nonetheless, I must make an attempt.

A wise individual once said 'always two, there are, no more, no less — a master and an apprentice'. I have been extremely fortunate to be an exception to this rule, having spent the last three years apprenticed to not one, two, or even three masters, but four. First of all, to my primary supervisor Simon Benjamin, I am enormously grateful for your guidence, patience, wisdom, and support. I thank you for taking a chance on me in my pandemic-and-quarter-life-crisis-induced transfer to your group, for always finding time and input when needed, and for seemingly always being able to effortlessly procure the optimal solution to any predicament. To my co-supervisor Bálint Koczor, you have been an inspiring teacher, a seemingly endless fountain of brilliant ideas, and a reliable guide through the vagaries of academic publishing. To my Los Alamos mentors Marco Cerezo and Lukasz Cincio, I cannot thank you enough for the opportunities you opened the door to, from the 2022 QCSS to our ongoing collaborations. I am grateful for the time you have spent teaching me the finer points of everything from quantum machine learning to academic mentorship, and for guiding me to stand on my own two feet as a researcher.

I'm privileged to have spent my time among extremely talented peers. To all members of the Quantum Technology Theory Group at Oxford during my time here, to the fellow students and postdocs at LANL, and to my exceptional collaborators on the works included in this thesis — thank you for fostering an intellectually stimulating and thought-provoking environment.

Many friends in Oxford have made my time here remarkable. In no particular order beyond alphabetical, I thank Amy Booth, Beth Hamilton, Christian Farrier, Elizabeth Tuck, Emily Boucher, Joe Arroway, Marcus Dahl, Minnie Robson, and Sam Hauptman for their friendship, support, and camaraderie. And to my friends further afield, supporting me from the sunny shores of Australia, thank you for projecting a sense of home all the way across the globe. The denizens of the somewhat-obliquely-named group chats The Wonderland, The Tribunal, and

Icewind Dale have been particularly irreplaceable sources of advice, distraction, and amusingly unwise behaviour.

This thesis is not a short document — proofreading it was a substantial task, and I am grateful to all of those whose input has helped to refine it. In no particular order, my thanks go out to Simon Benjamin, Bálint Koczor, Ruvi Lecamwasam, Janet Zhong, and Philippa Sims for their comments and input.

It took the devoted work of many people over the last 27 years to shape me into the sort of person capable of completing a doctorate in quantum physics. To all of my past mentors and teachers who guided me along the way, I hope that this thesis reflects at least some of the intellectual maturity you have imparted to me. And to my family, who valued my education above all, thank you for unknowingly — yet deliberately — forging the path that led me here. I offer the deepest thanks to the closest members of my family, who were there for me throughout the ups and downs of my childhood and early adulthood. To my grandmother Queenie: your love and care are still felt today and will last a lifetime; you are dearly missed by all you loved. To my brother James: thank you for being my partner in crime through childhood, and for all you have done to support us through the hardest of times. To my mother Lynn: for creating a loving and welcoming home and a childhood full of treasured memories, for the hard work and sacrifices you made to create opportunity for me, and for carrying us through great adversity even though you could not walk, I cannot thank you enough. I know that you would be proud of me no matter what, but this thesis reflects the pride I have in you, and the determination and cleverness I inherit from you.

Last, but quite possibly most of all, I thank my wonderful fiancée Philippa Sims. Philippa, I am unable to capture in words exactly how much your unwavering love, support, and kindness have done for me. The existence of this thesis is as much an achievement of your generosity as it is of my own work, and I am truly beyond grateful that returning to Oxford brought us together. The rest of this thesis outlines the second most important thing I did in Oxford; these sentences credit the most important thing, which was meeting you.

## Institutional

Those who have listened to my complaints about Oxonian bureaucracy will be well aware of my claim that ‘the University of Oxford is a lie we all tell ourselves; it’s just a loose collection of colleges, departments, and fiefdoms in a trenchcoat, none of whom will talk to each other’. This statement is, of course, eminently true, but it has been to my benefit. The forced interation with many different institutions under the trenchcoat has increased the cross-section over which I might

find support, and hidden among the Kafkaesque bureaucracy and curmudgeons are some truly exceptional people and places.

Completing a doctorate costs money — considerably more than most cities in the UK, and most of which found its way into the pockets of unscrupulous landlords. I am of course exceedingly grateful to the Rhodes Trust for funding my DPhil via a Rhodes Scholarship, and to Merton College and the Department of Materials for additional financial support. I also extend my thanks to the US taxpayer and the Los Alamos National Laboratory for funding my collaborations with their Theoretical Division, which together compose about half of this thesis.

The impact of these institutions goes well beyond financial. I'm grateful to the Rhodes Trust for the transformative opportunity to be part of such an exceptional community of young people, and I extend particular thanks to Mary and Maureen for their support throughout the more tumultuous parts of my studies. It has been a privilege to be part of the Merton College community, and I'm particularly thankful for the teaching opportunities Merton provided me via a college lectureship.

My research, like most theoretical work in quantum technology, consumed a number of CPU and GPU hours which I dare not count — I thank the Oxford Advanced Research Computing facility for their generous allocation of computing resources.

# Abstract

Quantum computing is a rapidly advancing field that, if fully realized, promises to be revolutionary for applications including drug discovery, optimization, and cryptography. However, the challenges remaining to achieve this dream are considerable. Today's quantum computers (and those the foreseeable near future) are plagued by limited system sizes, decoherence, faulty operations, and a variety of other issues that have thus far inhibited quantum advantage on genuinely useful problems. While others pursue the ultimate goal of fully-fault-tolerant quantum computing, this thesis is about learning to live with these limitations, and to extract useful information from imperfect, noisy devices with no or limited quantum error correction. In this thesis, I introduce new algorithms that are resilient to these effects, study protocols that can best compensate for them, and probe the boundaries between quantum and classical systems.

This thesis spans three rather distinct topics in quantum technology: the use of imperfect quantum computers to study quantum systems; classically-efficient simulation of quantum devices; and quantum sensing in the presence of noise. I introduce new protocols and algorithms for each of these areas, and support them with both mathematical study and computer simulations. Broadly speaking, the contributions of this thesis are as follows. I introduce a new algorithm for resource-efficient estimation of excitation energies in quantum systems, studying its near-term applicability and scalability. I develop a new method for computing thermal properties of quantum systems on quantum computers, which appears particularly applicable to the early-fault-tolerant era, and expands upon previous approaches in both the resources required and the systems it can be applied to. I introduce a new classical simulation framework for variational quantum algorithms, expanding the domain of classically simulable problems in quantum computing, and demonstrate its application to archetypal problems in variational quantum computing and quantum machine learning. Finally, I compare protocols that compensate for the presence of noise in quantum sensing, showing that learning-based techniques are particularly promising.

# Contents

**List of Figures** **x**

**1 Introduction** **1**
1.1 Quantum computing: the promises . . . 1
1.2 Quantum computing: the reality . . . 5
1.3 Quantum sensing . . . 7

**2 Background** **10**
2.1 Simulating quantum devices . . . 11
2.2 Variational quantum algorithms . . . 17
2.2.1 Methods . . . 17
2.2.2 Applications . . . 20
2.2.3 Barren plateaus, trainability, and simulability . . . 23
2.2.4 Trainability beyond barren plateaus . . . 28
2.3 Quantum error mitigation . . . 30
2.4 Calculating spectral properties of Hamiltonians . . . 34
2.4.1 Quantum phase estimation . . . 34
2.4.2 Variants of phase estimation . . . 36
2.4.3 Thermal properties . . . 37
2.4.4 Excited states . . . 39
2.4.5 Eigenvalue gaps . . . 40
2.5 Classical shadows . . . 41
2.5.1 Outline . . . 41
2.5.2 Convergence and unitary ensembles . . . 43

**3 Algorithmic shadow spectroscopy** **45**
3.1 Methods . . . 46
3.1.1 Classical shadows . . . 48
3.1.2 Signal postprocessing . . . 49
3.1.3 Summary and resource requirements . . . 54
3.1.4 Robustness to gate noise . . . 55
3.2 Scalability . . . 57

3.2.1 Analytic model . . . 58
3.2.2 Verifying scalability with tensor network simulations . . . 60
3.3 Application to near-term problems . . . 65
3.3.1 Variational dynamics . . . 66
3.3.2 Demonstration of variational shadow spectroscopy . . . 67
3.4 Conclusions . . . 70

**4 Direct estimation of the density of states for fermionic systems 74**
4.1 Preliminaries . . . 76
4.1.1 Physical systems of constant particle number . . . 76
4.1.2 Physical systems of variable particle number . . . 77
4.2 Results . . . 79
4.2.1 Estimating the FDOS using DQC1 circuits . . . 80
4.2.1.1 One Clean Qubit (DQC1) computation . . . 80
4.2.1.2 Modified DQC1 computation for general subspaces 81
4.2.2 Estimating the FDOS via random-state initialization . . . 83
4.2.3 Reconstructing the DOS . . . 87
4.2.3.1 Windowed time evolution . . . 87
4.2.4 Directly estimating thermodynamic properties . . . 90
4.3 Random state sampling requirements . . . 91
4.4 Applicability in the early-fault-tolerant era . . . 96
4.4.1 Resilience to algorithmic error . . . 97
4.4.2 Resilience to gate noise . . . 100
4.5 Applicability in the NISQ era . . . 103
4.5.1 Variational dynamics via classical shadows and covariance root-finding . . . 103
4.5.2 Demonstrating noisy variational DOS estimation . . . 107
4.6 Conclusions . . . 111

**5 Lie-algebraic classical simulations for variational quantum computing 114**
5.1 Framework . . . 115
5.1.1 Foundations . . . 117
5.1.2 Lie theory . . . 119
5.1.3 $\mathfrak{g}$-sim principles . . . 122
5.1.3.1 Simulation with observables supported by $\mathfrak{g}$ . . . 123
5.1.3.2 Simulation with products of observables supported by $\mathfrak{g}$ . . . 124
5.1.3.3 Evaluation of gradients . . . 127
5.1.3.4 Simulation of non-unitary quantum channels . . . 127

5.1.4 Generalization to operators not supported by $\mathfrak{g}$ . . . . . 130
5.2 Guidelines for efficient implementation . . . . . 132
5.2.1 Efficient evolution . . . . . 133
5.2.1.1 Observables . . . . . 133
5.2.1.2 Gradients . . . . . 133
5.2.2 Sparsity of adjoint representation . . . . . 135
5.2.3 Efficient calculation and eigendecomposition of Pauli-basis adjoint representations . . . . . 137
5.3 Scalability and demonstration . . . . . 138
5.3.1 Scalability of $\mathfrak{g}$-sim . . . . . 138
5.3.2 A polynomial-sized Lie algebra . . . . . 140
5.3.3 Example simulation task . . . . . 141
5.3.4 Resource benchmark . . . . . 144
5.4 Applications . . . . . 146
5.4.1 Characterizing VQAs . . . . . 147
5.4.1.1 Overparametrization in VQE . . . . . 148
5.4.1.2 Simulation results . . . . . 148
5.4.2 Pre-training quantum circuits . . . . . 151
5.4.2.1 Pre-training strategy with $\mathfrak{g}$-sim . . . . . 152
5.4.2.2 Pre-training VQE for the LTFIM . . . . . 153
5.4.2.3 Pre-training QAOA . . . . . 156
5.4.3 Circuit synthesis . . . . . 158
5.4.3.1 Variational compilation of unitaries in $\mathfrak{g}$-sim . . . . 159
5.4.3.2 Compilation cost function gradients . . . . . 160
5.4.3.3 Compiling random unitaries . . . . . 161
5.4.3.4 Faithfulness of compilation . . . . . 163
5.4.3.5 Application to dynamical simulation . . . . . 166
5.4.4 Supervised quantum machine learning . . . . . 170
5.4.4.1 General framework . . . . . 170
5.4.4.2 Training a binary quantum-phase classifier . . . . . 171
5.5 Conclusions . . . . . 174
5.5.1 Scope for future work . . . . . 176

**6 Why learning trumps mitigation in noisy quantum sensing 179**
6.1 Framework . . . . . 182
6.1.1 Noisy-state quantum sensing . . . . . 182
6.1.2 An illustrative toy model . . . . . 186
6.1.3 Zero-noise extrapolation . . . . . 187
6.1.3.1 The method . . . . . 187

6.1.3.2 The shot cost . . . 188
6.1.4 Hyperparameter optimization . . . 189
6.1.5 Learning-based inference methods . . . 191
6.1.5.1 The method . . . 191
6.1.5.2 The shot cost . . . 193
6.1.6 Parameter measurement via phase measurement . . . 195
6.2 Analytic estimates of error bounds . . . 197
6.2.1 Noise-aware noisy quantum sensing . . . 200
6.2.2 Naive noisy quantum sensing . . . 201
6.2.3 Noisy sensing mitigated by zero-noise extrapolation . . . 202
6.2.4 Inference-based noisy sensing . . . 203
6.2.5 Inference-based noisy sensing mitigated by zero-noise extrapolation . . . 208
6.2.6 Comparison of error bounds . . . 209
6.3 Numerical experiments . . . 213
6.3.1 Noise models . . . 214
6.3.2 Numerical comparison of errors across varying regimes . . . 215
6.3.3 Requirements for pre-characterized inference-based sensing . 218
6.4 Conclusions . . . 220

**7 Conclusions** **223**

**Appendices**

**A Lie-algebraic quantum dynamics** **228**
A.1 Proof of Lemma 1 . . . 228
A.2 Proof of Lemma 2 . . . 229
A.3 Proof of Theorem 3 . . . 230

**B Evaluating error bounds for quantum sensing protocols** **231**
B.1 Noise-aware noisy quantum sensing . . . 231
B.2 Naive noisy sensing . . . 232
B.3 Noisy sensing mitigated by zero-noise extrapolation . . . 232
B.4 Inference-based noisy sensing . . . 233
B.5 Inference-based noisy sensing mitigated by zero-noise extrapolation 234

**References** **235**

# List of Figures


2.1 Computer memory requirements for various classical simulation methods. . . . . 14
2.2 Schematic depiction of a variational quantum algorithm. . . . . 19
2.3 Bias-variance tradeoff in quantum error mitigation. . . . . 33
2.4 Standard circuit for quantum phase estimation (QPE). . . . . 35

3.1 Walkthrough of algorithmic shadow spectroscopy. . . . . 56
3.2 Diagrammatic representation of MPS methods. . . . . 62
3.3 Logarithmic intensity shadow spectra of a disordered Heisenberg spin chain. . . . . 64
3.4 Simulated and experimental shadow spectroscopy on a 6-qubit NISQ system. . . . . 69

4.1 The Fermi-Hubbard model and its fixed-particle-number DOS. . . . 79
4.2 Quantum circuits for evaluating the FDOS. . . . . 81
4.3 Effects of finite simulation length. . . . . 89
4.4 Comparison of random initial-state sampling methods. . . . . 93
4.5 Convergence of sampling methods for fermionic problems. . . . . 95
4.6 Effect of initial-state-preparation circuit depth on DOS error. . . . . 96
4.7 Effect of algorithmic error on DOS calculations. . . . . 98
4.8 Effect of hardware noise on DOS estimation. . . . . 102
4.9 Circuits for variational dynamics compilation. . . . . 104
4.10 Recompiling Trotter steps with CoVaR. . . . . 106
4.11 Decomposition of controlled multi-qubit Pauli rotations. . . . . 108
4.12 Calculating density of states with noisy variational quantum dynamics. 110

5.1 Theoretical framework and applications of $\mathfrak{g}$-sim. . . . . 117
5.2 Tensor contractions for $\mathfrak{g}$-sim evolution. . . . . 126
5.3 Sparsity of gates generated by Pauli strings versus general gates. . . 136
5.4 Noiseless and noisy dynamics of a 200-qubit magic state using $\mathfrak{g}$-sim. 143
5.5 Performance benchmarks of $\mathfrak{g}$-sim. . . . . 145
5.6 Overparametrization in large system sizes. . . . . 149
5.7 Pre-training VQE for the LTFIM using $\mathfrak{g}$-sim. . . . . 155

5.8 Pre-training QAOA using $\mathfrak{g}$-sim. . . . . 157
5.9 Scaling properties of circuit compilation with $\mathfrak{g}$-sim. . . . . 162
5.10 Compiling short-duration dynamics for global Hamiltonians. . . . . 163
5.11 Successfully compiling circuits in an algebra with non-trivial group center elements. . . . . 165
5.12 Compiling Hamiltonian dynamics with $\mathfrak{g}$-sim. . . . . 168
5.13 Supervised QML with a classically-efficient circuit. . . . . 172
5.14 Comparison of exact label inferences to their 2-local approximation. 174

6.1 Summary of this chapter's results. . . . . 181
6.2 Using zero-noise extrapolation for noisy quantum sensing. . . . . 190
6.3 Choosing hyperparameters for ZNE. . . . . 192
6.4 The effect of shot budget on response function inference. . . . . 194
6.5 Flowchart of the protocols studied in this chapter. . . . . 198
6.6 Comparison of analytically estimated error bounds. . . . . 211
6.7 Individual error contributions in parameter estimation. . . . . 212
6.8 GHZ state preparation channel for numerical experiments. . . . . 214
6.9 Numerical experiments for magnetometry protocols. . . . . 216
6.10 Error convergence of pre-characterized, inference-based sensing. . . 219

*In the beginning the Universe was created. This has made a lot of people very angry and been widely regarded as a bad move.*

— Douglas Adams, *The Restaurant at the End of the Universe* [1]

# 1 Introduction

## Contents


**1.1 Quantum computing: the promises . . . . . 1**
**1.2 Quantum computing: the reality . . . . . 5**
**1.3 Quantum sensing . . . . . 7**


## 1.1 Quantum computing: the promises

Digital computing rests on a solid theoretical foundation. Drawing upon seminal works such as Turing's universal model of classical computation [2], the principles of Boolean algebra [3], and Shannon's application of these to logic circuits [4], scientists and mathematicians of the early 20th century reasonably believed they knew *what* a computer was in the abstract sense. Building upon these foundations, the theory of computational complexity emerged [5, 6], providing a rigorous framework for understanding the inherent difficulty of computational problems. In the decades that followed, computing took physical form: from early electromechanical systems, to vacuum tube computers, to the silicon-based integrated circuits through which I now write this thesis, various physical systems manifested the abstract notion of 'computation'.

Another scientific revolution was underway in the early 20th century: while mathematicians laid the foundations of computing, physicists were uncovering the strange and counterintuitive world of quantum mechanics. Computation is a physical process, and the systems that implement it are subject to the laws of physics. If a computer can leverage the core quantum principles of superposition, entanglement and interference, what effect might this have on its computational capabilities? This question led Benioff to extend the notion of Turing machines to a quantum-mechanical generalization [7]. The idea of quantum computing truly took off with Feynman's famous conjecture that a quantum-mechanical computer could efficiently simulate quantum systems [8], later proven by Lloyd [9]. Since classically representing an entangled quantum system requires an amount of memory exponential in the number of constituents, it seemed hopeful that a computer based on quantum mechanics could prove useful. Our universe is quantum-mechanical, we have much to gain scientifically by modelling it, and surely quantum problems require quantum solutions?

These ideas quickly developed into a theory of universal *quantum* computation [10], and by 1992 Deutsch and Jozsa had identified a class of problems which could be solved exponentially faster on a quantum computer compared to a classical computer [11]. While there was some doubt as to whether such problems were practically relevant, this was quickly extinguished by Shor's discovery of a polynomial-time quantum algorithm for factoring integers [12]. The Rivest-Shamir-Adleman (RSA) cryptosystem [13], which relies on the presumed exponential classical difficulty of integer factoring [14], was now vulnerable to quantum attack. A range of other quantum algorithms and subroutines with asymptotic speedups relative to their classical counterparts soon followed, including Grover's algorithm for unstructured search problems [15], the quantum Fourier transform (QFT) [16, 17], quantum phase estimation (QPE) [16, 18], and quantum amplitude estimation [19]. As I discuss in Chapter 2, quantum phase estimation is particularly useful for computing eigenvalues and eigenstates of Hamiltonians [20, 21]. Not only is this useful for modelling physical quantum systems, but a number of optimization

problems can be encoded in the ground state of Hamiltonians [22–24], rendering it a powerful tool for many other problems. For modelling quantum systems, a vast range of quantum algorithms for simulating dynamics of quantum systems have emerged, including Trotterization [25], linear combination of unitaries [26], qubitization [27], quantum signal processing [28], and variational methods [29]. Even non-unitary open quantum systems now have a variety of quantum algorithms available to simulate their dynamics [30]. This has fuelled considerable excitement for applications in chemistry, materials science, and condensed matter physics. Although the discovery of new elementary subroutines for quantum algorithms has slowed, there have been a few notable exceptions since the turn of the millennium, such as the Harrow-Hassidim-Lloyd algorithm (HHL) for solving linear systems [31], the quantum singular value transform [32], and the generalizations that follow from it [33]. Overall, we have a rich toolbox of quantum subroutines that can be combined in various ways to achieve quantum speedups; today there are many known quantum algorithms for a broad range of applications [34, 35].

This realization that computation is a physical — and therefore quantum-mechanical — process stands to enormously broaden the bounds of what is computable. Unfortunately, this tantalizing realization must be tempered by the gruelling realities of the physical world in which we hope to implement it. While the advantages of the algorithms discussed above would be undeniable in an *isolated* quantum system, the real world is noisy, and one cannot build a truly isolated quantum device. Quantum states are extremely fragile, and even the slightest interaction with an incoherent environment can destroy the entanglement required for quantum algorithmic speedups. This phenomenon, known as *decoherence*, limits the time over which quantum information can be stored and manipulated, and therefore the depth of quantum circuits that can be executed. Just as the promise of quantum computing became appealing, it was shown that this could limit quantum algorithmic advantages [36], sparking caution and debate as to whether quantum computing could ever be practical [37, 38]. Although careful engineering can reduce these errors, they cannot be eliminated entirely.

Early classical computers were error-prone, and classical error-correcting codes were developed to mitigate this [39], whereby information is stored redundantly so that errors can be detected and corrected. These classical error correction schemes cannot be directly applied to quantum memory, since it is more complicated: not only can quantum states suffer phase-flip as well as bit-flip errors, but quantum superpositions cannot be measured without collapsing them, preventing one from directly comparing the states of redundant physical qubits. Nonetheless, early quantum error correction (QEC) schemes were developed [40–43], which similarly encode a logical qubit in multiple fault-prone physical qubits. In these error-correcting schemes, one measures non-destructive error syndromes on ancillary qubits in such a way that errors can be detected and corrected without collapsing the state via full measurement. For example, the simplest QEC code is the three-qubit bit-flip code, where a qubit's basis state is encoded in the subspace of the three-qubit Hilbert space spanned by $|000\rangle$ and $|111\rangle$; a single bit-flip error can be identified and corrected by parity-check measurements on the three physical qubits, which do not destroy the quantum information. Today, the theory of QEC codes is a mature field with a broad range of sophisticated codes available, with surface codes in particular being a leading candidate for scalable fault-tolerant quantum computing [44, 45].

The battle against the deleterious effects of noise in quantum computers seems, at least in theory, winnable. In a practical setting, one would expect the correcting operations themselves to be noisy, and perhaps fear that this would compromise QEC schemes — but fault-tolerance can still be achieved in this case [46]. More importantly however, the *threshold theorem* shows that, provided that the error rate of the physical qubits is below a certain threshold, arbitrarily long quantum computations can be executed with arbitrarily low error rates by concatenating QEC codes [47–51]. Therefore, the dream of fault-tolerant quantum computing is not a Sisyphean task, but one with a clear victory condition.

## 1.2 Quantum computing: the reality

Fault-tolerant quantum computing may not be a Sisyphean task, but the boulder has yet a long way to climb. Today, several different types of quantum hardware vie for dominance. Particularly promising recent results have been seen in superconducting qubits [52, 53], trapped ions [54], and neutral atoms [55], and other notable platforms include photonic systems [56] and silicon architectures [57]. Although impressive progress has been made in the past few years in implementing error correction and logical qubits in a variety of different quantum hardware platforms [54, 55, 58–62], it is only in the past few months that the first instances of sub-threshold error correction have been reported for the surface code [53]. Implementing fault-tolerant quantum computing is an extremely challenging task, requiring high-fidelity entangling gates, topologies compatible with the code in question, mid-circuit measurements, and classical infrastructure such as decoders [63]. At least for the time being, we must learn to live with the noise, and question what can be achieved in spite of it.

Perhaps there is some reason to be hopeful. Today's best quantum computers have tens to hundreds of qubits, two-qubit gate fidelities on the order of $10^{-3}$, and greatly varying qubit topologies (e.g nearest-neighbour architectures in superconducting devices, or all-to-all connectivity in some ion traps). This state of affairs is most commonly known as the Noisy Intemediate-Scale Quantum (NISQ) era, coined by Preskill in 2018 [64]; the nature of algorithms in this era is characterized by small to modest system sizes, shallow circuits, and careful coping strategies for the significant limitations of these devices (see Section 2.3). While noise destroys the coherence that is necessary for quantum advantage, these devices can still produce and manipulate non-negligibly entangled quantum states, albeit with circuit depths falling far short of the requirements for the algorithms discussed in Section 1.1. Claims of experimental quantum advantage on contrived sampling problems [65], while initially defeated [66, 67], have recently been strengthened on newer devices [53, 68, 69]. It seems that the quantum devices we have in hand are capable of doing *something* that classical computers cannot. Clearly, we are

able to explore something of the exponential richness of multipartite Hilbert spaces, so can we do anything *useful* with this?

Caution would be warranted in answering this question. Bold claims of quantum advantage on useful problems (Trotterized simulation of an Ising model) have been made recently [52], but amusingly were refuted within days by classical simulations [70–72], with numerous other approaches soon following [73–77] including one executed on a Commodore 64 [78]. Given that even tens of Trotter steps for a modest-sized, hardware-tailored Hamiltonian proved too deep to achieve a genuine quantum advantage on today's devices [52, 70–78], an orthogonal approach relying upon shallow circuits is likely needed if we are to ever extract value from NISQ devices. Most proposals for doing so are based upon variational quantum algorithms [79], which are hybrid quantum-classical algorithms that repeatedly execute a parametrized quantum circuit on a quantum processing unit (QPU), and use numerical optimzation techniques to tune its parameters on a classical computer. Much of this thesis is concerned with these: I review the applications, pitfalls and limitations in the literature in Chapter 2, and apply them in various parts of Chapters 3, 4 and 5.

While error correction is beyond the capabilities of NISQ devices, not all is lost. In addition to the ongoing march of hardware improvements and error suppression via control techniques, one can also consider the possibility of error *mitigation*. Such techniques aim to compensate for the effects of noise in quantum computations without actually directly reducing the logical error rate, and often reduce the effect of errors rather than eliminate them entirely. These are an invaluable tool in the NISQ era, and I review the relevant literature more closely in Chapter 2.

With experimental demonstrations of QEC continuing to breach new milestones [53–55, 58–62], including early demonstrations of sub-threshold error correction [53], the horizon of the NISQ era is already in view. Indeed, as I discuss in Chapters 2 and 5, the looming spectre of classical simulability threatens variational quantum computing as a whole, throwing the pursuit of quantum advantage on NISQ systems into question. As hardware improves, we should soon expect to

enter the era of *early fault-tolerant* quantum computing [80, 81], where systems with small numbers of sub-threshold error-corrected logical qubits can conduct computations with logical error rates lower than the underlying physical error rates (e.g. error rates on the order of $10^{-6}$ may be achievable). In the early fault-tolerant era, unitaries will likely be constructed from small universal gate sets such as the Clifford+$T$ set [82–84], with errors in computation originating from approximation of certain gates rather than the underlying errors in the physical qubits. For example, the $T$ gate (which, in combination with Clifford gates, is universal for quantum computation) will likely be implemented using $T$-gate gadgets [85], bottlenecked by magic state distillation [86].

These eras of quantum computing are the key context for this thesis. Chapters 3, 5 and 6 are predominantly concerned with implementation in the NISQ era, while Chapter 4 focuses chiefly on early-fault-tolerant implementation (with a brief adaptation to the NISQ context). Nonetheless, the barriers between eras of quantum computing are not always so clear-cut, and I anticipate that the techniques of Chapters 3, 4 and 5 in particular will remain useful even into the era of fully-fault-tolerant quantum computing, should we be lucky enough to ever see it arrive.

## 1.3 Quantum sensing

Most of this thesis is concerned with the use of quantum systems for processing information — that is, quantum computing — but there is also great promise in using them for *acquiring* information. Broadly speaking, the field of *quantum sensing* is concerned with using quantum systems, properties or phenomena to perform measurements of physical quantities. Indeed, this is an extremely broad definition, and one could reasonably argue about what is, and is not, a quantum sensor. In some sub-communities, merely using quantized energy levels of a system to measure a physical quantity (e.g. atomic clocks [87]), or using wavelike coherent quantum properties (e.g. thermal-source atom interferometry [88, 89]) would be considered to be quantum sensing. However, this thesis is grounded in the conventions of quantum computing, where entanglement is king; for the purposes of this thesis, I

consider a quantum sensor to be a device that exploits entanglement to measure a physical quantity with greater precision than is possible classically. 'Quantum sensors' satisfying these less restrictive definitions are beyond the scope of this thesis, but nonetheless underpin the current frontiers of sensing capabilities. A comprehensive review thereof can be found in Ref. [90] (see discussion of type-I and type-II quantum sensors).

The tremendous progress in state-of-the-art technologies for quantum control has not only enabled the development of quantum computers, but also opened up the exciting possibility of using entangled quantum systems for high-precision measurements [90–97]. To see why entanglement is a veritable resource for sensing, note that the precision of a measurement of some parameter $\theta$ is limited by the quantum Cramér-Rao bound as

$$\Delta^2\theta \geq \frac{1}{NF_Q(\rho, H)}, \tag{1.1}$$

where $N$ independent repetitions (shots) are conducted, and $F_Q(\rho, H)$ is the quantum Fisher information (QFI) of the probe state $\rho$ with respect to the Hamiltonian $H$ that encodes the parameter $\theta$. The QFI is a measure of the probe state's sensitivity to $\theta$, can be used to quantify its metrological usefulness, and is formally defined as

$$F_Q(\rho, H) = 2\sum_{j,k} \frac{(\lambda_j - \lambda_k)^2}{(\lambda_j + \lambda_k)} |\langle j| H |k\rangle|^2, \tag{1.2}$$

where $\lambda_j$, $|j\rangle$ are corresponding eigenvalues and eigenvectors of the probe state $\rho$, and the summation only includes terms such that $\lambda_j + \lambda_k > 0$. While the sensitivity is inversely proportional to the number of repetitions, the QFI is influenced by entanglement: if the probe state consists of $n$ unentangled qubits, it is well known that the attainable precision is bounded as

$$\Delta^2\theta \geq \frac{1}{Nn}. \tag{1.3}$$

Equation 1.3 is known as the *standard quantum limit* (SQL), and is the best precision achievable by a classical (unentangled) probe state. When the probe state is allowed to be entangled, the bound is instead

$$\Delta^2\theta \geq \frac{1}{Nn^2}, \tag{1.4}$$

known as the *Heisenberg limit* (HL), which allows asymptotically (in $n$) better precision than the SQL. Although better bounds are attainable when the encoding Hamiltonian $H$ contains interaction terms [93], this is beyond the scope of this thesis as I only consider local magnetic-field-like Hamiltonians for metrological purposes. Another interesting subtlety is that several of the assumptions underlying the derivation of the quantum Cramér-Rao bound are not well justified in the limit of very few shots. In that regime, alternate metrics such as probably approximately correct (PAC) metrology may be more appropriate [98]. Nonetheless, the relevant results of this thesis are in the regime where the quantum Cramér-Rao bound is well justified, and therefore I use the SQL and HL as a point of comparison throughout.

Clearly, the presence of the QFI in the quantum Cramér-Rao bound tells us of the importance of creating and maintaining entanglement for quantum sensing. As I outlined (in a quantum computing context) in Section 1.2, this is a daunting task plagued by the inherent noise from environmental interactions. Despite recent advances in the implementation of QEC [53–55, 58–62], an enormous amount of work remains to make scalable sub-threshold error correction practical. Consequently, all quantum sensing schemes for the foreseeable future will be afflicted by noise, which is no less dangerous in a quantum sensing context: it is well known that noise hinders a probe state's capability as a sensor [99–102], and that even the smallest noise levels make it impossible to reach the HL. This creates a pressing need for quantum sensing protocols that account for, mitigate, or otherwise acknowledge the presence of noise. Chapter 6 is a response to this need.

*Pfffff. Facts are meaningless. You can use facts to prove anything that's even remotely true.*

— Homer Simpson

# 2 Background

## Contents


**2.1 Simulating quantum devices . . . . . 11**
**2.2 Variational quantum algorithms . . . . . 17**
2.2.1 Methods . . . . . 17
2.2.2 Applications . . . . . 20
2.2.3 Barren plateaus, trainability, and simulability . . . . . 23
2.2.4 Trainability beyond barren plateaus . . . . . 28
**2.3 Quantum error mitigation . . . . . 30**
**2.4 Calculating spectral properties of Hamiltonians . . . . 34**
2.4.1 Quantum phase estimation . . . . . 34
2.4.2 Variants of phase estimation . . . . . 36
2.4.3 Thermal properties . . . . . 37
2.4.4 Excited states . . . . . 39
2.4.5 Eigenvalue gaps . . . . . 40
**2.5 Classical shadows . . . . . 41**
2.5.1 Outline . . . . . 41
2.5.2 Convergence and unitary ensembles . . . . . 43


Following the general introduction to the field in Chapter 1, I now present a summary of more specialized topics in the literature. Quantum computing and quantum sensing are extremely active fields, and a broad overview of even very shallow depth would greatly exceed the scope of this thesis. Because the chapters of this thesis relate to several rather distinct sub-fields of quantum computing and quantum sensing, this review focuses closely on aspects of the literature that are

directly related to the work I present in this thesis.

## 2.1 Simulating quantum devices

With quantum computing currently in the limited state outlined in Section 1.2, our study of quantum devices must currently be conducted using analytical tools and classical computers. Since analytic study can be extremely challenging, and typically requires simplifying assumptions that may not hold in practice, it is critical to develop the capability to simulate quantum devices numerically on classical computers. Even in the era of fully-fault-tolerant quantum computing, classical simulation would still remain a valuable tool for developing, benchmarking and validating quantum algorithms. Of particular value is the ability to 'peek under the hood' of quantum dynamics, viewing individual quantum amplitudes, inner products, probabilities, non-physical quantities, simultaneous measurements of non-commuting observables, and other features that would not be practical to access on a real quantum device (or would require many repeated circuit executions). Furthermore, classical simulation allows one to explore regimes independent of the physical realities of the quantum device, such as differing noise models, error rates, and connectivity constraints. In this section, I briefly outline the aspects of classical simulation that are relevant to the work presented in this thesis.

The most conceptually straightforward method of simulating a quantum device is to directly simulate its state vector. An $n$-qubit pure state $|\psi\rangle$ can be written in some basis $\{|j\rangle\}$ as

$$|\psi\rangle = \sum_{j=0}^{2^n-1} c_j \, |j\rangle \,, \tag{2.1}$$

where the $c_j \in \mathbb{C}$ are the amplitudes of the state in this basis. By storing the vector of $2^n$ complex amplitudes $\vec{\psi} \equiv [c_0, \ldots, c_{2^n-1}]^T$, one can evolve the state under unitary transformations and track its dynamics. This approach, which I refer to as 'state-vector simulation' throughout this thesis, is the most general method of simulating a quantum device without noise. In the presence of noise (assumed throughout this thesis to be representable as a completely-positive-trace-preserving

map, or quantum channel), one may instead need to represent the full density operator $\rho$ of the system, which in the same basis may be written

$$\rho = \sum_{j=0}^{2^n-1} \sum_{k=0}^{2^n-1} \rho_{jk} |j\rangle \langle k| , \tag{2.2}$$

where the $\rho_{jk}$ are the elements of the density matrix. While one should be aware of the difference between the abstract density operator on the Hilbert space and its basis-dependent matrix representation, I will use the terms 'density operator' and 'density matrix' interchangeably throughout this thesis, as is common in the literature, and it should be inferred from context which is relevant. It should also be noted that some noisy quantum channels can be simulated using state-vector methods and Monte Carlo sampling of unravellings of the quantum channel, although this technique is not employed in this thesis. A great deal of work has been done on optimizing state-vector and density-matrix simulation. With modern software, one can easily leverage highly-optimized implementations with interfaces in high-level languages, taking full advantage of modern computing facilities such as distributed computing and GPU acceleration. Two such examples are the QuEST library [103] and its QuESTlink frontend for Mathematica [104], used in Chapters 3, 4, and 6; and TensorFlow Quantum [105], used in tandem with $\mathfrak{g}$-sim in Chapter 5. For the purposes of this thesis, however, exact classical simulation of quantum systems is merely a tool as well as a point of reference for the specialized classical simulation methods of Chapter 5. An excellent introduction to several facets of exact classical simulation of quantum computers can be found in Ref. [106]. This approach to classical simulation is termed *strong* simulation — in contrast to *weak* simulation, which computes limited properties such as samples or averages for (possibly limited sets of) expectation values, strong simulation allows one to obtain the full probability distributions for all possible measurement outcomes.

If we can simulate quantum computers exactly on classical computers, then why bother building them at all? The answer, of course, lies in the exponential resources required for exact state-vector or density-matrix simulation. For $n$ qubits, the state vector requires $2^n$ complex amplitudes to be stored (2.1), and the density matrix

requires $2^{2n}$ complex amplitudes (2.2). Assuming double-precision floating-point computer arithmetic (as is commonplace in most implementations), an $n$-qubit state vector requires $\approx 2^{n-26}$GB of computer memory, while an $n$-qubit density matrix requires $\approx 2^{2n-26}$GB.[1] Due to this exponential scaling, these exact methods are considered to be 'classically inefficient': storing the state vector for a mere 30-qubit pure state would entirely saturate the memory of the device on which I write this thesis, and by 52 qubits a pure state vector would exceed the memory of the world's (current) largest supercomputing cluster. In contrast, specialized classical simulation methods may, under certain constraints or approximations, have computational resource requirements scaling only polynomially in the system size. These simulations are deemed 'classically efficient' or 'scalable', and are critical for much of this thesis. In Figure 2.1, I compare the memory requirements for a selection of methods discussed in this section.

A classic example is the case of stabilizer states, which can be efficiently simulated on a classical computer (the Gottesman-Knill theorem [107, 108]). One can efficiently simulate the action of Clifford circuits on stabilizer states, since the $n$-qubit Clifford group is the normalizer of the $n$-qubit Pauli group (i.e. conjugation of a Pauli string by an element of the Clifford group yields another Pauli string). This formalism can even be extended by doping the circuit with a small number of $T$ gates, although the complexity will grow exponentially in the number of $T$ gates. Here the nature of efficient classical simulation is clear: in order to obtain polynomial scaling, constraints must be placed upon the quantum system, in this case the restriction to stabilizer inputs and Clifford circuits.

Although classical simulation methods sometimes arise from gates forming discrete structures (e.g. the Clifford group), there are efficient methods that arise from continuous structures as well. For example, some systems have underlying free-fermion mappings, allowing their continuous dynamics to be simplified in terms

[1] Note that for state-vector simulation, when computing the action of local gates, one does not need to represent the full $2^n \times 2^n$ matrix for a unitary $U$, but instead can simply apply $\mathcal{O}(2^n)$ operations to the state vector [106]. Since most quantum algorithms are decomposed into elementary gates, this typically allows simulation on the order of the Hilbert space dimension, rather than that of the operators acting upon it.

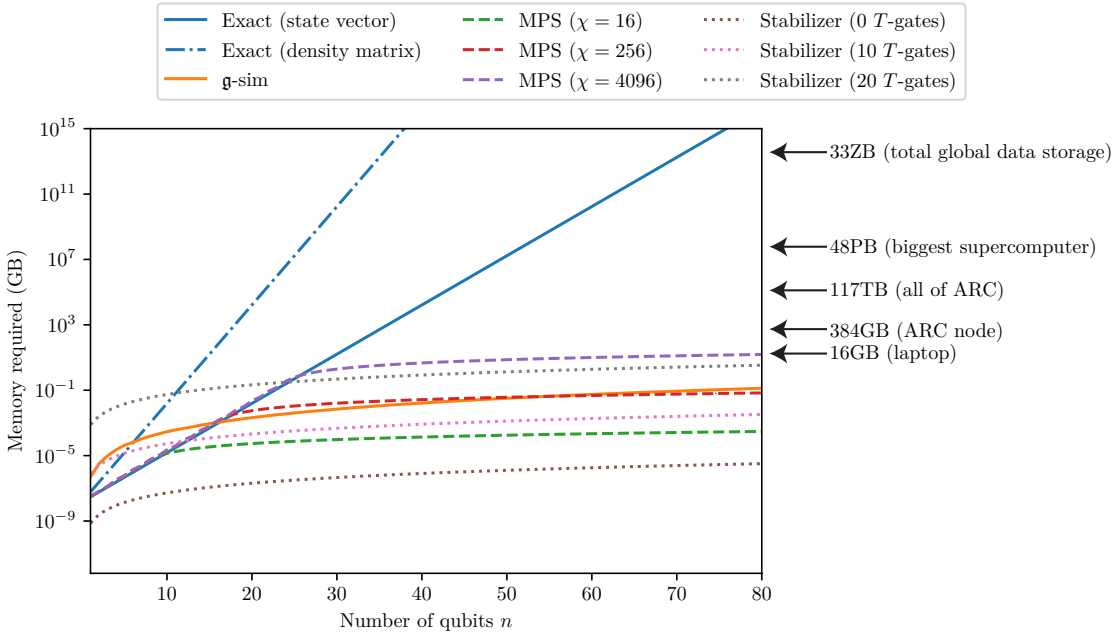



**Figure 2.1: Computer memory requirements for various classical simulation methods.** I plot the memory requirements (assuming double-precision floating point arithmetic) for a range of classical simulation schemes in gigabytes on a logarithmic scale. Exact simulation with state vectors (solid blue) and density matrices (dotted blue) requires resources exponential in $n$, making them not a scalable a approach. 𝔤-sim, the method I develop in Chapter 5, is an exact method of weak simulation, and its actual memory footprint for a particular benchmarking problem (outlined in Section 5.3.4) scales polynomially in $n$ (orange). In contrast to the other methods depicted here, the system state (classical representation) is a negligible contribution to the memory footprint, most of which is comprised of algebra adjoint representations. MPS methods with capped bond dimension $\chi$ are an approximate simulation method, with memory footprint polynomial in $n$ and exponential in $\chi$. I plot the memory requirements for $\chi = 16$ (green dashed), $\chi = 256$ (red dashed), and $\chi = 4096$ (purple dashed). Stabilizer-state simulation is an exact simulation method, with memory footprint polynomial in $n$ and exponential in the number of $T$ gates. I plot the memory requirements for 0 (brown dotted), 10 (pink dotted), and 20 (grey dotted) $T$ gates, assuming a $(4n^2 + 2n)$-bit tableau [107] that doubles in size with each $T$ gate. Along the memory axis, I note a number of key reference points for scale: the memory of a typical modern laptop (16GB), a single node of the Oxford ARC facility (384GB), the total memory of all nodes in the Oxford ARC facility (117TB), the total memory of the world's most powerful supercomputer (48PB), and an estimate of the total storage media capacity of the world (33ZB).

of these mappings. These methods [109, 110], now typically known as fermionic linear optics (FLO), and the very closely related methods for simulating matchgate circuits [111], serve as another example of exact efficient classical simulation arising from restricted sets of allowable transformations. Closely related — but still distinct — is the work of Somma on classical simulation of quantum circuits via the DLA [112, 113] using Wick's theorem, which I generalize in Chapter 5 to develop the $\mathfrak{g}$-sim framework. A deeper discussion of the nature of these methods will be provided in Chapter 5.

In addition to constraints on operations, measurements and initial states, one may also be able to leverage approximations to obtain efficient classical simulation. Depending on the method, these approximations may be controlled or uncontrolled. The most popular class of approximate methods are *tensor network* methods, which typically rewrite a quantum state as a contraction of tensors. Although in full generality a tensor-contraction representation of a quantum state will have the same complexity, under certain conditions the representation may allow for one to selectively discard parts of the tensor network which are of lesser importance. For example, in matrix-product state (MPS) methods, the state is written as a contraction of tensors at each qubit site, and the 'bond dimension' $\chi$ of the indices joining these (in general exponential in $n$) may be constrained for states with low entanglement. For weakly-entangled input states and $\log(n)$-depth local circuits, this can lead to efficient classical simulability. I provide a more detailed introduction to MPS methods in Section 3.2.2, and introductory resources may also be found in Refs. [114–116].

There is a continuing 'arms race' between quantum methods and advanced classical simulations. As the size of real quantum computers now exceeds the capabilities of state-vector simulation, it is crucial to obtain faithful, large-scale classical simulation results to serve as a ground truth for comparison. This is not only important for verification and benchmarking [117, 118], but scalable classical simulations can also power auxiliary tools for quantum computing — for example, to provide exact inputs for learning-based error mitigation strategies [119–122]. It

has also typically been the case that claims of quantum advantage are challenged by classical simulations. As I discussed in Section 1.2, in the case of random sampling experiments used to claim the earliest circuit-based quantum advantage, initial claims of quantum advantage [65] were refuted by classical simulations [66, 67], which themselves provided key benchmarks to shore up later quantum attempts [68, 69]. For practical problems, initial claims of quantum advantage by IBM on a practical simulation problem [52] were rapidly refuted by classical simulations achieving the same tasks [70–78], and one can only assume that the duel does not end there. It is not overly optimistic to say that classical and quantum methods inspire, challenge and motivate each other: bold claims of experimental quantum advantage invite challenge and provide a well-defined moving target for classical simulation methods to defeat, while the most advanced classical simulations provide benchmarks that experimentalists must cross to justify the 'quantum-ness' of their attempts. As I summarized in Section 2.2.3, classical simulability also threatens the entire field of variational computing via its link to trainability, a connection which I explore in Chapter 5.

The interplay between classical and quantum methods is not merely limited to rivalry and one-upmanship, however. It is not uncommon that this competition leads to the emergence of *quantum-inspired* classical methods, which leverage the insights of quantum algorithms to develop new classical algorithms (e.g. by dequantization). These include dequantized algorithms for linear algebra and machine learning [123–126], MPS approaches to the classical Fourier transform based on the QFT [127], and a range of other machine learning tasks [128]. Interpolating between classical machine learning and QML, it is now also understood that classical machine learning algorithms equipped with data from quantum computers can outperform their purely classical counterparts [129–133], and it is currently suspected that efficient classical simulations of quantum algorithms will play a key role in this capability [134, 135] due to the alignment of their quantum-mechanical structure and the quantum sources of data. Far less common, but still extant, are classical-inspired quantum methods, where an efficient classical simulation technique

is used as the basis for an improved quantum algorithm. One such example is the recent work by Somma *et al.* on 'shadow Hamiltonian simulation' [136], which utilizes a simulation algorithm similar to the classical framework of $\mathfrak{g}$-sim (Chapter 5), but with amplitudes stored in the register of a QPU.

In summary, classical simulations of quantum computers are not merely a useful tool for developing quantum algorithms, but a friendly adversary in the race for quantum advantage, a source of inspiration for new classical and quantum algorithms, and a key component in understanding the boundaries of 'quantum-ness'.

## 2.2 Variational quantum algorithms

### 2.2.1 Methods

Variational techniques have been a popular tool in studying quantum systems since well before the conception of quantum computers. With the variational method, one seeks to approximate the ground state of some Hamiltonian $H$ by optimizing parameters in a trial state $|\psi(\boldsymbol{\alpha})\rangle$, where $\boldsymbol{\alpha} \in \mathbb{R}^N$ is a set of variational parameters. One typically attempts to find a parameter configuration $\boldsymbol{\alpha}_*$ that minimizes the Hamiltonian expectation value

$$\boldsymbol{\alpha}_* = \arg\min_{\boldsymbol{\alpha}} \frac{\langle\psi(\boldsymbol{\alpha})| H |\psi(\boldsymbol{\alpha})\rangle}{\langle\psi(\boldsymbol{\alpha})|\psi(\boldsymbol{\alpha})\rangle}, \tag{2.3}$$

which provides an upper bound on the ground state energy of $H$ by the variational theorem. For small problems, it may be possible to compute $\boldsymbol{\alpha}_*$ analytically, but otherwise one typically resorts to numerical optimization techniques. If the ansatz is sufficiently capable of representing the ground state (for example, by building phenomenological insight into its functional form), then the optimized upper bound may be reasonably tight, and therefore the method may yield a good approximation to the true ground state.

These principles can be translated to a quantum computer, leading to the extremely popular field of variational quantum algorithms (VQAs), reviews of which can be found in Refs. [79, 135, 137]. As I outline in Section 2.4.1, ground state preparation on a fault-tolerant device can be achieved via QPE [20, 21].

Unfortunately, this requires deep circuits and many high-fidelity controlled operations, placing practical implementation well beyond the reach of current flawed quantum devices. In an attempt to develop an approximate alternative, Peruzzo *et al.* proposed the variational quantum eigensolver (VQE) [138], a hybrid quantum-classical implementation of the variational method. In this scheme, one employs a parametrized quantum circuit (PQC) $U(\boldsymbol{\theta}) = U_N(\theta_N)\dots U_1(\theta_1)$ (known as an *ansatz* throughout this thesis) to prepare a trial state $|\psi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})\,|\mathbf{0}\rangle$ on a QPU, where $|\mathbf{0}\rangle$ is some (normalized) fiducial input state (usually the $n$-qubit computational zero state $|0\rangle^{\otimes n}$). Here, the $\boldsymbol{\theta} = (\theta_1, \dots, \theta_N)$ are a set of real-valued classical parameters that can be optimized over. Throughout this thesis, I will restrict my attention to fixed ansatz structures with rotation gates $U_j(\theta_j) = \exp(-i\theta_j G_j)$ for some Hermitian generator $G_j$, although non-parametrized gates and adaptive ansätze are also possible [139–142]. Even though the Hilbert space will be exponentially large in the system size $n$, and thus expectation values will not be classically tractable in general, one can compute Hamiltonian expectation values $\mathcal{L}(\boldsymbol{\theta}) = \langle\mathbf{0}|\, U^\dagger(\boldsymbol{\theta}) H U(\boldsymbol{\theta})\,|\mathbf{0}\rangle$ on a *quantum* computer. One may then use a *classical* optimization algorithm to minimize $\mathcal{L}(\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$, attempting to find

$$\boldsymbol{\theta}_* = \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}). \tag{2.4}$$

This forms a hybrid quantum-classical loop, whereby a QPU is used to evaluate the objective function $\mathcal{L}(\boldsymbol{\theta})$, and these quantum evaluations are fed to a classical optimization algorithm on a classical coprocessor at each iterative step to find new parameters $\boldsymbol{\theta}$ that will decrease the objective function, informing the next values of $\boldsymbol{\theta}$ used to query the QPU for $\mathcal{L}(\boldsymbol{\theta})$. Crucially, individual evaluations of $\mathcal{L}(\boldsymbol{\theta})$ (and even its gradient [143–145]) merely require preparation and measurement of $U(\boldsymbol{\theta})$, rather than the deep, error-sensitive circuits required for QPE. Furthermore, the state of the hybrid algorithm is stored entirely in classical parameters $\boldsymbol{\theta}$, allowing progress towards the ground state to be 'saved' in spite of the poor coherence times of NISQ devices. As I describe in Section 2.2.2, this is a versatile tool that can be used as a building block for a variety of applications beyond ground-state

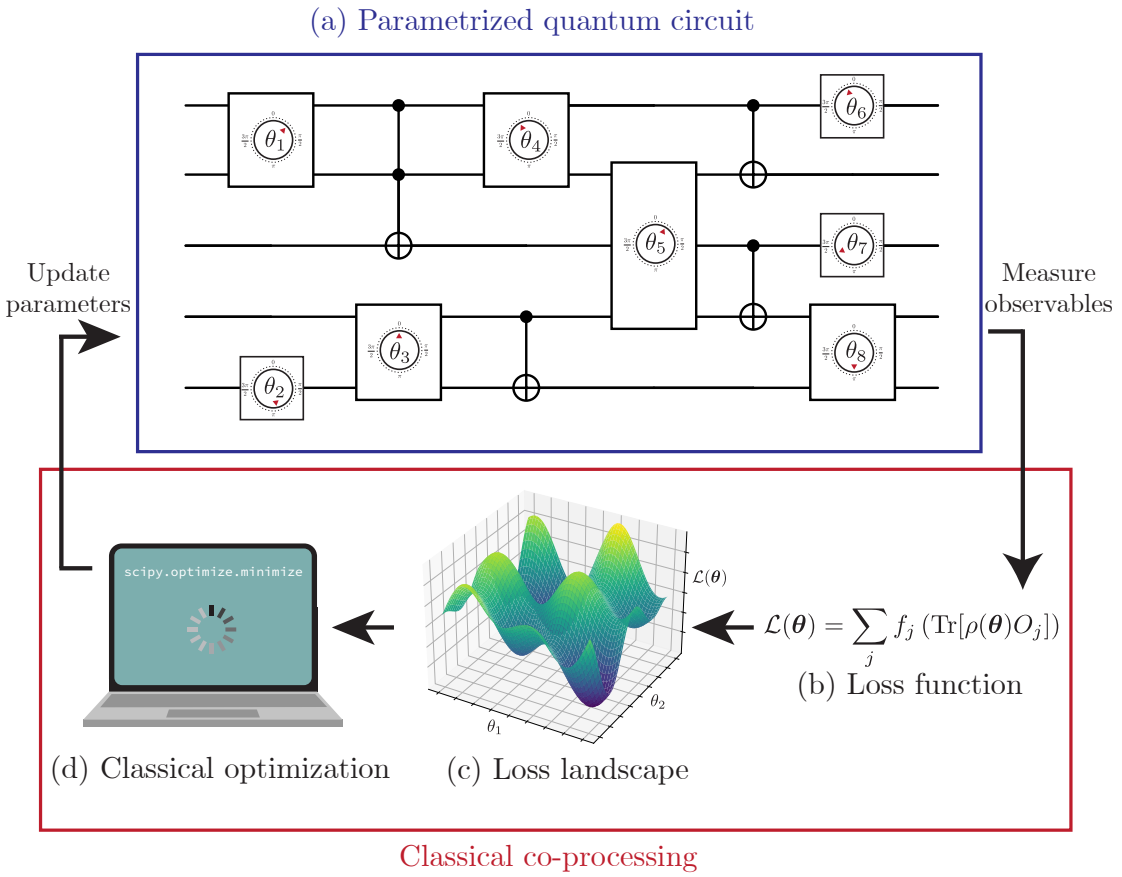



**Figure 2.2: Schematic depiction of a variational quantum algorithm.** This graphic depicts the typical features of a variational quantum algorithm. **(a)** A quantum circuit with classical parameters $\theta_j$, depicted here as tunable dials, is used as the core of the porcedure and repeatedly evaluated at different parameter values. **(b)** Measuring observables in the state prepared by the ansatz allows one to construct a scalar loss function $\mathcal{L}(\boldsymbol{\theta})$. **(c)** The scalar loss function $\mathcal{L}(\boldsymbol{\theta})$ forms an optimization landscape, which can in general be highly nontrivial and must be navigated with iterative numerical methods. **(d)** Using a classical optimizer, one can determine the next set of parameters with which to execute the circuit, aiming to reduce the loss $\mathcal{L}(\boldsymbol{\theta})$.

preparation, and has been a basis for a majority of proposals for advantage with NISQ systems. In Figure 2.2, I depict the essential features of a VQA.

While one can optimize a VQA with *any* black-box classical optimization algorithm for multivariate scalar functions, there are a range of specialized methods that can further enhance this. This simplest approach is, of course, to feed $\mathcal{L}(\boldsymbol{\theta})$ to a gradient-free [138] or gradient-based [143–145] classical optimizer. However, leveraging the quantum-mechanical structure of the problem can prove advantageous. For example, one can use variational methods to approximate imaginary-time

evolution [146], which drives the state towards the ground state of the evolving Hamiltonian (outlined in depth in Section 3.3.1). Ultimately, this is essentially transformation of the optimization landscape to a more 'natural' form, and is essentially equivalent to a quantum generalization [147, 148] of a method from classical machine learning known as natural gradient [149]. Another problem-inspired approach is to leverage the local trigonometric polynomial form of typical VQA loss functions [150, 151], determining a locally valid approximation to the landscape through which the optimizer can classically 'jump'. A particularly relevant approach to this thesis (leveraged in Chapter 4) is the CoVaR method [152], which exploits the fact that covariances $\langle OH\rangle - \langle O\rangle\langle H\rangle$ for arbitrary observables $O$ vanish in eigenstates of the Hamiltonian $H$. By using classical shadows (outlined in Section 2.5) to efficiently estimate a large number of constraints, and using Levenberg-Marquadt optimization to search for their shared roots, this method is particularly performant for shot-frugal VQA optimization when initialized near an eigenstate.

Early excitement surrounding VQAs was grounded in the intuition that some of the computational 'hardness' might be offloaded from the QPU to the classical coprocessor, while still retaining the power of evaluating objective functions in the exponentially large Hilbert space. Assuming that the variational ansatz is sufficiently expressive to contain a good solution, and the classical optimization algorithm succeeds in finding it, one might expect that a VQA would prove useful. However, there is trouble in paradise — not only are these two *very* generous assumptions, but they are in fact often directly at odds with one another. As I detail in Section 2.2.3, recent work by the community — including my own work outlined in Chapter 5 — has revealed the looming spectre of limitations that appear to be fundamental.

### 2.2.2 Applications

The volume of proposed applications of VQAs is truly vast; I do not give a comprehensive list here, but instead provide a brief overview of some of those most relevant to this thesis or popular in the literature. More comprehensive overviews can be found in relevant review papers [79, 137]. In addition to ground states via

VQE [138], VQAs have been proposed to study excitations in quantum systems. These include the approximation of excited states [152–160] and the calculation of Green's functions [161–165]; I discuss these in more detail in Section 2.4.4 as they are relevant background for Chapters 3 and 4. There have also been a range of proposals for using VQAs to solve linear algebra problems, including matrix-vector multiplication [166, 167] and solving linear systems [166–169] (i.e. a variational alternative to the HHL algorithm). In quantum sensing, variational techniques have been constructed to prepare the best possible probe state and measurement protocol [170–181].

A particularly popular variational technique is the quantum approximate optimization algorithm (QAOA) [23], which is a variational algorithm for the solution of combinatorial optimization problems. In essence, the method encodes the combinatorial optimization problem as a Hamiltonian whose ground state encodes the solution, and then uses VQE-like techniques to approximate this ground state. A recent (and much more comprehensive) review of QAOA can be found in [24]; here I describe the method in brief, as applied to its most popular target problem, the maximum cut (MaxCut) problem. Given a graph $G$ with edges $E$ and vertices $V$, the maximum cut (MaxCut) problem is to find a partition of V into two sets $S$ and $T$ that maximizes the number of edges $e \in E$ having endpoints in both $S$ and $T$. Encoding a partition as a bitstring $\boldsymbol{z} \in \{0,1\}^n$, its fitness for the MaxCut problem can be quantified by the approximation ratio $r$ defined as

$$r(\boldsymbol{z}) \equiv \frac{C(\boldsymbol{z})}{\arg\max_{\boldsymbol{z}} C(\boldsymbol{z})},\ C(\boldsymbol{z}) \equiv \sum_{(m,l)\in E} z_m(1-z_l). \tag{2.5}$$

For general graph problems, finding an exact MaxCut solution is known to be NP-hard [182]. However, the Goemans-Williamson (GW) algorithm [183] allows one to efficiently find an approximation with a ratio of at least $r_{\mathrm{GW}} \approx 0.878$. To obtain quantum advantage with QAOA, one must at least outperform this threshold. QAOA recasts a MaxCut problem as a VQE problem, where the goal is to find the ground state of the phase Hamiltonian

$$H_G = \frac{1}{2}\sum_{(m,l)\in E}(\sigma_m^z\sigma_l^z - I), \tag{2.6}$$

that depends on the underlying graph $G$. To prepare such a ground state, one applies a circuit

$$U(\boldsymbol{\beta},\boldsymbol{\gamma}) = \prod_{m=1}^{p} e^{-i\beta_m H_M} e^{-i\gamma_m H_G}\,, \tag{2.7}$$

where $H_M = \sum_{j=1}^{n} \sigma_j^x$ is the *mixing Hamiltonian*, applied to an initial state $|+\rangle^{\otimes n}$. By optimizing the variational parameters ($\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$) to minimize the expectation value $\langle+|^{\otimes n} U^\dagger(\boldsymbol{\beta},\boldsymbol{\gamma}) H_P U(\boldsymbol{\beta},\boldsymbol{\gamma}) |+\rangle^{\otimes n}$ and measuring the resulting state in the computational basis, one may construct approximate solutions to MaxCut. In correspondence to Equation 2.5, the approximation ratio of the solution generated by QAOA is defined as

$$r(\boldsymbol{\beta},\boldsymbol{\gamma}) \equiv \frac{\langle+|^{\otimes n} U^\dagger(\boldsymbol{\beta},\boldsymbol{\gamma}) H_G U(\boldsymbol{\beta},\boldsymbol{\gamma}) |+\rangle^{\otimes n}}{\langle\psi_{\mathrm{GS}}| H_G |\psi_{\mathrm{GS}}\rangle}, \tag{2.8}$$

where $|\psi_{\mathrm{GS}}\rangle$ is the ground state of the Hamiltonian $H_G$. Although optimal parameters can be identified for $p = 1$ [184], optimizing them for larger depth (where improved solutions can be found) remains a challenge. Indeed, several works show that general problem instances are likely to experience unfavorable optimization landscapes in the absence of any special underlying structure [185, 186]. There is therefore a compelling need to find better initialization strategies for deep-circuit QAOA, which I address in Chapter 5.

Two closely related applications are also important for this thesis: dynamics of quantum systems, and unitary synthesis / quantum compilation. Novel approaches to, and applications of, both are found in Chapters 3, 4 and 5. Simulating the time evolution of quantum systems is a very high-value problem. A range of methods have been proposed for this, including those based upon McLachlan's variational principle [29, 187], recompiling Trotter steps [188–193], variational fast-forwarding [188, 194], and learning-based techniques [192]. These approaches are not limited to unitary dynamics, and variants of these methods targeting Lindbladian evolution of open quantum systems have been proposed [167]. While this is typically for a fixed input state, a more general problem is that of unitary synthesis, wherein one attempts to find a circuit approximating some target unitary $V$ (in the case

of dynamics, the time-evolution operator $V = e^{-iHt}$). Variational techniques for unitary synthesis [142, 195–197] typically involve a *Hilbert-Schmidt test*, by which one seeks to minimize the loss function

$$\mathcal{L}_{\mathrm{HST}}(U(\boldsymbol{\theta}), V) \equiv 1 - \frac{1}{d^2} |\operatorname{Tr}\Big(U^\dagger(\boldsymbol{\theta}) V\Big)|^2, \tag{2.9}$$

thus ensuring that the ansatz $U(\boldsymbol{\theta})$ is close to the target unitary $V$ (up to a possible global phase). Of particular note is the learning-based approach of Caro *et al.* [197], who prove results relating to out-of-distribution generalization for learning unknown unitaries. Notably, these results allow one to learn the action of a unitary on a broad class of entangled states having trained on a training set of just product states.

At this point, I note an important distinction. Learning-based techniques using quantum computers, broadly known as quantum machine learning (QML), often (but do not always) use VQA methods as their backbone. As this thesis is not concerned with non-variational QML methods, I refer to VQAs and QML somewhat interchangeably throughout. Nonetheless, it is important to note that non-QML VQAs (e.g. VQE [138]) are problem-focused, typically aiming to solve a defined problem and utilizing fixed input states, whereas QML problems are data-focused, typically aiming to learn some property from a training dataset (and possibly generalizing this property to unseen test data). While a full accounting of variational QML applications is beyond the scope of this thesis, notable applications include autoencoders [198–200], generative models [201–204], and supervised learning [143, 205–207], the latter of which I explore further in Chapter 5. More complete reviews can be found in Refs. [208–210].

### 2.2.3 Barren plateaus, trainability, and simulability

The early study of VQAs was fuelled by an unchecked optimism, which was then tempered by the discovery of the *barren plateau* phenomenon. A comprehensive modern review of the topic can be found in Ref. [135], but I nonetheless provide an overview here. Early attempts at VQAs typically used random circuits, since they are simple, can be constructed in a hardware-efficient manner, and were heuristically

expected to have good expressibility. McClean *et al.* observed that for a wide class of reasonable random ansatz circuits, the gradient of the loss function vanishes exponentially on average in the system size, attributing to the approximate unitary 2-design characteristic of these random circuits [211]. For the purposes of this thesis, this is captured formally in the following definition.

**Definition 1** (Barren plateau)**.** *Consider a VQA constructed from a PQC $U(\boldsymbol{\theta})$, and a training set $\mathcal{S}$ consisting of initial state and measurement operator pairs $(\rho_j, O_j)$. Let the loss function $\mathcal{L}(\boldsymbol{\theta})$ be defined as*

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{|\mathcal{S}|} \sum_{(\rho_j, O_j) \in \mathcal{S}} c_j \operatorname{Tr}\Big[U(\boldsymbol{\theta}) \rho_j U^\dagger(\boldsymbol{\theta}) O_j\Big]. \tag{2.10}$$

*The VQA is said to exhibit a barren plateau if for all $\theta_\mu \in \boldsymbol{\theta}$, the variance of the gradient of its loss function vanishes exponentially in system size $n$, i.e.*

$$\operatorname{Var}_{\boldsymbol{\theta}}[\partial_\mu \mathcal{L}(\boldsymbol{\theta})] \leq \mathcal{O}(b^{-n}) \tag{2.11}$$

*for some constant $b > 1$, and where the subscript indicates that the variance is taken with respect to $\boldsymbol{\theta}$.*

Since observables in quantum computers are measured statistically by repeated sampling, this exponentially-flat loss landscape structure means that gradient-based optimization methods require exponentially many circuit repetitions. In modern parlance, these circuits are therefore said to be *untrainable*, since the exponential scaling of required circuit repetitions precludes efficient scaling of the method (that is, estimating an observable to precision $\epsilon$ requires $\mathcal{O}(1/\epsilon^2)$ shots, and the observables in question vanish at an exponential rate).

Further investigation into the barren plateau phenomenon quickly revealed that it is not limited to random circuits. Until recently, the causes of barren plateaus were attributed to a 'zoo' of distinct phenomena. Barren plateaus were independently attributed to circuit expressivity [212, 213] (that is, the relative proportion of accessible states in Hilbert space), entanglement [214–216], globality of cost functions [217], hardware noise [218], and choice of target problem [219].

In each of these cases, there was seemingly little commonality between the causes of barren plateaus, and it was unclear whether this 'zoo' of phenomena could be unified in a single framework. Today, the picture is mostly unified, but many ingredients are needed to compose the full story.

One such ingredient is the connection between barren plateaus and other landscape features. While a barren plateau consists of a loss landscape that is exponentially flat on average, a *narrow gorge* is an exponentially-narrow well around a minimum in a VQA optimization landscape [217]. These phenomena concern the gradients of loss functions — as for the loss functions themselves, another relevant feature is *cost concentration*, where the loss function itself is concentrated around the mean value. It was shown in Ref. [220] that these three features are equivalent.[2] Consequently, barren plateaus can be diagnosed by considering loss function *differences*, which is mathematically substantially easier than gradient variances (2.11). A closely related unfortunate realization is that one cannot circumvent the problem of vanishing gradients by using gradient-free optimizers [221], since these optimizers will still require exponentially many circuit repetitions to choose a viable search direction. Furthermore, the links between barren plateaus and exponential loss concentration also connects this to analogous phenomena in quantum kernel methods [222].

Major progress in understanding the barren plateau phenomenon was achieved with the realization that it is intimately connected to quantum optimal control theory [213, 223, 224], wherein one seeks to optimize the dynamics of a quantum system by applying time-dependent controls. The dynamical Lie group (DLG) $\mathcal{G}$ of a Hamiltonian with time-dependent controls (or equivalently in this case, a PQC $U(\boldsymbol{\theta})$) is the set of all unitaries that can be realized by such a system. A formal introduction to these tools is given in Chapter 5, but for now it suffices to think of the dynamical Lie algebra (DLA) $\mathfrak{g}$ as the vector space formed by a set of anti-Hermitian

[2]It should be noted that Definition 1 is the standard definition that has appeared throughout most of the literature, and allows for well-behaved deviations from exponential flatness like narrow gorges. However, more recently Definition 1 has been known as a 'probabilistic' barren plateau [135], in contrast to deterministic barren plateaus induced by noise [218], which truly flatten the entire landscape. The distinction is important, but not relevant to this thesis.

generators for $\mathcal{G}$. The DLA dimension $\dim(\mathfrak{g})$ effectively quantifies the expressivity of a PQC — a low $\dim(\mathfrak{g})$ means a smaller manifold of unitaries realizable by $U(\boldsymbol{\theta})$, whereas a full-rank DLA with $\dim(\mathfrak{g}) = 2^{2n-1}$ (i.e. $\mathfrak{g} = \mathfrak{su}(2^n)$) indicates that the PQC can realize any unitary on the $n$-qubit Hilbert space. These tools originate from quantum optimal control theory, and are invaluable in the study of VQAs. Miscellaneous applications of the DLA in VQAs include overparametrization [225], quantifying the privacy of QML models [226], and classical simulation (Chapter 5). Crucially for the understanding of barren plateaus, Larocca *et al.* conjectured that expressibility-induced barren plateaus are a consequence of the DLA being 'too large' [213]: very expressive ansätze have a large DLA dimension $\dim(\mathfrak{g})$, to which the gradient variance can be directly connected. Indeed, ansätze with $\dim(\mathfrak{g}) \in \text{poly}(n)$ are provably free of barren plateaus [227, 228] (unless induced by other factors, such as noise [218] or global cost functions [217]); meaning that circuits with certain highly-symmetric structures, such as matchgates [111] and permutation-equivariant circuits [229], can be efficiently trained. More recently, two papers submitted days apart [227, 228] (mostly) unified the zoo of barren plateau phenomena in terms of the DLA, showing that it alone can account for barren plateaus induced by circuit expressivity, entangled input states, global cost functions, and some types of noise. This showcases the primacy of the DLA as a tool for studying VQAs and QML, which I will use extensively in Chapter 5 of this thesis. Further generalizations are underway — recently, some of these results were extended to ansätze for which the Lie-algebraic framework is less apropriate using a framework based on Jordan algebras [230], and generalized notions of globality have yielded more nuanced insights [231].

A variety of other architectures that are provably barren-plateau-free have also been identified. These include shallow hardware efficient ansätze [217], quantum convolutional neural networks [232], $U(1)$-equivariant systems [233], some quantum generative models [234, 235], and circuits with small-angle initializations [236–238]. It is important to question the extent to which the constraints of these problem structures limit their prospects of a quantum advantage. At the end of Section 2.2.1,

I hinted that expressivity and trainability of VQAs were at odds with one another, which becomes clear in the Lie-algebraic context of expressibility-induced barren plateaus: to achieve a quantum advantage one simultaneously desires a low $\dim(\mathfrak{g})$ for trainability, but a high $\dim(\mathfrak{g})$ for circuit expressivity. This was an early clue that barren plateaus could be understood as a 'curse of dimensionality', the implications of which go far deeper than a mere expressibility-trainability tradeoff. Supposing a loss function of the form in Equation 2.10, the terms evaluated on the quantum processor are of the form $\mathrm{Tr}\left[U(\boldsymbol{\theta})\rho U^{\dagger}(\boldsymbol{\theta})O\right]$ for some input state $\rho$ and observable $O$. These may therefore be written as a Hilbert-Schmidt inner product $\langle \rho(\boldsymbol{\theta}), O\rangle$, where $\rho(\boldsymbol{\theta}) = U(\boldsymbol{\theta})\rho U^{\dagger}(\boldsymbol{\theta})$ is the ansatz-evolved state. As explained in Ref [135], optimizing the loss function is then equivalent to trying to anti-align two exponentially large vectors via their inner product, which is inherently troublesome as this will on average be exponentially small and concentrated, yielding a barren plateau. Barren plateaus are then typically provably absent when some special structure leads to more careful alignment between $\rho(\boldsymbol{\theta})$ and $O$, such as a low-expressivity $U(\boldsymbol{\theta})$ limiting movement in Hilbert space (small $\dim(\mathfrak{g})$), and not-too-entangled input states and local observables leading the inner product to depend on some effective subspace, preventing it from being drowned out by the exponential Hilbert space dimension.

Worryingly, this seems at odds with the motivation for VQAs in the first place: carefully constraining the problem to avoid exploring large portions of Hilbert space or high entanglement would intuitively seem to drain it of the non-trivial 'quantum-ness' that one hoped to exploit in the first place. Such problems of restricted structure are often efficiently simulable on a classical computer, negating any prospect of quantum advantage. This hints at a deeper connection between trainability and classical simulability of VQAs, which was more carefully outlined by Cerezo *et al.* in Ref. [134], leading to a dramatic shift in how the community thinks about VQAs. While it has not yet been proven that trainability and simulability are equivalent in general, it has been observed that not only is every known barren-plateau-free architecture classically simulable, but also that in each of these cases, the method of classical simulation is related to the same mathematical tools used to prove

the absence of barren plateaus. For example, quantum convolutional neural networks can be proven barren-plateau-free using a tensor-network formalism [232], and can be classically simulated using tensor-network methods [134, 239]. Another example observed by Ref. [134] was my own work presented in this thesis: circuits with low expressivity can be proven barren-plateau-free using Lie-algebraic methods [213], and in Chapter 5 I use these same methods to develop a simulation method that is efficient on a classical computer for these circuits. The link between trainability and simulability is arguably the most pressing issue in VQAs — should it prove absolute, this would necessitate an orthogonal approach. In the meantime, while barren plateaus may never be fully eliminated, a range of techniques are being explored to mitigate their effects [240].

### 2.2.4 Trainability beyond barren plateaus

The literature on trainability of VQAs has been predominantly focused on barren plateaus, and with good reason: if one cannot even obtain the gradients or loss differences efficiently in the first place, then the question of how to effectively use them for classical optimization is moot. Nonetheless, even if one can defeat barren plateaus and efficiently compute gradients, this leaves a host of other issues to be addressed in VQA trainability. In essence, barren plateaus are the dominant difficulties for the QPU when training a VQA, but there are also challenges for the classical coprocessor which I address here. Classical optimization tasks can be extremely difficult, and the classical component of VQAs are no exception — indeed, training VQAs is in general NP-hard [241]. In practice, the issue is that VQA optimization landscapes can be plagued with many poor-quality local minima [242–244]. The complicated nature of iterative optimization algorithms makes analytic study extremely challenging, and thus when compared with the barren plateau problem, far less is known about this side of trainability [135].

Nonetheless, it is known that some conditions lead to favourable optimization landscapes. To this end, a particularly desirable phenomenon is *overparametrization*. In classical machine learning, overparametrization is a phenomenon arising when

using a neural network with a capacity larger than necessary to represent the distribution of the training data. Although one may naïvely expect this to lead to overfitting and optimization challenges, it can lead to improved performance [245–247] and even provable convergence [248, 249]. Larocca *et al.* generalized this to VQAs (and by extension variational QML) in Ref. [225], showing that overparametrization of a VQA is provably attained when one has $N_p \geq M_c$ parameters, where $M_c \leq \dim(\mathfrak{g})$ is a threshold that depends on the DLA dimension of the PQC. This proof relies upon the same Lie-algebraic tools mentioned in the previous for the study of barren plateaus [213, 227, 228], which I also leverage for classical simulation in Chapter 5. When a VQA crosses the overparametrization threshold $M_c$, spurious local minima disappear (e.g. they become saddle points which can be escaped from), leading to assurances of optimizability. I utilize this phenomenon throughout Chapter 5, and analyze it in Section 5.4.1 at previously unattained system sizes beyond the reach of state-vector simulation methods.

Another appealing avenue for improving trainability is the use of pre-training, also known as a warm start. Although the optimization landscape of a VQA may be a perilous field of poor-quality local minima overall, there is some hope that with the right heuristic, one may be able to initialize the optimization close to a good minimum and find it even if there was little hope of doing so from a random initialization. Furthermore, pre-training can potentially mitigate barren plateaus — although the landscape may be exponentially flat on average, one may be able to initialize sufficiently close to a narrow gorge [220] to obtain sufficiently large gradients. While this mitigation of barren plateaus is a trait shared with some small-parameter initialization strategies [236–238, 250], in those cases it is typically unlikely that a good minimum will be found near the initialization point. In contrast, when successful, pre-training can be advantageous for both aspects of trainability (barren plateaus and successfully navigating the optimization landscape). A broad range of pre-training strategies have been trialled, with many showing early promise in numerical simulations [207, 251–262]; in Chapter 5 I utilize my '$\mathfrak{g}$-sim' framework to develop new pre-training strategies for VQE and QAOA.

Understanding pre-training is of particular relevance for variational quantum dynamics. When one initializes the ansatz close to the identity and seeks to recompile small dynamical steps one at a time (as is the case in Refs. [189–191] and Chapter 4), the optimization is effectively warm-started. Provided the timestep $\Delta t$ is sufficiently small, the initial evolution step will be close to the identity (near which the ansatz is initialized), and provided that the previous step was well-optimized, the total time-evolution unitary at the next step will be close to the ansatz. In Ref [263], Puig *et al.* identify key caveats in this approach: while one can construct conditions under which the optimization is guaranteed to find the adiabatic minimum, it is possible for the true minimum to 'jump' from the adiabatic minimum between timesteps. In some cases, a 'fertile valley' in optimization space connects the two, allowing the variational approximation to continue to track the dynamics accurately (to the best of its expressive power). It is currently poorly understood what conditions are needed for this, leaving open the question of whether variational dynamics can be used to obtain some degree of quantum advantage.

## 2.3 Quantum error mitigation

To squeeze maximal value out of NISQ devices, it is prudent not only to design compatible algorithms, but to also attempt to reduce the impact of inherent flaws in these devices. Of course, the ideal way to do so would be via QEC, but even the most advanced experimental demonstrations of QEC are not sufficient for practical computations [53, 55, 58–62, 264] — QEC is extremely daunting, requiring low base hardware error rates, fast and numerous mid-circuit measurements, real-time classical decoding, and a host of other challenges. Clearly, in the NISQ era, less demanding alternative strategies are required. A weaker, but profoundly more permissive alternative is *quantum error mitigation*, wherein one attempts to mitigate the impact of errors by post-processing measurement outcomes of an ensemble of circuit runs at the same (or higher) noise level inherent to the platform, rather than trying to directly reduce the error rate itself. In contrast to QEC, error mitigation does not improve the quantum coherence of individual quantum states or circuit

runs, but rather combines an ensemble of noisy outputs in a way that reduces the impact of noise *on the final result*. A comprehensive recent review of quantum error mitigation going well beyond the treatment of this section can be found in Ref. [265].

The earliest quantum error mitigation technique, and most relevant to this thesis, is zero-noise extrapolation (ZNE) [29, 266]. While reducing the fault rate $\lambda$ of a quantum circuit requires either QEC or engineering improvements, it is often straightforward to *increase* the fault rate. This could be by repeating idempotent gates such that noisy resolutions of the identity are inserted (e.g. composing three noisy CNOTs is equivalent to one noisy CNOT with triple the fault rate), known as unitary folding [267], or by hardware-specific manipulations like pulse stretching [268]. In any case, taking $\lambda_0$ to be the base fault rate of the hardware, by measuring an observable $\mathrm{Tr}[O\rho_\lambda]$ (where the subscript in $\rho_\lambda$ indicates the dependence of the quantum register's state on the fault rate) at a range of boosted fault rates $\lambda \geq \lambda_0$ and extrapolating the resulting curve to $\lambda = 0$, one can obtain an estimate of the noiseless observable $\mathrm{Tr}[O\rho_0]$. There is some flexibility in the choice of extrapolation technique — simply fitting an exponential curve can prove particularly effective for Markovian noise [269, 270], while Richardson extrapolation has been shown to more accurately estimate the noiseless value when the noise and system size are small [266], can be applied independently of the underlying noise model, and has provably optimal node spacings [270] that minimize the number of needed hyperparameters. For the latter reasons, I restrict my attention to Richardson extrapolation in this thesis.

Another popular error mitigation technique is probabilistic error cancellation (PEC) [266, 271, 272]. In PEC, one notes that a noiseless expectation value can be written as a linear combination of noisy expectation values, where the coefficients are determined by tomography [271] or by a learning-based approach [272]. Remarkably, this enables one to *fully* cancel out the noise-induced bias, albeit at the significant expense of a sampling overhead that grows exponentially in the circuit fault rate $\lambda$. The two methods discussed thus far neatly illustrate the difference between *passive* and *active* error mitigation techniques — while passive techniques such

as ZNE rely entirely on postprocessing, active techniques like PEC apply more invasive techniques like alterations to the circuit across repetitions or simultaneous use of multiple state copies.

Beyond these demonstrative examples, numerous other error mitigation techniques have arisen in the literature. One can exploit known constraints on the state, such as symmetries [273, 274] — while this bears some similarity to the checks used to construct QEC syndromes (e.g. parity checks), it is a less demanding protocol as it requires only post-hoc detection of errors, omitting their contribution rather than correcting them and retaining that circuit execution's contribution to the final result. Most quantum algorithms target the preparation of a pure state, and by exploiting purity constraints one can also mitigate errors, a practice known as virtual distillation [275] or error suprression by derangement [276]. In eigenstate preparation problems, one can combine multiple solution attempts via a technique known as subspace expansion [277, 278], whereby one constructs a low-dimensional subspace spanned by approximate solutions, which can be diagonalized to identify superior solutions consisting of linear combinations of the approximate ones. Finally, learning-based error mitigation techniques use classical machine learning techniques to attempt to learn the inverse map between noisy and ideal circuits [119–122], where the training data consists of measurements from the target device and their noiseless counterparts, typically computed using efficient classical simulation methods (see Section 2.1). Quantum error mitigation is a diverse field, and I again refer the interested reader to the treatment of Ref. [265] for a complete review.

A common feature of all error mitigation techniques is the overhead of additional resource requirements, typically circuit repetitions (shots). Consider an $N$-shot estimator $\overline{O}_N$ of some expectation value $O$. The overall faithfulness of this estimator is typically quantified via the mean-squared error (MSE)

$$\mathrm{MSE}[\overline{O}_N] \equiv \mathbb{E}\left[(\overline{O}_N - O)^2\right] = \underbrace{\mathbb{E}\left[\overline{O}_N - O\right]^2}_{(\mathrm{Bias}[\overline{O}_N])^2} + \underbrace{\mathbb{E}\left[\overline{O}_N^2\right] - \mathbb{E}\left[\overline{O}_N\right]^2}_{\mathrm{Var}[\overline{O}_N]}, \tag{2.12}$$

where the bias quantifies the systematic error of the estimator, and the variance quantifies the random error (i.e. the spread of possible predictions). At a fixed

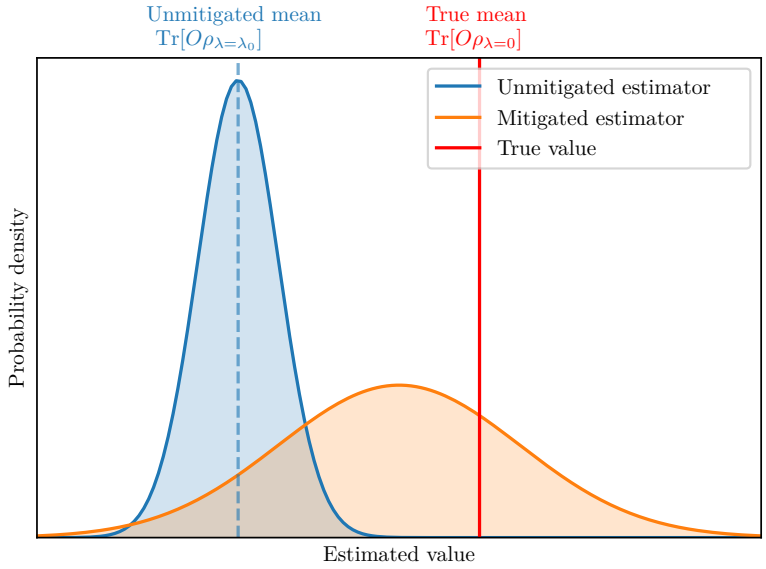



**Figure 2.3: Bias-variance tradeoff in quantum error mitigation.** A cartoon depiction of the bias-variance tradeoff in quantum error mitigation. The unmitigated estimator (blue) has comparatively low variance, leading to a narrower distribution than the error-mitigated estimator (orange). However, at the cost of larger variance, the error-mitigated estimator has reduced (but not eliminated) bias, being closer to the true value (red). Such an alteration to the estimator may in principle increase, decrease, or leave unchanged the MSE; ascertaining this in a sensing context is part of the goal of Chapter 6.

shot budget $N$, using an error-mitigated estimator typically reduces the bias at the cost of increasing the variance, the so called *bias-variance tradeoff* of quantum error mitigation. This is depicted in Figure 2.3, and its relevance to quantum sensing is the subject of Chapter 6.

One must also take care when claiming quantum advantage from error-mitigated computations. Not only does error mitigation come with unfavourable repitition overheads (e.g. the exponential measurement overheads of PEC) that can negate asymptotic improvements in runtime, but the presence of noise itself can lead to classical simulability. In addition to the classical simulation techniques discussed in Section 2.1, some approximate simulation techniques are classically efficient for sufficiently noisy circuits [279] — for example, schemes can be devised that exploit the tendency of noise to strongly suppress non-local correlations. Indeed, it has been noted that any circuit for which error mitigation scales efficiently with the amount of noise must also be classically simulable [280]. At least one of the

aforementioned classical simulations of IBM's 'quantum utility' experiment [52] exploited properties of the noise to achieve efficient simulation [76].

Although error mitigation usually appears in a quantum computing context, its relevance to this thesis is mostly in the context of quantum sensing. Since noise is as much of an affliction for quantum sensing as it is for quantum computing, why not port the tools used to compensate for the flaws of NISQ computers to noisy quantum sensors? This is a comparatively under-explored avenue of research. While there has been work suggesting some potential for active error mitigation techniques in quantum sensing [281–284], there has been no such evidence in favour of passive techniques, and in fact slight evidence to the contrary [285]. Chapter 6 is dedicated to resolving this gap in the literature, contrasting ZNE to learning-based inference techniques [286] to critically assess the potential of error mitigation in quantum sensing.

## 2.4 Calculating spectral properties of Hamiltonians

### 2.4.1 Quantum phase estimation

Perhaps the most promising application of quantum computers is the study of quantum systems. As we learn early in our study of quantum mechanics, the most important properties of a quantum system are encoded in the eigenstates and eigenvalues of its Hamiltonian. A vast range of methods have been developed to leverage quantum computers in computing these, either directly or indirectly. The most fundamental method, to which all others must be compared, is the QPE algorithm [16, 18]. Given an eigenstate $|\psi\rangle$ of a unitary operator $U$ with eigenvalue $e^{i\theta}$ such that $H|\psi\rangle = e^{i\theta}|\psi\rangle$, QPE allows one to estimate $\theta$ to precision $\epsilon$ using $\mathcal{O}(1/\epsilon)$ queries to $U$. This is accomplished via the circuit in Figure 2.4, whereby one resolves the eigenphase of the $n$-qubit system (lower subregister) to $B$ bits of estimation precision (upper subregister). Following the application of Hadamard gates to the estimation subregister, the composite system's state is $|\Psi\rangle = |\psi\rangle \otimes \sum_{b=0}^{2^B-1} |b\rangle / \sqrt{2^B}$, where $|b\rangle$ indicates the computational basis state

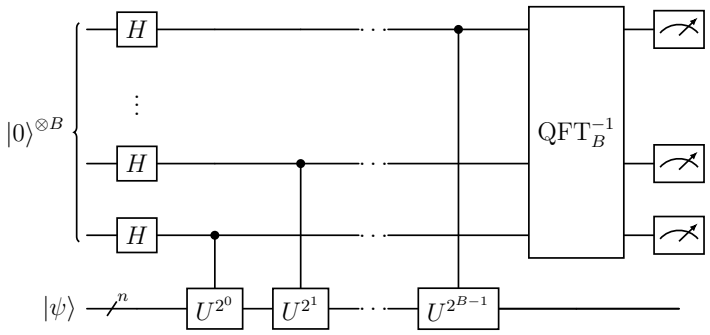



**Figure 2.4: Standard circuit for quantum phase estimation (QPE).** Controlled-time-evolution operations imprint phases corresponding to each of $B$ bits of binary precision in the estimation subregister, each corresponding to a binary digit of the desired eigenphase. An inverse QFT on the estimation subregister then transforms these imprinted phases to the actual binary representation of the eigenphase in the computational basis, which can then be directly measured.

corresponding to the binary bitstring of the integer $b$. Now, applying a controlled-$U^{2^m}$ gate to the system subregister (controlled on the $m$-th qubit) for each qubit $m$ in the estimation subregister, each bitstring in the superposition accumulates a phase proportional to its place value, yielding

$$|\Psi\rangle = \frac{1}{\sqrt{2^B}} \sum_{b=0}^{2^B-1} \left( e^{2\pi i \theta b} |\psi\rangle \otimes |b\rangle \right). \tag{2.13}$$

If one then applies an inverse QFT, clearly from the form of Equation 2.13 this results in the state $|\Psi\rangle = |\psi\rangle \otimes |\theta\rangle$, where $|\theta\rangle$ is the binary representation of $\theta$ to $B$ bits of precision. Measuring the estimation subregister in the computational basis then yields the binary representation of $\theta$ to precision $B$ — if $\theta$ is not exactly representable to $B$ bits of precision, as is usually the case, then this may be a superposition with highest weight in the closest rounded value.

It should be noted immediately that QPE is not a practical algorithm in the NISQ era, since the operation of the controlled-$U^{2^m}$ gates require deep circuits (with a depth growing exponentially in the bits of precision $B$). Nonetheless, it remains an appealing target in the pursuit of fully-fault-tolerant quantum computing, and forms a fundamental building block for many other quantum algorithms. In particular, consider relaxing the assumption that the input state $|\psi\rangle$ is an eigenstate of $U$, and

is instead a general state $|\psi\rangle = \sum_j c_j |\psi_j\rangle$ where $|\psi_j\rangle$ are the eigenstates of $U$. By linearity, the QPE algorithm will yield the superposition $\sum_j c_j |\psi_j\rangle\otimes|\theta_j\rangle$, where $\theta_j$ are the eigenphases of the $|\psi_j\rangle$, and measuring the estimation subregister will yield the eigenphase $\theta_k$ with probability $|c_k|^2$. Furthermore, upon measurement, the system subregister will collapse to the corresponding eigenstate $|\psi_k\rangle$, allowing one to prepare the system in the eigenstate corresponding to the measured eigenphase. Given some Hamiltonian $H$, QPE can therefore be used to estimate the eigenphases of $H$ and prepare the corresponding eigenstates [20, 21] by applying quantum simulation algorithms to implement the necessary controlled-$e^{-iHt}$ gates, since $H$ and its time-evolution unitaries $e^{-iHt}$ share the same eigenstates. In addition to modelling physical systems, this can be used to solve a variety of optimization problems that can be encoded in synthetic Hamiltonians [22–24] (modulo the below restrictions).

The ability to prepare an eigenstate $|\psi_k\rangle$ assumes an initial state $|\psi\rangle$ that has a strong overlap with $|\psi_k\rangle$. As I discuss in Chapter 3, one might attempt to realize this with adiabatic state preparation [21], system-related approximations (such as Hartree-Fock states in quantum chemistry), or variational approximations. Nonetheless, obtaining good initial states is a challenging task in general which can bottleneck even fault-tolerant protocols [287], which is a consequence of the exponential hardness of finding ground states of general Hamiltonians, even with quantum resources [288]. Good initial state overlap is critical to the shadow spectroscopy techniques I present in Chapter 3, though not to the methods for computing the density of states I develop in Chapter 4.

### 2.4.2 Variants of phase estimation

Studying the spectral properties of Hamiltonians is one of the most desired applications of quantum computing — numerous methods exist, and I will not attempt to cover them all. Nonetheless, I will outline some of the more relevant examples. QPE requires very deep circuits, but very few measurements (i.e. few circuit repetitions). One can vary this tradeoff — in the simplest case, a Hadamard test can determine

eigenphases with shallower circuits, but instead with the constraint of being shot-noise-limited (i.e. requiring $\mathcal{O}(1/\epsilon^2)$ samples to get an estimate to precision $\epsilon$, as opposed to just $\mathcal{O}(1/\epsilon)$ queries to $U$ for QPE). Interpolating between these extremes, one could also estimate each binary bit of the eigenphase separately, starting from the least significant bit, using a method known as iterative phase estimation [289]. Other variants of phase estimation have differences including randomized measurements [290], Bayesian approaches to tracking likely values of the phase [291], methods based on random walks [292], measurement schemes enhanced by machine learning [293], and adaptive precision in the Hamiltonian [294], to name just a few.

Of particular relevance to this thesis are methods that rely upon the Loschmidt echo $\langle\psi| e^{-iHt} |\psi\rangle$ (i.e. the overlap of some initial state $|\psi\rangle$ with its time-evolved counterpart $e^{-iHt} |\psi\rangle$), with the earliest such approach being that of Somma *et al.* [295]. Typically, the Loschmidt echo is measured by a Hadamard test (i.e. controlled-$e^{-iHt}$ with a single ancilla); as shall become apparent in Chapter 4, when measured for a range of times $t$, the Fourier transform of this signal provides initial-state-dependent information about the Hamiltonian spectrum. Variants of these approaches can be suitable for NISQ and early-fault-tolerant settings [296], although typically still require controlled time evolution. Furthermore, when combined with sophisticated classical signal processing techniques to efficiently extract information from the signal, these Loschmidt-echo-based methods can form the basis for so-called 'statistical phase estimation' methods [297–299].

### 2.4.3 Thermal properties

Quantities like the Loschmidt echo contain information not only about individual eigenvalues, but also other spectral properties. When supplied with an initial state with strong support on particular eigenstates, these methods can target the corresponding individual eigenstates; in contrast, I develop a related method in Chapter 4 that uses completely random initial states to compute the density of states (DOS) of Hamiltonians (and thus equilibrium thermal properties). Since

the DOS is a central object in statistical mechanics, estimating it is a key task in materials science, condensed matter physics and quantum chemistry. A broad range of classical techniques to estimate the DOS exist [300–303], but it is well known that computing it is a computationally hard problem in general [304]. Given the exponential computational resources required for quantum simulation in general, it is desirable to develop quantum algorithms for DOS estimation, for which fault-tolerant algorithms have been proposed [305]. Nonetheless, since quantum simulation of some systems seems feasible in the NISQ and early-fault-tolerant eras, it seems prudent to develop algorithms for DOS estimation that can be run on these devices. Early work in this direction has mainly involved Monte Carlo methods utilizing time evolution [306–308] (typically for the Loschmidt echo), or porting classical kernel polynomial methods [300] to quantum computers [309]. These methods typically require rather complicated initial state sampling, such as unitary 2- or 3-designs. Fortunately, these methods can be refined to require only trivial initial states (e.g. random bit-flips) with no reduction in performance, as I demonstrate in Chapter 4.

In addition, the DOS calculations in Refs. [306–309] were only demonstrated on simple spin models, and lack the ability to be restricted to specific subspaces of Hilbert space. Because thermal properties in the canonical and grand canonical ensembles depend upon the DOS on fixed-number subspaces, useful physical information about fermionic and bosonic systems cannot be extracted from these methods. Since the constituents of matter are fermions, almost all *ab initio* quantum models of genuine utility in applied fields like chemistry and materials science will have a fermionic component with a Fock space of variable particle number — therefore, useful applications of the above methods will require adapting them to subspaces of Hilbert space. Presently, computing the DOS of fermionic Hamiltonians on NISQ or early-fault-tolerant devices would require one to first compute Green's functions (from which the DOS can then be obtained). While analytic methods can successfully approximate these in the regime of perturbative interactions [310, 311], for strongly-correlated systems in general one would likely require quantum resources. Although variational approaches exist for this task [161–165], these

methods typically require explicit preparation of ground and/or excited states of the Hamiltonian, which is a highly non-trivial problem in general. In a variational setting, this is extremely dependent on the ability of the ansatz and optimizer to prepare these eigenstates, which as discussed in Section 2.2 is a challenging task. Similarly, proposed VQAs to explicitly prepare thermal states [312–314] are entirely dependent on expressibility and trainability. Basing near-term DOS calculations on simple time evolution is a clear alternative to circumvent these challenges for fermionic systems; the methods I develop in Chapter 4 overcome these challenges by adapting simple time-evolution-based methods [306–308] to subspaces of Hilbert space.

### 2.4.4 Excited states

Although QPE and related methods can be used to prepare excited states (assuming sufficient initial state overlap), the deep circuits involved render this impractical in the NISQ era. VQE, with its approach of *minimizing* the energy, cannot prepare excited states. Nonetheless, many VQAs have been proposed for excited states. One of the best-known approaches is variational quantum deflation [153, 154], wherein one solves a ground-state VQE problem, then projects out the (approximate) found ground state to prepare the eigenstate of next smallest value, and so on. This can be used to prepare low-lying eigenstates, but since each eigenstate depends on the approximation of the previous ones via the projection, the method increases in cost and decreases in fidelity for each successive eigenstate. Another method known as subspace-search VQE [155] attempts to find low-lying eigenstates by supplying orthogonal input states to the variational ansatz (since unitary operations preserve orthogonality), constructing a low-energy subspace of the Hamiltonian in which approximate eigenstates can be found. A particularly simple approach is a variational implementation [156] of the folded-spectrum method [315], whereby one instead variationally minimizes $\langle (H-\lambda)^2 \rangle$, where $\lambda$ is a hyperparameter that sets the target energy (i.e. the method should converge to the eigenenergy nearest to $\lambda$). Another distinct approach to targeting low-lying excitations adapts the venerable

'equations-of-motion method' from nuclear physics [316], based upon classical equations of motion for excitation operators, to the context of VQAs [157, 158].

Some variational methods do not necessarily target excited states, but rather converge to the 'closest' eigenstate. The variance-VQE algorithm [159] seeks to variationally minimize the energy variance $\langle H^2 \rangle - \langle H \rangle^2$, which vanishes if and only if the system is in an eigenstate of the Hamiltonian $H$. Similarly, the CoVaR method [152] uses root-finding methods to simultaneously minimize generic covariances $\langle HO \rangle - \langle H \rangle \langle O \rangle$ for random observables $O$; given an appropriate operator pool this is also a sufficient condition for an eigenstate. In both cases, if initialized sufficiently close in parameter space to an eigenstate (even if it is not the ground state), the methods may converge to it. In existing studies, it appears that CoVaR typically converges much faster than variance-VQE. More directly, CoVaR has also recently been used to intentionally prepare excited states by adiabatically deforming between trivial and target Hamiltonians [160].

### 2.4.5 Eigenvalue gaps

Although eigenvalue estimation is an undeniably high-value target problem, in some cases, one might instead only need to know the Hamiltonian eigenvalue differences, or gaps, of a system. In chemistry, the gaps of a Hamiltonian might reveal the photochemical properties of a molecule via the frequencies of its low-lying excitations [317]. In materials science, much can be learned about the elctronic properties of materials via their band gaps, and classical methods are inaccurate for systems with strong quantum correlations [318]. In adiabatic quantum computing, the timescales for annealing schedules must be determined based on low-lying gaps, which can generally be difficult to determine [319]. Clearly these are important quantities to compute, and in each of these cases, one requires only the eigenvalue *differences*, and not the eigenvalues themselves or the ability to explicitly prepare the eigenstates in quantum memory. This is a profoundly more permissive task, and one can potentially devise more efficient methods within these constraints.

Existing literature is somewhat limited to this end. While some approaches have proposed methods based on Fourier analysis of observable signals, such an approach requires good initial state overlap with the eigenstates in question and is highly dependent on the choice of measurement (see Section 3.1 for a more detailed explanation of this). Existing studies along these lines are often limited in their applicability due to these factors: some assume very specific (and generally hard) state preparation [320, 321], while others circumvent this via quantum annealing [322] (which is highly dependent on the minimum gap of the intermediate Hamiltonians). Another approach is the randomized gap estimation framework of Zintchenko and Wiebe [323], although this requires approximate Haar-random unitary operations, which can be challenging to realize. There is great potential to expand upon these methods, and therefore in Chapter 3 I introduce a NISQ-compatible method for estimating eigenvalue gaps that utilizes classical shadows (see Section 2.1) to overcome the issue of observable choice.

## 2.5 Classical shadows

### 2.5.1 Outline

Thus far in this chapter, it has been clear that one must weigh measurement resources just as seriously as computational (gate) requirements in quantum computing. In Section 2.3, I discussed how one can exactly cancel out errors with PEC at the cost of exponential measurement overhead; in Section 2.4, the tradeoff was clear in the difference between QPE (deep circuits, few measurements) and the Hadamard test (shallower circuits, many measurements). Circuit repetitions cost time, and the difference between a practical and impractical quantum protocol can often come down to frugality in measurement resources. Recently, a popular approach to efficient measurement has arisen in the form of *classical shadows*, introduced by Huang *et al.* [324] and refining upon earlier (but less practical) shadow tomography techniques by Aaronson [325].

The classical shadows protocol begins with the following observation. Suppose one has an $n$-qubit quantum state $\rho$ and applies a random unitary $U$ to it as

$\rho \to U^\dagger \rho U$. Measuring $\rho$ in the computational basis would yield an $n$-bit classical bitstring $\vec{b} \in \{0,1\}^n$, where I will use $|b\rangle$ to denote the corresponding computational basis state $|b\rangle = |b_1\rangle \otimes \cdots \otimes |b_n\rangle$ where $\vec{b} = [b_1 \cdots b_n]^T$. The average (over both the random unitary and the measurement outcomes) may, by linearity, be thought of as a 'measurement' channel $\mathcal{M}$, where

$$\mathcal{M}(\rho) = \mathbb{E}\left[U^\dagger |b\rangle \langle b| U\right]. \tag{2.14}$$

If the ensemble of unitary transformations $U$ is tomographically complete, then $\mathcal{M}$ is invertible (although $\mathcal{M}^{-1}$ is not physical, since it is not completely positive). Thus, one may write

$$\rho = \mathbb{E}\left[\mathcal{M}^{-1}\left(U^\dagger |b\rangle \langle b| U\right)\right]. \tag{2.15}$$

Although the presence of $\mathcal{M}^{-1}\left(U^\dagger |b\rangle \langle b| U\right)$ *inside* the average in Eq. (2.15) is simply an application of linearity, the implications of this are powerful. Suppose now that in addition to being drawn from a tomographically complete ensemble, the unitaries $U$ are also efficiently simulable on a classical computer for computational basis states (this is readily achieved for some ensembles, as I discussed in Section 2.1). Then the inverse-mapped (unphysical) snapshot $\mathcal{M}^{-1}\left(U^\dagger |b\rangle \langle b| U\right)$ may be computed efficiently on a classical computer. This defines the following sampling procedure:

1. Sample a random unitary $U_j$ from a tomographically complete ensemble that is classically simulable on computational basis states.

2. Construct the state $\rho$, apply the unitary as $\rho \to U_j^\dagger \rho U_j$, and measure the state in the computational basis to sample a bitstring $\vec{b}_1$.

3. Compute the snapshot $\rho_j = \mathcal{M}^{-1}\left(U_j^\dagger |b\rangle \langle b| U_j\right)$ using an efficient classical simulation method.

Repeating this procedure $N$ times yields a set of $N$ snapshots $S(\rho, N) = \{\rho_j\}_{j=1}^N$ known as a 'classical shadow', explicitly defined as

$$S(\rho; N) \equiv \left\{\mathcal{M}^{-1}\left(U_j^\dagger |b_j\rangle \langle b_j| U_j\right)\right\}_{j=1}^N. \tag{2.16}$$

In principle, one can then use these snapshots to estimate an expectation value $\mathrm{Tr}[O\rho]$ of an observable $O$ by averaging individual snapshot expectations $\mathrm{Tr}[O\rho_j]$, which can be computed classically. Crucially, the classical shadows protocol is independent of the observable $O$, and one may choose the observable to be estimated post hoc, *after* the measurement resources to construct the shadow have been expended. However, while observables will be faithfully reproduced in the limit $N \to \infty$, their convergence rate with respect to $N$ will depend on the ensemble of unitaries $U_j$ and the choice of $O$. I explore this further in the following section.

### 2.5.2 Convergence and unitary ensembles

The convergence rate of the classical shadows protocol is highly dependent on the observables being estimated and the choice of tomographically complete unitary ensemble. Huang *et al.* show that to estimate $M$ observables $\mathrm{Tr}[O_1\rho], \ldots, \mathrm{Tr}[O_M\rho]$ to additive error $\epsilon$, one requires a number of snapshots $N$ scaling as

$$N \geq \mathcal{O}\left(\log(M) \max_i \|O_i\|^2_{\mathrm{shadow}}/\epsilon^2\right), \tag{2.17}$$

where the definition of the 'shadow norm' $\|O_i\|_{\mathrm{shadow}}$ depends on the unitary ensemble. For example, in the case of random Pauli rotations (used throughout this thesis), the shadow norm is bounded as $\|O_i\|^2_{\mathrm{shadow}} \leq 4^q \|O_i\|^2_\infty$, where $q$ is the locality of the operator; this improves to $\|O_i\|^2_{\mathrm{shadow}} \leq 3^q \|O_i\|^2_\infty$ if $O_i$ is a tensor product of $k$ single-qubit observables (as is the case in all the applications I present in this thesis). If random Clifford measurements are used, the shadow norm depends on the Hilbert-Schmidt norm $\mathrm{Tr}[O_i^2]$. In the NISQ era, where circuit depth and connectivity is at a premium, this leads to a tradeoff. More sophisticated ensembles can result in a more powerful estimator (via the different shadow norm), but may be more difficult to implement on NISQ devices. For example, while the shadow norm for Clifford measurements is more favourable than for Pauli measurements, the latter requires only a depth-1 circuit of single-qubit gates, while the former requires a depth of $\mathcal{O}(n \log(n))$ on fully-connected topologies [326] (and possibly more on restricted topologies, which are common on NISQ devices). There is

therefore a pressing need for NISQ applications that make effective use of many local observables, which I address in Chapter 3.

Classical shadow protocols have drawn much attention in the last few years, leading to many further developments. The degree of freedom in choosing the tomographically complete unitary ensemble, for example, has led to variants tailored to different classical simulation schemes, each of which has different convergence properties that may be more suited to particular measurements. Some examples include refinements to the Clifford-based protocol [327] or interpolation between the Pauli and Clifford protocols [328], methods based on free-fermion techniques [329] and matchgates [330, 331], and tensor-network-based methods [332].

*The dark fire will not avail you, flame of Udûn. Go back to the Shadow!*

— Gandalf, *The Fellowship of the Ring* [333]

# 3

# Algorithmic shadow spectroscopy

*The research in this chapter was conducted collaboratively with Hans Chan, Richard Meister, and Bálint Koczor. Following a core idea and scope conceived by Bálint, the research team consisting of me, Richard and Hans performed analysis and jointly developed a full methodology. Each of us investigated the algorithm in a different era of quantum computing, with my unique contributions being the application to NISQ systems and study of scalability with tensor network methods. The project led to the published paper of Ref. [334]; in places where I use text verbatim from that manuscript, I am the original author of that text. The scope of this chapter is constrained to the overall methodology and my specific contributions.*

## Contents


**3.1 Methods . . . . . 46**
3.1.1 Classical shadows . . . . . 48
3.1.2 Signal postprocessing . . . . . 49
3.1.3 Summary and resource requirements . . . . . 54
3.1.4 Robustness to gate noise . . . . . 55
**3.2 Scalability . . . . . 57**
3.2.1 Analytic model . . . . . 58
3.2.2 Verifying scalability with tensor network simulations . . 60
**3.3 Application to near-term problems . . . . . 65**
3.3.1 Variational dynamics . . . . . 66
3.3.2 Demonstration of variational shadow spectroscopy . . . 67
**3.4 Conclusions . . . . . 70**

In Chapters 1 and 2, I described a range of cases where using a quantum computer to prepare eigenstates and compute eigenvalues of Hamiltonians could be useful. Placing further constraints on the problem, in Section 2.4.5 I considered the case where one is instead interested in the *eigenvalue differences* of a system. As I detail there, this could prove valuable for photochemistry [317], materials science [318], and quantum annealing [319]. Being a profoundly more permissive task, there is significant appeal in devising methods within these constraints that are more efficient, and therefore may have more plausible prospects in the NISQ era. However, existing literature on this particular problem is limited, leaving a compelling opportunity to develop new methods for eigenvalue gap estimation. In this chapter, I outline a resource-efficient method for estimating these gaps using only uncontrolled time evolution and few measurements on a quantum computer. In Section 3.1, I detail the novel method of *shadow spectroscopy*, which utilizes recent advances in efficient measurement techniques (classical shadows, Section 2.5), simple time evolution, and classical signal postprocessing to efficiently estimate eigenenergy differences. Following this, I study the scalability of the method in Section 3.2 using an analytic model and tensor network simulations. Taking the protocol to the NISQ era, in Section 3.3 I consider the use of variational time-evolution techniques to perform shadow spectroscopy, demonstrating its applicability with noisy simulations comparable to current quantum computers. Finally, in Section 3.4 I discuss the broader implications of this chapter.

## 3.1 Methods

Our approach to computing eigenenergy differences begins with the following simple observation. Consider some $d$-dimensional (typically $d = 2^n$ for $n$ qubits) Hamiltonian $H$ with eigenstates $H|\psi_j\rangle = E_j|\psi_j\rangle$, for which we wish to compute the low-lying

eigenenergy differences, and an observable $O$. Its expectation value evolves as

$$S(t) = \langle\psi(t)|\, O\, |\psi(t)\rangle = \sum_{jk} c_j^* c_k O_{jk} \exp(i(E_j - E_k)t), \tag{3.1}$$

where $c_j = \langle\psi_j|\psi(0)\rangle$ and $O_{jk} = \langle\psi_j|\, O\, |\psi_k\rangle$ are the coefficients of the initial state and matrix elements of the observable in the energy eigenbasis, respectively. We note that the observable oscillates in time, with a frequency determined by the energy differences $\Delta E_{jk} \equiv |E_j - E_k|$. In principle, one could therefore extract the eigenenergy differences by selecting an appropriate observable $O$, measuring $\langle O(t)\rangle$ at multiple (discrete) times $t$, and performing a Fourier analysis of the resulting time series. However, in practice this is not so straightforward. Since expectation values on quantum computers are estimated statistically using circuit repetitions (shots), resolving oscillations in $S(t)$ of amplitude $\mathcal{O}(\epsilon)$ requires $\mathcal{O}(1/\epsilon^2)$ shots. The amplitude of individual oscillations (corresponding to eigenenergy differences $\Delta E_{jk}$) in $S(t)$ is given by $|c_j|^2|c_k|^2|O_{jk}|$ (see Eq. 3.1), which may in principle be small or even zero. For randomly chosen states and observables, this amplitude will vanish exponentially in the system size, leading to an exponentially-poor scaling of the signal-to-noise ratio (SNR). While one may be able to heuristically choose observables on a case-by-case basis that yield better SNR, this is not a generalizable approach. Any method based on this observation must therefore account for two factors:

1. The initial state must have significant overlap with the relevant (low-lying) eigenstates of the Hamiltonian, such that $|c_j|^2|c_k|^2$ is not small.

2. The eigenbasis matrix elements $O_{jk}$ of the chosen observable(s) must be large for the relevant transitions.

The first factor is critical to most eigenvalue estimation methods, and ensuring good support in eigenstates is even relevant for fully fault-tolerant methods such as QPE [20]. I describe a NISQ-compatible approach to this in Section 3.3 (as well as alluding to other approaches in Section 3.4). The second factor has, to the best of my knowledge, not been addressed satisfactorily in the literature. In the remainder of Section 3.1 I outline a method that systematically addresses this issue

by using classical shadow techniques to efficiently estimate fluctuations in *all* local observables, and identify the statistically significant signals among these.

### 3.1.1 Classical shadows

As outlined above, in order to efficiently obtain a signal with good SNR that is characteristic of the eigenenergy differences, I first utilize classical shadows to estimate all $q$-local observables at a range of evolution times. I outlined the fundamentals of classical shadows in Section 2.5, and here I elaborate on the practical implementation for the purposes of shadow spectroscopy. Although essentially any variant of classical shadows may be useful for shadow spectroscopy (each variant giving efficient access to a different characteristic set of observables), many of these variants are not practical for NISQ devices. For example, while choosing $U$ to be random $n$-qubit Clifford operations in Equation 2.15 leads to a more powerful estimator, this requires $\mathcal{O}(n^2/\log(n))$ entangling gates to implement [324]. In contrast, the far simpler choice of $U$ being a random single-qubit Clifford rotation (that is, a random rotation into the $x$, $y$, or $z$ basis per qubit) requires only $n$ single-qubit gates with depth 1, making its resource cost trivial. While this constrains one to $q$-local observables, as shown in the remainder of this chapter, this is sufficient for the purposes of shadow spectroscopy, since strong fluctuations will be observed in *some* local observables (though we will not, and need not, know *a priori* which ones).

Therefore, within this chapter, I utilize this simpler form, and outline its practical implementation in this section. At every sample time $t$, a number of snapshots $N_S$ are sampled. Each of these snapshots is produced as follows. First, prepare the state $\rho = |\psi(t)\rangle \langle\psi(t)|$ on the quantum computer, using the chosen time-evolution method. Next, separately for each qubit $m$, randomly choose between the $x$, $y$ or $z$ bases with uniform probability. Rotate into the appropriate basis for measurement: if the $x$ basis is chosen, apply a Hadamard gate; if the $y$ basis is chosen, apply a Hadamard gate followed by an $S^\dagger$ gate; if the $z$ basis is chosen, do nothing. Measure the state in the computational basis, and record the outcome (following the rotation, this is equivalent to measuring $\sigma_m^x$, $\sigma_m^y$, or $\sigma_m^z$, respectively). Store the measured

bitstrings in a vector $\vec{b}_j \in \{0,1\}^n$, where $j \in \{1,\ldots,N_S\}$ indexes the snapshots, and record the choice of measurement basis in a vector $\vec{\mathcal{P}}_j \in \{x,y,z\}^n$. This information constitutes the classical shadow of Equation 2.16. After collecting the shadow, one may wish to estimate an expectation value of a $q$-local Pauli string $O = \prod_{m\in\mathcal{I}} \sigma_m^{\nu_m}$ where $\nu_m \in \{x,y,z\}$ and $\mathcal{I}$ is a set of $\bar{q} \leq q$ qubit indices. Although the inverse mapping of the measurement basis rotation can trivially be performed on a classical computer, for Pauli shadows there is an even simpler probabilistic interpretation. If a measurement basis not matching the $\bar{q}$ indices in $\mathcal{I}$ is chosen, then

$$\mathcal{M}^{-1}\left(U_j^\dagger \left|b_j\right\rangle \left\langle b_j\right| U_j\right) = 0, \tag{3.2}$$

and therefore the contribution to the expectation value is zero. Consequently, one needs only count the instances that the measurement basis matches the subset of $q$ indices in the Pauli string, and the expectation value is estimated as

$$\langle O\rangle \approx \frac{3^{\bar{q}}}{N_S}\sum_{j=1}^{N_S}\prod_{m\in\mathcal{I}}\left(\vec{b}_j\right)_m. \tag{3.3}$$

Here, one counts the number of times the measurement basis actually (by chance) matches the Pauli string on the $\bar{q}$ indices of interest, and divides by the expected number of times this should occur in $N_S$ snapshots. As noted in Section 2.5.2, this has merely a logarithmic sample complexity, and is therefore an extremely efficient method for estimating a large quantity of observables. In the section that follows, I detail the signal processing techniques that can be used to exploit the redundancy of this large dataset to efficiently extract eigenenergy differences.

### 3.1.2 Signal postprocessing

By simulating time evolution according to the system's Hamiltonian on a quantum computer, one can prepare the state $|\psi(t)\rangle$ at any desired time $t$. Typically, one would prepare the state at a series of $N_T$ discrete times $t = n_t\Delta t$, where $n_t \in \{1,\ldots,N_T\}$ indexes these timesteps. Utilizing the shadow tomography techniques outlined in Section 3.1.1, one can therefore efficiently obtain a set of $N_o$ temporal signals consisting of expectation values of $q$-local Pauli observables $P^{(k)}$, denoted as

$$S_k(n_T) = \left\langle\psi(n_T\Delta t)\right| P^{(k)} \left|\psi(n_T\Delta t)\right\rangle + \mathcal{E}_k(n_T), \tag{3.4}$$

where $\mathcal{E}_k(n_T)$ is the shot noise error in the estimate of the $k$-th observable at each timestep $n_T$. Each signal contains information about the eigenenergy differences (3.1), and naively, one might expect that a simple averaging of signals $\sum_k S_k(n_T\Delta t)$ would yield a more resolvable combined signal. However, noting the functional form of Eq. 3.1, one can see that the oscillations of the individual signals will not be in phase, since

$$S_k(n_T) = \sum_{jl} |c_j^* c_l P_{jl}^{(k)}| \exp\left(i\left((E_j - E_l)n_T\Delta t + \phi_{kjl}\right)\right) + \mathcal{E}_k(n_T), \tag{3.5}$$

where $\phi_{kjl} = \arg(c_j^* c_l P_{jl}^{(k)})$ is an observable-dependent (and initial state-dependent) phase, depending on the phase of the matrix elements $P_{jl}^{(k)} = \langle\psi_j| P^{(k)} |\psi_l\rangle$ of the corresponding Pauli observable. Since these phases will differ between observables, a naive averaging of signals will sum over oscillations that are not in phase, leading to destructive interference and a reduction in the SNR. A successful approach must therefore combine the signals in a way that identifies the strongest shared frequency components of the signals, while accounting for the fact that they will be relatively out of phase. Done correctly, this will suppress the shot noise in the individual signals, since the noise is uncorrelated between observables, while the oscilliatory signal is correlated (albeit possibly out of phase). This allows the individual (possibly very noisy) signals to be combined in a way that leverages the redundancy in the data to strongly suppress the noise, making very effective use of the large number of local observables that can be simultaneously estimated with a single classical shadow. In the remainder of this section, I outline the postprocessing used to extract eigenenergy differences from these signals. First, the temporal signals are standardized such that they have zero mean and unit variance, as

$$f_k(n_T) = \frac{S_k(n_T) - \mu_j}{\sigma_k}, \tag{3.6}$$

where $\mu_j$ and $\sigma_j$ are the sample mean and standard deviation of $j$-th signal, respectively, given by

$$\mu_k = \frac{1}{N_t} \sum_{n_T} S_k(n_T), \tag{3.7}$$

$$\sigma_k = \sqrt{\frac{1}{N_T} \sum_{n_T} \left(S_k(n_T) - \mu_j\right)^2}. \tag{3.8}$$

One would expect *a priori* that many of these signals would be overwhelmed by shot noise due to small $O_{nm}$ for the $n, m$ corresponding to states with good support in the initial state. Consequently, the next step is to filter out signals $f_k(n_T)$ that are statistically indistinguishable from noise. Various options exist, but in this chapter I use the Ljung-Box test [335, 336] for signal rejection. This is a statistical test that determines whether a time series has a statistically significant nonzero autocorrelation. Since the temporal signals are expected to be temporally correlated due to the oscillations in Eq. 3.1, and the shot noise in the dynamics is inherently Markovian, the Ljung-Box test can be used to identify signals that are likely to contain a significant signal. Its output for each signal is a $p$-value, which is the probability that the observed autocorrelations are due to random chance; the threshold $p$-value above which to reject signals is a hyperparameter for this protocol. This step is highly parallelizable: both the signal generation from the classical shadow $S(\rho; N)$ and the Ljung-Box test can be completely parallelized into $N_o$ separate instances, each of complexity $\mathcal{O}(N_T)$.

Next, the $N_o'$ remaining signals are combined into a 'data matrix' $\boldsymbol{D} \in \mathbb{R}^{N_o' \times N_T}$, whose rows are the remaining standardized signals $f_k(n_T)$ and columns index the timesteps $n_T$, such that its matrix elements are

$$(\boldsymbol{D})_{lm} = f_l(m). \tag{3.9}$$

Although the individual signals are statistically likely to contain a resolvable signal, on an individual basis they will in practice be burdened by a large amount of shot noise. Next, one may wish to identify a temporal signal $v(n_T)$ that is characteristic

of the oscillations across the entire set of signals. Denoting the $L^2$ inner product between it and a signal $f_k(n_T)$ as

$$\langle v, f_k \rangle = \sum_{n_T} v(n_T) f_k(n_T), \tag{3.10}$$

one may wish to find a signal $v(n_T)$ that maximizes the average overlap with all signals, such that

$$v = \underset{\|v\|_2=1}{\arg\max} \left( \frac{1}{N_o'} \sum_{k=1}^{N_o'} |\langle v, f_k \rangle|^2 \right). \tag{3.11}$$

Manipulating the maximand of Eq. 3.11, one finds that

$$\frac{1}{N_o'} \sum_{k=1}^{N_o'} |\langle v, f_k \rangle|^2 = \frac{1}{N_o'} \sum_{k=1}^{N_o'} \left( \sum_{n_T}^{N_T} v(n_T) f_k(n_T) \sum_{m_T}^{N_T} v(m_T) f_k(m_T) \right) \tag{3.12}$$

$$= \sum_{n_T, m_T}^{N_T} v(n_T) \underbrace{\left( \frac{1}{N_o'} \sum_{k=1}^{N_o'} f_k(n_T) f_k(m_T) \right)}_{(\boldsymbol{C})_{n_T m_T} = (\boldsymbol{D}^T \boldsymbol{D}/N_o')_{n_T m_T}} v(m_T) \tag{3.13}$$

$$= v^T \boldsymbol{C} v, \tag{3.14}$$

where we have used the fact that the signals are real, and defined the square matrix $\boldsymbol{C} \in \mathbb{R}^{N_T \times N_T}$, given by $\boldsymbol{C} \equiv \boldsymbol{D}^T \boldsymbol{D}/N_o'$. Consequently, the dominant eigenvector of $\boldsymbol{C}$ is the signal $v(n_T)$ that maximizes the average overlap with all signals (3.11). Notably, since the dimension of $\boldsymbol{C}$ is indepdendent of the number of signals $N_o'$, this step represents a reduction in dimensionality. This procedure is essentially equivalent to a principal component analysis (PCA) of the signals, and in PCA terminology, $\boldsymbol{C}$ is the 'covariance matrix', and $v(n_T)$ is the 'first principal component' of the signals, which accounts for the largest amount of variance in the data. Stopping here and performing a Fourier analysis of $v(n_T)$ can in practice yield a quality spectrum, but one can improve upon this by incorporating higher-order principal components of the data. In practice, one chooses some cutoff $c$ and incorporates the first $c$ principal components $v_1, \ldots, v_c$ — in the results presented in this chapter, I typically use $c = 4$. This can be most usefully incorporated by performing a

spectral cross-correlation analysis. The cross-correlation of two principal component signals $v_k(n_T)$ and $v_l(n_T)$ is given by the convolution

$$X_{kl}(m_T) = \sum_{n_T=1}^{N_T-m_T-1} v_k(n_T + m_T)v_l(n_T). \tag{3.15}$$

One then discrete Fourier transforms each cross-correlation, obtaining a frequency-dependent cross-correlation matrix

$$[\boldsymbol{X}(n_\omega)]_{kl} = \mathcal{F}[X_{kl}](n_\omega), \tag{3.16}$$

where $n_\omega$ indexes the Fourier frequencies conjugate to the evolution times $n_T\Delta t$. If the signals were infinitely long and free of shot noise, then these matrix elements would simply be products of the individual Fourier transforms, as

$$[\boldsymbol{X}(n_\omega)]_{kl} \propto [\mathcal{F}[v_k](n_\omega)]^* \times \mathcal{F}[v_l](n_\omega), \tag{3.17}$$

and consequently the matrices $\boldsymbol{X}(n_\omega)$ would be rank-one, with a single nonzero singular value

$$\lambda(n_\omega) = \sum_k |\mathcal{F}[v_k](n_\omega)|^2, \tag{3.18}$$

giving the summed spectral density of the principal components. Since the signals will always be finite and noisy, a singular value decomposition (SVD) of the matrix will instead allow one to estimate the ideal noise-free spectral density. For the final postprocessing step, one constructs the signal

$$I(n_\omega) = \lambda_1(n_\omega), \tag{3.19}$$

where $\lambda_1(n_\omega)$ is the largest singular value of $\boldsymbol{X}(n_\omega)$. This yields a frequency-dependent signal with pronounced peaks at the eigenvalue differences for eigenstates with good initial state support, incorporating the redundant information from the many local signals that are cheaply extracted from a single classical shadow.

### 3.1.3 Summary and resource requirements

With the individual steps of algorithmic shadow spectroscopy now outlined in detail, I present a brief summary of the steps involved, as well as their resource complexities.

1. **Measure observables:** Using the classical shadows protocol outlined in Section 3.1.1, estimate the expectation values of up-to-$q$-local Pauli strings at $N_T$ timesteps. For a fixed error scale, this has sample complexity $\mathcal{O}(N_T \log(N_o))$, where $N_o \in \text{poly}(n)$.

2. **Filter significant signals:** Using a Ljung-Box test, reject signals that are statistically indistinguishable from noise. Although this has a classical complexity of $\mathcal{O}(N_o N_T)$, the tests are parallelizable into $N_o$ separate instances of $\mathcal{O}(N_T)$ complexity.

3. **Assemble data matrix:** Arrange the $N_o'$ remaining signals into a data matrix $\boldsymbol{D}$, which has a complexity of $\mathcal{O}(N_o' N_T)$.

4. **Dimensionality reduction:** Compute the data matrix $\boldsymbol{C} = \boldsymbol{D}^T \boldsymbol{D}/N_o'$, which has a complexity of $\mathcal{O}(N_T^2 N_o')$ due to rectangular matrix multiplication. This is the determining step for the overall asymtpotic complexity, as it is the bottleneck for classical processing.

5. **Principal component analysis:** Compute the eigenvalues and eigenvectors of $\boldsymbol{C}$, which has a complexity of $\mathcal{O}(N_T^3)$ using standard algorithms.[1]

6. **Fourier transform:** Compute the discrete Fourier transforms of the $c$ most dominant eigenvectors, which has a complexity of $\mathcal{O}(cN_T \log(N_T))$ using the fast Fourier transform [338] (parallelizable into $c$ instances of complexity $\mathcal{O}(N_T \log(N_T))$).

[1] In theory, the complexity is $\mathcal{O}(N_T^\omega)$, where $\omega \leq 2.371552$ is the matrix multiplication constant [337]. In practice, due to constant overheads in asymptotically-optimal matrix multiplication algorithms, the complexity is $\mathcal{O}(N_T^3)$.

7. **Spectral cross-correlation:** Construct (and perform an SVD of) the spectral cross-correlation matrix at each of the $N_T$ frequencies. This has a complexity of $\mathcal{O}(N_T c^3)$, parallelizable into $N_T$ instances of complexity $\mathcal{O}(c^3)$.

In Figure 3.1(a), this process is summarized visually in a flowchart. I also depict an example of the mathematical objects obtained during the procedure for a 6-qubit disordered Heisenberg chain (3.33), showing the data matrix $\boldsymbol{D}$ (Figure 3.1(b)), covariance matrix $\boldsymbol{C}$ (Figure 3.1(c)), and resulting spectrum $I(E)$ (Figure 3.1(d)).

### 3.1.4 Robustness to gate noise

In practice, shadow spectroscopy is extremely robust to gate noise. First, recall that most archetypal incoherent noise models (e.g. depolarizing, bit-flip, dephasing) admit a probabilistic interpretation [339]. Given an intended unitary gate operation $U_j$ (the $j$-th gate in some circuit), its noisy counterpart $\Phi_j(\rho)$ with fault probability $p$ consists of the ideal operation $U$ with probability $1-p$, and some erroneous operation $\Phi_{\text{err}}$ with probability $p$ as

$$\Phi_j(\rho) = (1-p)U_j\rho U_j^\dagger + p\Phi_{\text{err}}(U_j\rho U_j^\dagger). \tag{3.20}$$

In a $\nu$-gate quantum circuit, the channel of the noisy circuit then admits the probabilistic form

$$\Phi_\nu \circ \cdots \circ \Phi_1(\rho) = \eta\rho_{\text{ideal}} + (1-\eta)\rho_{\text{err}}, \tag{3.21}$$

where $\eta = (1-p)^\nu$ is the probability that all gates executed correctly, $\rho_{\text{err}}$ is an error state that is composed of a mixture of all error events in the circuit, and $\rho_{\text{ideal}} = U_\nu \dots U_1 \rho U_1^\dagger \dots U_\nu^\dagger$ is the ideal output. A vast range of error models can be transformed to admit this representation via e.g. Pauli twirling procedures [340, 341], and so this is generally a reasonable assumption. Per the below theorem, this results in a convenient form for the time-dependent signals utilized in shadow spectroscopy.

**Theorem 1.** *Suppose that the series of time-evolved states $\rho(t_n)$ used to compute time-dependent observable signals are prepared via noisy quantum circuits admitting*

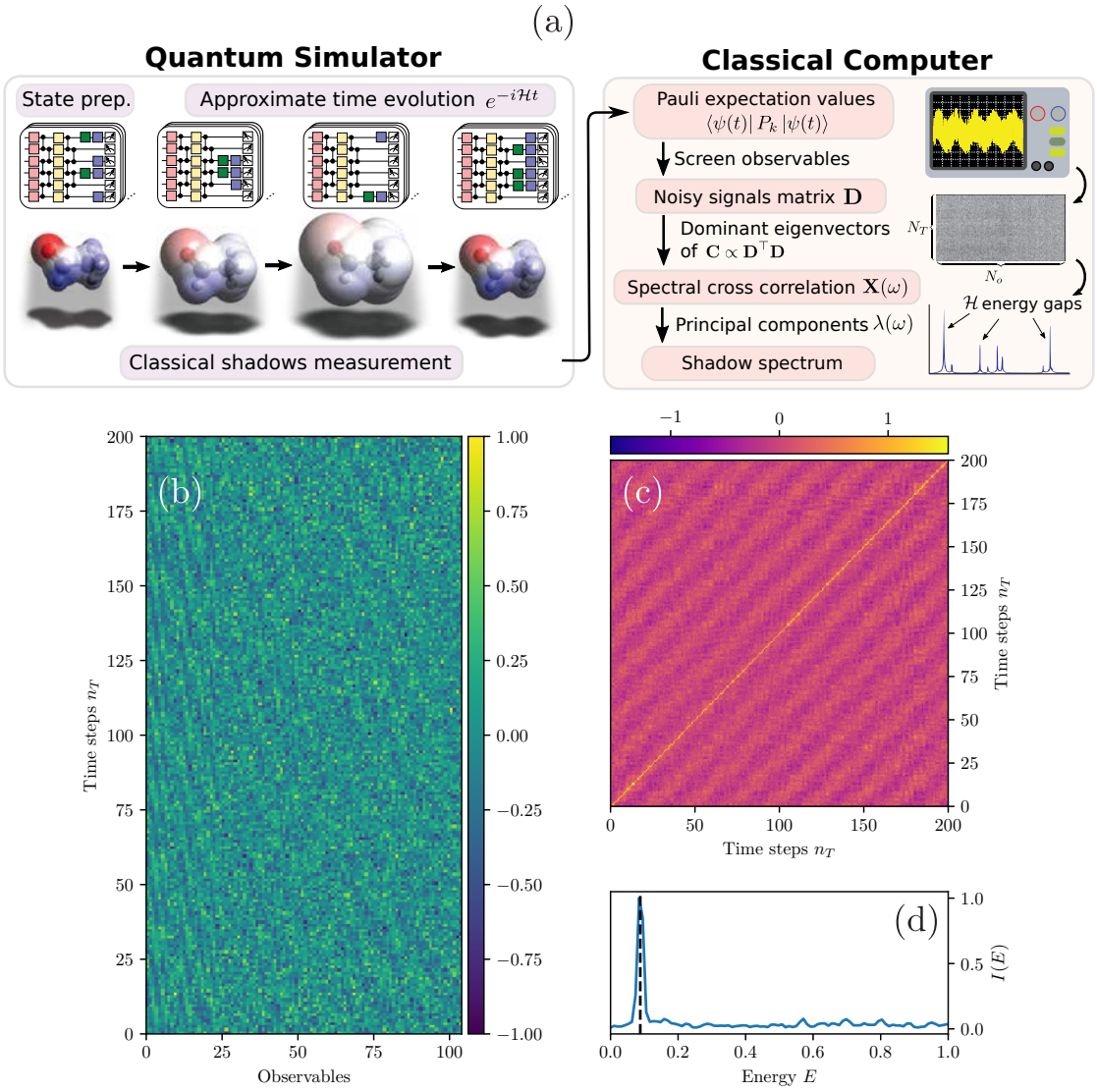



**Figure 3.1: Walkthrough of algorithmic shadow spectroscopy.** **(a)** A flowchart depicting the shadow spectroscopy process. Having prepared an initial state with good support in the relevant eigenstates, one applies uncontrolled time evolution, measuring a classical shadow at each sampled time $t$. These shadows, consisting of classical data, are then used on a classical computer to estimate the expectation values of all local observables at each time. These are filtered by statistical significance, composed into a data matrix, and a principal component analysis is performed. Finally, a spectral cross-correlation yields the shadow spectrum, whose peaks correspond to the desired gaps. **(b)** The data matrix $\boldsymbol{D}$ for a 6-qubit disordered Heisenberg chain (3.33), with $n_T = 200$ timesteps and signals with $p < 0.02$ under the Ljung-Box test retained. Here, $N_o' < N_T$ since the system is small, but for larger systems one may expect $N_o' \gg N_T$. **(c)** The covariance matrix $\boldsymbol{C}$. Note that this has dimension $N_T \times N_T$, and will be the same size regardless of the number of signals used. **(d)** The resulting shadow spectrum $I(E)$, with one peak corresponding to the gap between the ground state and first excited state of the system.

*the form in Equation 3.21. Then the noisy time-dependent signals (expected values of Pauli strings $P_k$) will admit the form*

$$S_k^{(noisy)}(n_T) = \eta S_k(n_T) + (1-\eta)W_k(n_T), \tag{3.22}$$

*where $W_k(n_T) \equiv \mathrm{Tr}[\rho_{err}(n_T)P_k]$ is an error signal, and $\eta$ depends on circuit depth (i.e. it will be a constant for the variational approach of Section 3.3, and $n_T$-dependent for most other methods). Furthermore, the error signal can be bounded as*

$$W_k(n_T) \leq \|\rho_{err}(n_T) - \mathbb{1}/d\|_1. \tag{3.23}$$

Theorem 1 is proven by my collaborator Bálint in Appendix C of our full manuscript [334]. Since the Fourier transform is linear, Equation 3.22 guarantees that the shadow spectrum will be a linear combination of the ideal spectrum and some additive artefacts, which are bounded in size based on how close the error model is to totally white noise (3.23). Crucially, these additive artefacts *do not shift the peaks of the spectrum.* Furthermore, so long as the circuit error rate is relatively low, Theorem 1 also implies that the peak intensity will be at worst slightly shrunk, with no broadening of the peak line shape. One can therefore expect that shadow spectroscopy will still robustly produce accurate peaks, even in the presence of mild to moderate hardware noise. In Section 3.3.2, I corroborate these claims of noise robustness with numerical evidence.

## 3.2 Scalability

For a quantum algorithm to be truly useful, it must be scalable. In this section, I consider the scalability of shadow spectroscopy from both an analytic and numerical perspective.

### 3.2.1 Analytic model

I first note that two metrics for scalability should be considered: the spectral resolution, and the SNR of the peaks. With regard to the former, shadow spectroscopy achieves Heisenberg-limited scaling: that is, to resolve an energy gap $\Delta$, a total evolution time of $T \sim 1/\Delta$ is required. Consequently, the shot requirements will also scale as $\sim 1/\Delta$. Although this is superior to the shot-noise-limited scaling achieved by methods such as VQE, it may not be efficient for very small gaps. In strongly-correlated systems, low-lying eigenstates may concentrate exponentially close. In such cases, the resource scaling to resolve these near-degenerate gaps will be exponential, and therefore not efficient. Nonetheless, resolving these gaps is still exponentially difficult for other methods, even for QPE [20, 21]. I cautiously note that shadow spectroscopy will only prove efficient for gapped systems where the gap closes at worst polynomially in the system size.

The situation is far more promising with regard to the scaling of SNR. Both analytic evidence (this section) and numerical evidence (Section 3.2.2) suggest that the SNR will *improve* with system size due to the redundancy in the data, even at fixed total shot budget. First, consider the mean squared spectrum (MSS) given by

$$\mathrm{MSS}\left[\omega_{n_T}\right] = \frac{1}{N_o}\sum_{k=1}^{N_o} |F_k(n_T)|^2, \tag{3.24}$$

where the $F_k(n) \equiv \mathcal{F}[f_k](n_\omega)$ are the Fourier transforms of the standardized signals. This is a simpler quantity than the full spectrum $I(n_\omega)$ obtained from the postprocessing procedures of Section 3.1.2, lending itself more readily to simple analytic study. Nonetheless, it produces a spectrum characteristic of the energy eigenvalue differences, and may be used as a worst-case proxy, since the full postprocessing strategy will only improve upon this. From Equation 3.1, one can infer that the individual signals $f_k(n_T)$ will be a sum of sinusoids of the form $I_k \cos(\omega_{\mathrm{peak}} \Delta t n_T + \phi_k)$, where $\omega_{\mathrm{peak}}$ is the relevant eigenenergy difference and $\phi_k$ is a transition-dependent, observable-dependent phase shift. The peak intensity will therefore be given by

$$\mathrm{MSS}\left[\omega_{\mathrm{peak}}\right] = C N_T \overline{I^2}, \tag{3.25}$$

where $C$ is some $\mathcal{O}(1)$ constant that may depend on the choice of Fourier window, and $\overline{I^2} \equiv \frac{1}{N_o}\sum_k I_k^2$ is the average intensity of the peaks. However, in practice the signal is burdened by shot noise, which for a modest number of shots can already be well approximated by a normally distributed random variable. Since the Fourier transform of normally distributed noise is also normally distributed, each point in the individual frequency-spaced signals may be written as $F_k(n) + \mathcal{E}_k(n)$. In this case, the expected peak intensity will be given by

$$\mathbb{E}\left(\mathrm{MSS}\left[\omega_{\mathrm{peak}}\right]\right) = \frac{1}{N_o}\sum_{k=1}^{N_o}\mathbb{E}\left[\left(CN_T I_k + \mathcal{E}_k(\omega_{\mathrm{peak}})\right)^2\right] \tag{3.26}$$

$$= CN_T\overline{I^2} + \epsilon^2, \tag{3.27}$$

where $\mathrm{Var}[\mathcal{E}_n(n)] = \epsilon^2$. Far away from the peak centre at some $\omega_{\mathrm{base}}$, where the ideal signal is near zero, one encounters a constant expected noise base

$$\mathbb{E}\left(\mathrm{MSS}\left[\omega_{\mathrm{base}}\right]\right) = \frac{1}{N_o}\sum_{k=1}^{N_o}\mathbb{E}\left[\mathcal{E}_k(\omega_{\mathrm{base}})^2\right] = \epsilon^2. \tag{3.28}$$

Given this (expected) constant baseline, the SNR is determined by its variance, given by

$$\mathrm{Var}\left(\mathrm{MSS}\left[\omega_{\mathrm{peak}}\right]\right) = \frac{1}{N_o^2}\sum_{k=1}^{N_o}\mathrm{Var}\left[\mathcal{E}_k(\omega_{\mathrm{base}})^2\right] = \frac{2\epsilon^4}{N_o}. \tag{3.29}$$

Combining these yields an SNR

$$\mathrm{SNR} \propto N_T N_S \overline{I^2}\sqrt{N_o}, \tag{3.30}$$

where I have assumed a standard shot noise scaling $\epsilon = \mathcal{O}(1/\sqrt{N_S})$. As the system size $n$ increases, the number of available observables $N_o$ will increase polynomially at fixed shot noise $\epsilon$ for only a logarithmic increase in $N_S$ [324]. Although $\overline{I^2}$ can be strongly system-dependent, so long as the average $q$-local observable intensity does not shrink asympotically faster than $\sqrt{N_o}$, this suggests the SNR will improve with system size. My collaborator Bálint affirmed this numerically at small system sizes using exact diagonalization [334]; in the section that follows, I verify this at much larger system sizes using tensor network simulations.

### 3.2.2 Verifying scalability with tensor network simulations

While the simple analytic study of Section 3.2.1 suggests that the SNR of shadow spectroscopy should improve with system size, it is important to verify this numerically. However, as discussed in Section 2.1, exact numerical simulation of quantum circuits scales exponentially in $n$ for full-state methods, and is impractical beyond a few tens of qubits. My collaborator's initial simulations using full-state methods hinted at good scaling, but were limited to $n \leq 14$, and ideally the scaling behaviour at system sizes well beyond the reach of exact classical methods should be assessed. For this purpose, I simulate shadow spectroscopy at scale, using MPS methods — namely the time-evolving block-decimation (TEBD) algorithm [114]. A full outline of tensor network methods is well beyond the scope of this thesis — the interested reader may refer to a number of introductory resources [114–116], and I provide a very brief introduction to MPS methods here. Note that an $n$-qubit quantum state $|\psi\rangle$, whose amplitudes can be written as a rank-$n$ tensor $\psi^{\sigma_1 \dots \sigma_n}$, can be written as a matrix product state (MPS)

$$\psi^{\sigma_1 \dots \sigma_n} = \sum_{\alpha_1 \dots \alpha_{n-1}} A^{\sigma_1}_{\alpha_1} A^{\sigma_2}_{\alpha_1 \alpha_2} A^{\sigma_3}_{\alpha_2 \alpha_3} \dots A^{\sigma_{n-1}}_{\alpha_{n-2} \alpha_{n-1}} A^{\sigma_n}_{\alpha_{n-1}}, \tag{3.31}$$

where the full-state tensor has been decomposed into a product of $n-2$ rank-3 tensors $A^{\sigma_i}_{\alpha_{i-1}\alpha_i}$, and 2 rank-2 tensors $A^{\sigma_1}_{\alpha_1}$ and $A^{\sigma_n}_{\alpha_{n-1}}$. This product is depicted in tensor diagram notation in Figure 3.2(a). Note that this is a fully general decomposition; while the physical indices $\sigma_i \in \{0, 1\}$ have dimension 2, the bond indices $\alpha_i \in \{1, \dots, D_i\}$ can in general have exponential dimension $D_i$ (their maximum dimension grows exponentially in distance from the edge of the chain). Because of this, the largest of these dimensions $\chi \equiv \max_i D_i$ is known as the *bond dimension*, and determines the size in memory of the MPS. The bond dimension $\chi$ is closely linked with the level of entanglement; indeed, an unentangled product state will have $\chi = 1$. Weakly entangled states can be approximated by efficient MPS representations by discarding the least relevant components and truncating

the bond dimension. Any bond can be written in a useful form using a singular value decomposition (SVD) of the tensor, since

$$A^{\sigma_j}_{\alpha_{j-1}\alpha_j} A^{\sigma_{j+1}}_{\alpha_j\alpha_{j+1}} = \sum_\beta U^{\sigma_j}_{\alpha_{j-1}\beta} \lambda_\beta V^{\sigma_{j+1}}_{\beta\alpha_{j+1}}, \tag{3.32}$$

depicted diagramatically in Figure 3.2(b). The $\lambda_\beta \geq 0$ are singular values, and one can 'compress' an MPS by discarding the smallest singular values beneath some cutoff $\epsilon_{\text{SVD}}$ (or alternatively an absolute truncation of the bond dimension $\chi$, though this is an uncontrolled approximation). Furthermore, note that single-qubit gates do not increase bond dimension, since they act on only one physical index $\sigma_j$ and thus can be absorbed into the corresponding tensor $A^{\sigma_j}_{\alpha_{j-1}\alpha_j}$. Similarly, local entangling gates will increase the bond dimension only locally, depicted diagramatically in Figure 3.2(c). In the TEBD algorithm, the state is stored as a MPS, and time evolution under a local Hamiltonian is Trotterized into a series of local gates. When entangling gates are applied, the bond dimension will grow (e.g. it will double for 2-qubit gates), but one then applies MPS compression up to some tolerance $\epsilon_{\text{SVD}}$, leading to slow growth of $\chi$ so long as the dynamics are not too strongly entangling.

Using these methods, I am able to simulate large system sizes well beyond the capabilities of exact diagonalization to confirm that our approach is scalable (as long as a sufficiently good initial state can be provided), by verifying that an increasing system size improves the performance of our approach. As an example system, I choose the Heisenberg chain, whose Hamiltonian is

$$\mathcal{H} = -J \sum_{j=1}^{n-1} \vec{\sigma}_j \cdot \vec{\sigma}_{j+1} + \sum_j^n h_j \sigma_j^z. \tag{3.33}$$

Here $\vec{\sigma}_j = [\sigma_j^x, \sigma_j^y, \sigma_j^z]$ is a vector of Pauli $x$, $y$ and $z$ matrices on the $j$th qubit, $J$ is a coupling constant, and $h_j \in [-h, h]$ is sampled from a uniform distribution (for disorder strength $h$). This Hamiltonian is relevant in studies of many-body localization and self-thermalization [342, 343], but its phase transitions remain poorly understood due to the difficulty of solving the model on a classical computer at larger coupling $J$. Therefore, efficiently probing the spectral gap of this model with shadow spectroscopy may help enhance understanding of key many-body quantum

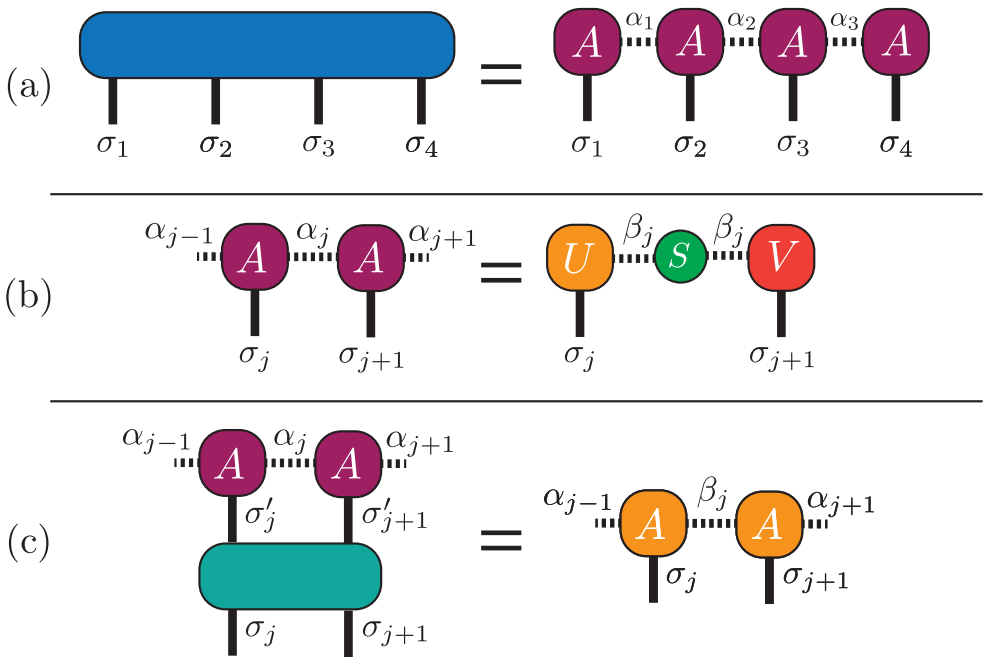



**Figure 3.2: Diagrammatic representation of MPS methods.** These diagrams depict tensor-network equalities from Section 3.2.2 in standard tensor diagram form: boxes depict individual tensors, internal lines joining tensors indicate indices that are summed over, and external lines indicate free indices in the resulting composite tensor. Physical indices are represented by solid lines, and bond indices by dashed lines. **(a)** Representation of Equation 3.31. A general $n$-qubit quantum state represented by tensor $\psi^{\sigma_1 \cdots \sigma_n}$ can in general be decomposed into a product of $n-2$ rank-3 tensors and 2 rank-2 tensors, each featuring one physical index $\sigma_i$ and internal 'bond' indices $\alpha_i$. **(b)** Representation of Equation 3.32. The product of two tensors in the MPS representation can be rewritten via the SVD as a a product of three tensors, the middle of which is a diagonal matrix of singular values (hence the repeated index $\beta_j$ in the diagram). Note that $\beta_j \leq \alpha_j$, since some of the singular values may be exactly zero; in practice, one may compress by discarding small singular values. **(c)** Representation of applying a (2-qubit) local entangling gate to the relevant tensors in a MPS. The bond dimension may grow by a factor of 2 such that $\beta_j = 2\alpha_j$, but can be compressed down with an SVD if the dynamics are not too entangling.

phenomena. In particular, for NISQ systems, this system has been identified as a key model for achieving early quantum advantage due to its combination of hardware-friendly definition and non-trivial physics [344]. This will prove of particular relevance in Section 3.3, where I will utilize it to demonstrate a NISQ-friendly variant of shadow spectroscopy.

In the remainder of this section, I consider the disordered Heisenberg spin chain with random site-dependent magnetic fields (Equation 3.33) at a relatively small

coupling $J = 0.01$ ($h = 1$). In this limit, the system can straightforwardly be initialized with a good initial state in the following way. In the limit $J \to 0$, the Hamiltonian is analytically solvable (as it has only on-site terms), with a state given by a bitstring $|b\rangle$. Here the $j$-th bit $b_j$ of the bitstring $|b\rangle$ is obtained as

$$b_j = \begin{cases} 0 & h_j < 0 \\ 1 & h_j \geq 0 \end{cases}. \tag{3.34}$$

The first excited state can then be obtained by flipping the $l$-th bit in this bitstring, where $l$ corresponds to the absolute smallest on-site energy term $l = \arg\min_k |h_k|$. Thus, an equal superposition of the first and excited states is simply created by applying a Hadamard gate to the ground state as $H_l |b\rangle$. Given that the bond dimension grows exponentially with $JT$, where $T$ is the total simulation time, choosing a significantly larger coupling $J$ would push this into a regime where quantum computers can outperform classical computational techniques, and beyond the realm of tractable tensor network simulation. Furthermore, since the $J = 0$ system is analytically solvable, the correctness of the spectral peaks can be verified at beyond-exact-diagonalization system sizes by standard (Rayleigh-Schrödinger) perturbation theory. In summary, the choice of a weak coupling $J$ is intentional, and facilitates this analysis, since (a) a good approximate initial state can be chosen, (b) the state accumulates only modest bond dimension, and (c) veracity of peak positions can be verified perturbatively.

In each simulation I initialize the initial state vector $H_q |b\rangle$ as an MPS with bond dimension $\chi = 1$ (i.e., a product state), and evolve it with the TEBD algorithm [114]. I perform this evolution using the TEBD implementation in the QUIMB Python package, using a Trotter step size

$$\Delta t = \left( \frac{\epsilon_{\text{Trotter}}}{T \|\mathcal{H}\|_F} \right)^{1/k}, \tag{3.35}$$

where $\epsilon_{\text{Trotter}}$ is an error tolerance, $T$ is the total evolution time, a $k$-th order Trotter decomposition is used, and $\|\mathcal{H}\|_F$ is the Frobenius norm of the Hamiltonian (equivalent to the sum of squares of its Pauli coefficients). I set a desired precision $\epsilon_{\text{Trotter}} = 10^{-3}$ and use $k = 4$. At each time step, I compress the state by performing

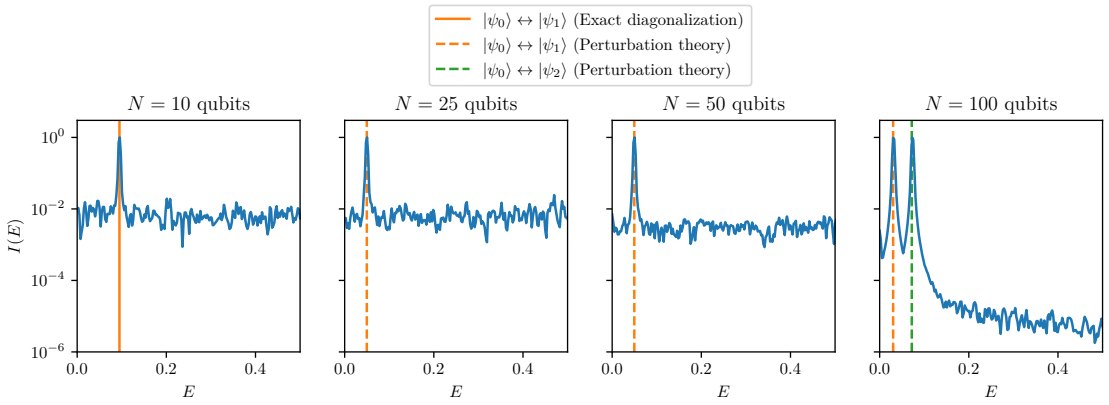



**Figure 3.3: Logarithmic intensity shadow spectra of a disordered Heisenberg spin chain ($J = 0.01$) up to $N = 100$ qubits verifying the scalability of our approach.** A fixed input state is used, consisting of an equal-weight superposition of the ground state and first excited state of the $J = 0$ problem. Shadow spectra are computed for system sizes $n \in \{10, 25, 50, 100\}$ using synthetic shot noise equivalent to $N_S = 100$ snapshots per timestep. At $N = 10$ qubits, the spectral peak matches the energy gap between the ground state and first excited state obtained from exact diagonalization (vertical solid orange line). At system sizes beyond the reach of exact diagonalization, I compare the peaks to second-order perturbation theory (vertical dashed lines), observing peaks corresponding to the gap between the ground and first excited states (orange), and at $N = 100$ an additional peak corresponding to the gap between ground and second excited states (green). Crucially, the standard deviation of the baseline decreases with an increasing number of qubits $N$ (i.e. SNR increases with $N$) supporting the scalability of shadow spectroscopy — at a fixed number of snapshots $N_S$, peaks become *more* distinguishable from noise at larger system sizes due to the increased number of available signals.

a singular-value decomposition at each bond, and discarding all singular values below a threshold $\epsilon_{\text{SVD}}$. This cutoff hyperparameter is chosen to be $\epsilon_{\text{SVD}} = 10^{-9}$ for the simulations presented here. In this way, I obtain a series of time-evolved MPS states as required for shadow spectroscopy, which I then use to compute time-dependent expectation values for all up-to-$q$-local Pauli strings (here $q = 3$) by applying the appropriate tensor contractions. I introduce synthetic shot noise to the signals by adding Gaussian random numbers of standard deviation $1/\sqrt{N_S}$ to all observables. I assume $N_S = 100$ shots per timestep, a comparatively very constrained budget for which Gaussian noise is nonetheless typically a very good approximation of shot noise in practice. I conduct simulations up to $n = 100$ qubits, noting that $n = 50$ qubits is already well beyond the capabilities of what one could ever simulate exactly on a classical computer. Although at these system sizes one

obtains a very large number of time-dependent signals (e.g., the number of up to 3-local Pauli observables for $n = 100$ is 4,410,750), the postprocessing protocol remains tractable on a laptop computer. I reject signals with $p > 0.01$ under a Ljung-Box test, and then perform the postprocessing detailed in Section 3.1.2 to generate the shadow spectra. This analysis is performed for system sizes up to $n = 100$ qubits, generating the spectra depicted in Figure 3.3.

To identify the shadow spectrum peaks, I compare the observed peaks to results from exact diagonalization (only at $n = 10$) and to second-order Rayleigh-Schrödinger perturbation theory ($n \in \{25, 50, 100\}$). Apart from a small visible inaccuracy in the green dashed line (where perturbation theory is less accurate for the higher excited states) I observe very good agreement, as expected. I reiterate that the choice of a perturbative regime was deliberate to allow peak identification, i.e., for a weak coupling $J$, the actual values found by shadow spectroscopy are close to the perturbative prediction, but do not exactly agree. Finally, I confirm that the SNRs of the spectral peaks indeed *increase* with the number of qubits $n$, providing further numerical evidence that shadow spectroscopy remains scalable at system sizes that would warrant a genuine quantum advantage in the strong-coupling regime. With both numerical (exact and tensor-network) and analytic evidence supporting the scalability of shadow spectroscopy, I now proceed to explore its applications to near-term problems.

## 3.3 Application to near-term problems

As outlined in Section 1.2, the constraints of the NISQ era severely limit the type of algorithms that can be executed on near-term quantum computers. In particular, the limited coherence times and gate fidelities of current devices constrain circuit depth, rendering it impossible to simulate time evolution to the timescales required to resolve energy gaps to a good resolution. Indeed, applying even 20 Trotter steps for modestly-sized Ising systems is currently regarded as state-of-the-art for quantum simulation [52]. Instead, in the NISQ era, it can be more practical to use ansatz-based variational quantum simulation, which was discussed in my literature

review (Section 2.2.2). In this section, I briefly outline the relevant aspects of this method (Section 3.3.1), and demonstrate its use for shadow spectroscopy of the disordered Heisenberg chain (Section 3.3.2).

### 3.3.1 Variational dynamics

This section makes use of the original approach to variational dynamics introduced in Ref. [29], in contrast to the more advanced method I subsequently developed in Chapter 4. While the methods outlined in this section may require significantly more shots than the shadow spectroscopy protocol itself, it seems likely that by combining the two methods, one could still perform shadow spectroscopy with only logarithmic shot cost, since the methods I outline in Chapter 4 also make use of classical shadows.

In this approach to variational dynamics, one uses a fiducial input state $|\mathbf{0}\rangle$ and a PQC $U(\boldsymbol{\theta})$, such that the state of a system at time $t$ is given by $|\psi(t)\rangle = U(\boldsymbol{\theta}_t)\,|\mathbf{0}\rangle$ for time-dependent parameters $\boldsymbol{\theta}_t$. At each timestep of duration $\Delta t$, one must find the optimal parameters $\boldsymbol{\theta}_t$ that minimize the distance between the variational state and the true time-evolved state such that $U(\boldsymbol{\theta}_{t+\Delta t})\,|\mathbf{0}\rangle \approx e^{-iH\Delta t}U(\boldsymbol{\theta}_t)\,|\mathbf{0}\rangle$. For a non-trivial problem and shallow ansatz, it will not be possible to exactly construct arbitrary states in the dynamics, and so this is achieved by projecting the evolution onto the subspace of expressible states using a variational principle. While several choices of variational principle exist [187], here I use the McLachlan variational principle. The authors of Ref. [29] present quantum circuits which allow one to compute the matrix elements

$$\begin{aligned} A_{jk} &= \frac{\partial\,\langle\psi(\boldsymbol{\theta})|}{\partial\theta_j}\frac{\partial\,|\psi(\boldsymbol{\theta})\rangle}{\partial\theta_k}, \\ C_j &= \frac{\partial\,|\psi(\boldsymbol{\theta})\rangle}{\partial\theta_j}H\,|\psi(\boldsymbol{\theta})\rangle\,. \end{aligned} \tag{3.36}$$

Crucially, these matrices are small, having dimensions equal to the number of variational parameters rather than the full (exponential dimension) Hilbert space. These form the linear system given by McLachlan's principle as

$$\sum_k \mathrm{Re}(A_{jk})(\dot{\boldsymbol{\theta}})_k = \mathrm{Im}(C_j). \tag{3.37}$$

One could then update the parameters with Euler's rule as $(\boldsymbol{\theta}_{t+\Delta t})_j = (\boldsymbol{\theta}_t)_j + (\dot{\boldsymbol{\theta}})_j \Delta t$. However, in practice, the metric tensor $A$ will often be ill-conditioned due to degeneracies in parameter space, and so I apply Tikhonov regularization (ridge regression) by instead computing

$$\dot{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\phi}} \|\boldsymbol{C} - \boldsymbol{A}\boldsymbol{\phi}\|^2 + \lambda \|\boldsymbol{\phi}\|^2, \tag{3.38}$$

thus regularizing the problem. For the Heisenberg chain studied in this section, I find that a regularization hyperparameter $\lambda = 10^{-4}$ and a timestep $\Delta t = 10^{-2}$ is sufficient to keep the fidelities with respect to the true dynamical state above 99%. I note also that McLachlan's principle minimizes the distance between the ideal state and its variational approximation with respect to the $L^2$ norm, and therefore the given forms of the metric tensor and gradient vector (3.36) do not account for the possibility of a global phase mismatch. Even though observables are global-phase-invariant, the *derivatives* of the state with respect to the variational parameters are not, which can lead to a very different metric tensor $A$, thus resulting in incorrect dynamics. This can be corrected by including a 'virtual' global phase parameter in the ansatz, which is equivalent to including the global phase correction terms in the linear system [187].

A Wick-rotated form of these dynamics yields an approximation of imaginary time evolution, allowing the estimation of ground states. This is known as variational imaginary time evolution, or quantum natural gradient [146–148], and is essentially VQE augmented with information about the shape of the optimization landscape. This procedure is identical to the one described, except the linear system in question is instead

$$\sum_k \mathrm{Re}(A_{jk})(\dot{\boldsymbol{\theta}})_k = -\,\mathrm{Re}(C_j). \tag{3.39}$$

### 3.3.2 Demonstration of variational shadow spectroscopy

Using the variational approach described above, I now present a demonstration of shadow spectroscopy within NISQ constraints. I consider the same Heisenberg

spin chain with random site-dependent magnetic fields as in the tensor-network simulations (3.33), albeit at a smaller system size ($n = 6$) since full-density-matrix simulations are required. For the variational ansatz $U(\boldsymbol{\theta})$, I use $L = 5$ layers of the ansatz structure depicted in Figure 3.4(a), which is amenable to the requirements of the NISQ era due to its local connectivity and relatively shallow depth. Per the discussion of Section 3.1, it is crucial to have an initial state with high overlap with both the ground state and low-lying excited states. To achieve this, a (intentionally) partially-converged VQE can be used. To produce the initial state for the dynamics, I initialize $U(\boldsymbol{\theta})$ from uniform random parameters, and begin iteratively minimizing its energy with the imaginary time evolution protocol detailed at the end of Section 3.3.1. However, I intentionally terminate the algorithm before complete convergence is attained, resulting in parameter values $\boldsymbol{\theta}$ that strongly, but not perfectly, overlap with the ground state, with its most dominant other components being low-lying excited states. An example of these overlaps is given in the inset of Figure 3.4(b). The advantage of these near-converged initial states is that a majority overlap with the ground state guarantees strong signals that correspond to energy differences between the ground state and the excited states, but not between the excited states themselves. In principle, one could even assign peaks to specific low-lying excitations by repeating shadow spectroscopy on initial states of varying convergence levels and comparing peak amplitudes.

In Figure 3.4(b) (blue line) I numerically simulate variational time evolution circuits in $N_T = 1000$ time steps and perform shadow spectroscopy by estimating up to 3-local Pauli operators (3 batches of $N_S = 50$ snapshots used in each step). Here, the shot noise is included by *explicitly* simulating the snapshot procedure outlined in Section 3.1.1, and therefore has the correct distribution for classical shadows. The circuit simulations use exact density matrices, and are implemented in QuEST [103] using the QuESTlink framework [104]. Furthermore, to capture the limitations of NISQ devices, the circuit simulations are burdened by a noise level comparable to presently available devices: each two-qubit gate is accompanied

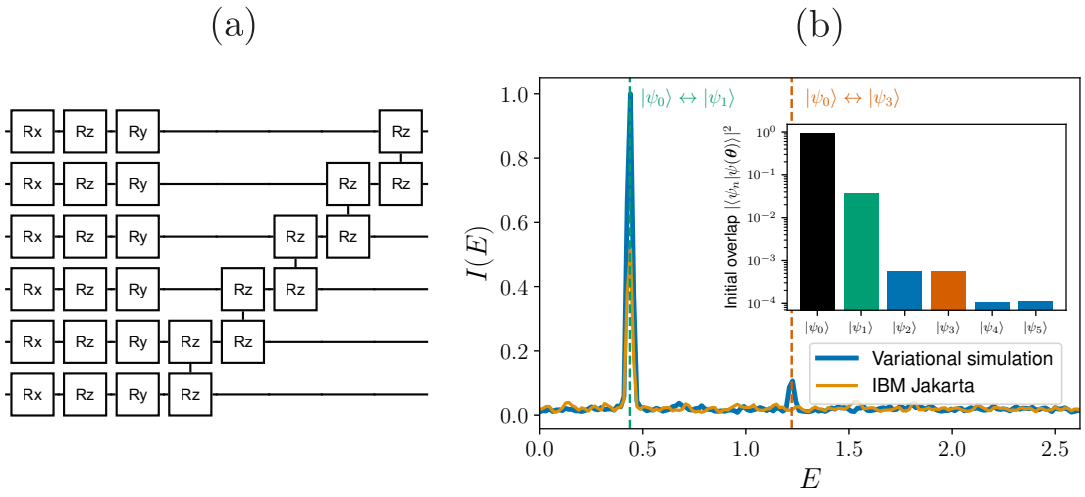



**Figure 3.4: Simulated and experimental shadow spectroscopy on a 6-qubit NISQ system. (a)** A single layer of the hardware-efficient ansatz used for variational dynamics of the disordered Heisenberg chain. **(b)** Shadow spectra of the 6-qubit disordered Heisenberg chain. A noisy simulation of the variational dynamics (blue) reveals a strong peak corresponding to the gap between the ground state and first excited state, and a less intense peak corresponding to the gap between the ground state and third excited state. In contrast, the experimental spectrum (orange) obtained from the IBM Jakarta QPU has a fainter primary peak, and an entirely suppressed (beyond shot-noise) secondary peak. In both cases, the spectra have the same error ($2.8 \times 10^{-3}$) in the gap estimate. **Inset:** overlaps $|\langle\psi_n|\psi(\boldsymbol{\theta})\rangle|^2$ between the initial variational state $|\psi(\boldsymbol{\theta})\rangle$ and eigenstates $|\psi_n\rangle$.

by a two-qubit depolarizing noise map described by the channel

$$\Phi^{\lambda}_{i,j}(\rho) = \left(1 - \frac{16\lambda}{15}\right)\rho + \frac{\lambda}{15}\sum_{\nu_1,\nu_2\in\{1,x,y,z\}} \sigma_i^{\nu_1}\sigma_j^{\nu_2}\rho\,\sigma_j^{\nu_2}\sigma_i^{\nu_1}. \tag{3.40}$$

The two-qubit gates are burdened with an error rate $\epsilon_2 = 10^{-3}$, and each single-qubit gate is accompanied by a single-qubit depolarizing noise map of error rate $\epsilon_1 = 0.25\epsilon_2$. Indeed, the spectrum in Figure 3.4(b) reveals that peaks corresponding to low-lying excitations are accurately resolved in a highly NISQ-friendly manner using shallow quantum circuits and very few shots.

I highlight that expected values of only 3-local Pauli strings were used. Local operators are a particularly appropriate choice for problems of this type, since in translation-invariant, gapped Hamiltonians of spin systems the intensity $|\langle\psi_k|O_i|\psi_l\rangle|$ is an exponentially decreasing function of the locality of $O_i$ [345].

As my simulations demonstrate the feasibility of this protocol on current devices, it is natural to attempt to apply it on actual quantum hardware. Using the circuits

I prepared (classically) by the above methods, my collaborator executed these circuits on the 7-qubit IBM Jakarta quantum computer using the same budget of snapshots as in my simulations. Further details of the implementation of this hardware experiment are provided in Appendix H of our full manuscript [334]. The experimental shadow spectrum obtained from this hardware experiment is also depicted in Figure 3.4(b) (orange line). It clearly resolves the main peak and achieves the same precision in predicting the energy gap as the simulated variational algorithm without using any error mitigation techniques, as one would expect from the gate error robustness outlined in Section 3.1.4. However, the faint secondary peak is suppressed by hardware noise, and is not resolvable against the shot-noise floor.

## 3.4 Conclusions

In this chapter, I have introduced shadow spectroscopy, a novel method for evaluating energy gaps in the spectrum of a Hamiltonian using a quantum computer.

Before drawing full conclusions, it is crucial to note that shadow spectroscopy is more versatile than the applications I have presented here, which are limited to my unique contributions to our full manuscript [334]. In this chapter, I have outlined the method itself, as well as tensor-network and NISQ simulations of shadow spectroscopy — nonetheless, it is a versatile tool that extends beyond the NISQ era and can be applied in every era of quantum computing. We anticipate that shadow spectroscopy will also find utility on both early-fault-tolerant and fully-fault-tolerant quantum computers. For the former, my collaborator Richard studied its use for the evaluation of gaps in the Fermi-Hubbard model under early-fault-tolerant assumptions, finding promising performance and introducing a novel method for mitigating the effect of algorithmic errors on the peak positions. For the latter, my collaborator Hans studied the application of shadow spectroscopy to chemical systems in the fully-fault-tolerant era, demonstrating the evaluation of low-lying gaps in the spectrum of methylene. I encourage the interested reader to consult their work for a full picture of shadow spectroscopy.

Throughout this chapter, I mostly motivated shadow spectroscopy as a *solution to the problem* of evaluating Hamiltonian spectrum gaps. However, in the development of quantum algorithms, solutions often precede problems, and this provides an alternate perspective on the usefulness of shadow spectroscopy. The classical shadow techniques outlined in Section [324] caused enormous excitement in the field: the ability to extract information about a huge number of observables to fixed additive error using only $\mathcal{O}(\log(n))$ measurements is an enormously appealing prospect, and is a very efficient use of (typically limited) measurement resources. Holding such a powerful hammer in hand begs the question: what nails can we hit with it? When is it useful to extract many (or indeed all) $q$-local observables cheaply? This pairs conveniently with our observation that the amplitude of oscillations in observable signals $\langle O \rangle$ is proportional to the eigenbasis matrix elements $\langle \psi_j | O | \psi_k \rangle$, which we cannot know *a priori*. If one does not know which observables will have significant signals, then one can gain a competitive edge over similar methods by measuring as many as possible as efficiently as possible with classical shadows, and using judicious classical postprocessing to filter out noise and exploit redundancy in the signals. Shadow spectroscopy is therefore not only a useful new tool for probing low-lying transition energies in quantum systems, but a compelling answer to the pertinent question 'what are classical shadows good for?'.

In addition to efficiently utilizing quantum measurement resources and exploiting the underlying redundancy in observable signals, shadow spectroscopy is advantageous in its use of *uncontrolled* time evolution. Most related methods require that time evolution be controlled on one or more ancilla qubits, adding additional overhead and gate connectivity requirements. QPE requires $B$ ancillary qubits for $B$ bits of precision, and methods based on Loschmidt echoes require at least one ancillary qubit for a Hadamard test. In contrast, shadow spectroscopy does not require ancillary qubits and can be executed using regular uncontrolled time evolution, requiring only initial state preparation, time evolution, and measurement in arbitrary Pauli bases (i.e. single-qubit Pauli rotations). Not only does this reduce the resources required to perform shadow spectroscopy in a gate-based quantum

computer, but it also raises the possibility of performing shadow spectroscopy on analog quantum simulators which lack universal operations. Analog quantum simulators, particularly those based on ultracold atoms, have achieved remarkable results recently [346], and shadow spectroscopy could be a valuable addition to their capabilities. Furthermore, operating a gate-based quantum computer (e.g. a trapped-ion device) in an analog fashion to simulate a hardware-tailored Hamiltonian and making appropriate measurements [347] could prove to be a fruitful platform for shadow spectroscopy.

One key limitation of shadow spectroscopy is the requirement that the initial state has a significant overlap with the relevant eigenstates of the Hamiltonian. Depending on the system in question, there are a variety of methods that one could use to attempt to prepare such an initial state. In the case of the variational variant of shadow spectroscopy I presented in Section 3.3, I used partially-converged VQE to prepare such a state — an appealing prospect, since this potentially turns the tendency of VQE to get 'stuck' in local minima into an advantage. For some systems, a problem-specific choice of initial state may be easily constructed. For example, my collaborator Hans considered Hartree-Fock states (which can be calculated classically and then efficiently prepared in linear depth using Givens rotations [348, 349]) as the foundation for initial states for shadow spectroscopy on chemical Hamiltonians. Heuristically, one expects that such classical approximations will have a reasonable overlap with the ground state, except perhaps in cases where the system is very strongly correlated. One could also consider methods like adiabatic state preparation, which are commonly employed in other quantum algorithms that depend on initial state overlap [21]. This limitation is not unique to shadow spectroscopy, and even fully-fault-tolerant eigenstate calculations with QPE require good overlap with the target state, as I discussed in Section 2.4.1. Thus, I still anticipate that shadow spectroscopy will compare well to QPE in the near-to-mid term future, and will be a useful addition to the toolbox for studying quantum systems on quantum computers.

Overall, this chapter has shown that quantum simulation techniques can be of great utility in evaluating gaps in the spectrum of a Hamiltonian. In the following chapter, I explore how related techniques can be employed to evaluate more structured aspects of the spectrum (namely, the density of states), and therefore evaluate thermodynamic properties of quantum systems using a quantum computer.

*Now it is our turn to study statistical mechanics.*

— D.L. Goodstein, *States of Matter* [350]

# 4

# Direct estimation of the density of states for fermionic systems

*I conducted the research in this chapter, supervised by Bálint Koczor. Expanding upon a core idea conceived by Bálint, I developed a full framework and methodology which led to the preprint of Ref. [193], which this chapter closely follows. We worked together on the analytic results, and all numerical results were solely my work. Text from the published manuscript is used verbatim in places — in these cases I am the original author of this text.*

## Contents


**4.1 Preliminaries . . . . . 76**
4.1.1 Physical systems of constant particle number . . . . . 76
4.1.2 Physical systems of variable particle number . . . . . 77
**4.2 Results . . . . . 79**
4.2.1 Estimating the FDOS using DQC1 circuits . . . . . 80
4.2.2 Estimating the FDOS via random-state initialization . . 83
4.2.3 Reconstructing the DOS . . . . . 87
4.2.4 Directly estimating thermodynamic properties . . . . . 90
**4.3 Random state sampling requirements . . . . . 91**
**4.4 Applicability in the early-fault-tolerant era . . . . . 96**
4.4.1 Resilience to algorithmic error . . . . . 97
4.4.2 Resilience to gate noise . . . . . 100
**4.5 Applicability in the NISQ era . . . . . 103**
4.5.1 Variational dynamics via classical shadows and covariance root-finding . . . . . 103
4.5.2 Demonstrating noisy variational DOS estimation . . . . 107

**4.6 Conclusions . . . . . 111**

In Chapter 3, I showed how frequency-space information from signals obtained by time evolution on a quantum computer can reveal useful properties of a Hamiltonian's spectrum. Another valuable property related to a Hamiltonian's spectrum is the density of states (DOS), a central object in statistical mechanics which is a distribution for the density of eigenstates as a function of energy, and can be used to compute equilibrium thermodynamic properties of quantum systems (see formal definitions in Section 4.1). As I outlined in Section 2.4.3, computing the DOS is a known computationally hard problem [304] for which quantum computers may provide an advantage, and a range of fault-tolerant [305] and nearer-term algorithms [161–165, 306–309] have been proposed for its (direct or indirect) estimation. The latter can be broadly grouped into two categories: those based on time evolution [306–308] (or related Hamiltonian moments [309]) and a Hadamard test, which are not explicitly compatible with fermionic systems; and those based on first computing Green's functions [161–165], which typically depend on expensive ground and excited state preparation (which can in general be exponentially hard [288]). Indeed, in Chapter 3, the greatest bottleneck to our methods was initial state preparation.

Fortunately, this need not be the case here, and in this chapter I bridge this gap, combining the simplicity of random-state Hadamard-test methods with the utility of fermionic systems. Using a conceptually simple approach based on time evolution and extremely simple initial states, I introduce several key innovations that significantly improve the practicality and extend the generality of previous techniques. First, this approach allows one to compute the DOS on *any* subspace of the full Hilbert space for which a basis can be prepared, which crucially includes fermionic subspaces of fixed number. Second, this approach applies time evolution to very simple random initial states — as simple as random bit-flips — representing a major improvement compared to techniques using more demanding random initializations, such as 2- and 3-design circuits [306–309, 323]. Third, one uses a

simple windowed Fourier transform to estimate the DOS up to a broadening of lines, allowing for clear tradeoffs between resolution in the DOS and evolution duration, and providing high levels of robustness to algorithmic and hardware errors.

I summarize basic preliminaries from statistical mechanics in Section 4.1, and build upon these in Section 4.2 to derive the key analytic results for this new protocol. Supporting my work with detailed numerical simulations, I study the convergence of different sampling protocols in Section 4.3. In Section 4.4, I numerically study the robustness of the approach against algorithmic (Trotter) errors and hardware noise in an early-fault-tolerant setting, finding surprising resilience in both. Finally, I consider a variational variant of the approach for NISQ applications in Section 4.5, introducing a new variational technique for time evolution, and draw conclusions from my work in Section 4.6.

## 4.1 Preliminaries

In this chapter, I outline novel quantum algorithms to estimate thermal properties of quantum systems at finite temperature. Therefore, I first review the relevant thermodynamic definitions and quantities.

### 4.1.1 Physical systems of constant particle number

I first consider a $d$-dimensional quantum system on a Hilbert space $\mathcal{H}$ with a Hamiltonian $H$ defined only for a fixed number of particles, such as a spin chain. The key thermodynamic quantity required to determine equilibrium thermal properties in the canonical ensemble (i.e., equilibrium with a thermal reservoir at inverse temperature $\beta$) is the *partition function*, which is defined as

$$\begin{aligned} Z(\beta) &:= \mathrm{Tr}\left[e^{-\beta H}\right] \\ &= \int dE e^{-\beta E} g(E). \end{aligned} \tag{4.1}$$

Here $g(E)$ is the DOS, which is defined as a series of Dirac delta distributions at the Hamiltonian eigenenergies as

$$g(E) := \sum_{k=1}^{d} \delta(E - E_k). \tag{4.2}$$

The DOS is thus a central quantity as it allows one to determine $Z(\beta)$, and therefore any further equilibrium thermodynamic property — for example, the internal energy is obtained as $U(\beta) = -\partial(\log Z)/\partial\beta$. In this chapter, I outline new quantum algorithms for estimating $g(E)$ that improve upon and generalise related algorithms introduced by prior works [161–165, 306–309]. A review of other related approaches is given in Section 2.4.3.

### 4.1.2 Physical systems of variable particle number

Many physical systems encountered in practice are defined on Hilbert spaces that simultaneously accommodate multiple, varying particle numbers, typically with fermionic or bosonic exchange properties. For example, *ab initio* models in quantum chemistry are defined on Fock spaces that accommodate varying numbers of electrons, with an $n$-mode fermionic system partitioning into $n+1$ subspaces of fixed particle number. Thus for such systems, applying the conventional expression for the DOS in Equation 4.2 would simultaneously contain information about system configurations of multiple particle numbers. For such systems, the central object we seek instead is the density of states $g_M(E)$ on a fixed-particle-number subspace of the full Hilbert space

$$g_M(E) \equiv \sum_{k=1}^{k_M} \delta(E - E_{k,M}), \tag{4.3}$$

where the subscript $M$ indicates correspondence to the subspace of $M$ total particles. Here I have assumed that the system's Hamiltonian $H$ commutes with a total number operator $N$ as $[H, N] = 0$, such that eigenstates $|k, m\rangle$ can be labelled by a quantum number $k \in \mathcal{S}_m$ within a fixed subspace of total particle number $m$ (here $\mathcal{S}_m$ collects all indices that belong to the same particle number). This object can be used to compute the density of states in both the canonical ensemble (thermal equilibrium with a thermal reservoir at inverse temperature $\beta$) and the grand canonical ensemble (thermal equilibrium with a thermal and particle reservoir at inverse temperature $\beta$ and chemical potential $\mu$). First, consider the canonical-ensemble partition function

for a fixed particle number, defined as

$$\begin{aligned} Z_M(\beta) &\equiv \mathrm{Tr}_M[e^{-\beta H}] \\ &= \int dE e^{-\beta E} g_M(E). \end{aligned} \tag{4.4}$$

Indeed, this closely resembles Equation 4.1, but the trace only includes eigenstates contained in the subspace of particle number $M$ as $\mathrm{Tr}_M[\cdot] = \sum_{k \in \mathcal{S}_M} \langle k, M| \cdot |k, M\rangle$. The subspace DOS may also be used to compute the partition function in the grand canonical ensemble, given by

$$\begin{aligned} \mathcal{Z}(\beta, \mu) &\equiv \mathrm{Tr}\left[e^{-\beta(H-\mu N)}\right] \\ &= \sum_M e^{\beta \mu M} Z_M(\beta), \end{aligned} \tag{4.5}$$

which is a weighted sum of canonical partition functions at varying particle number $M$, weighted by powers of the fugacity $z = e^{\beta\mu}$. Therefore, in both the canonical and grand canonical ensembles, determining the subspace density of states $g_M(E)$ is sufficient to access all equilibrium thermal properties. The methods presented in this chapter represent, to the best of my knowledge, the first quantum algorithms capable of doing so without prohibitively expensive eigenstate preparation routines.

Throughout this chapter, as an archetypal example of a fermionic many-body Hamiltonian, I will refer to the Fermi-Hubbard model, whose Hamiltonian is

$$H = -J \sum_{\langle \boldsymbol{j}, \boldsymbol{k} \rangle, \sigma} \left( c^\dagger_{\boldsymbol{j}\sigma} c_{\boldsymbol{k}\sigma} + c^\dagger_{\boldsymbol{k}\sigma} c_{\boldsymbol{j}\sigma} \right) + U \sum_{\boldsymbol{j}} n_{\boldsymbol{j}\uparrow} n_{\boldsymbol{j}\downarrow}. \tag{4.6}$$

Here $c_{\boldsymbol{j}\sigma}$ ($c^\dagger_{\boldsymbol{j}\sigma}$) is the fermionic annihilation (creation) operator for site $\boldsymbol{j}$ and spin $\sigma \in \{\uparrow, \downarrow\}$, $n_{\boldsymbol{j}\sigma} = c^\dagger_{\boldsymbol{j}\sigma} c_{\boldsymbol{j}\sigma}$ is the spin-density operator for spin $\sigma$ on site $\boldsymbol{j}$, and $\langle \boldsymbol{j}, \boldsymbol{k} \rangle$ indicates summation over neighbouring sites. This system is depicted in Figure 4.1(a). Since $[H, N] = 0$ for total number $N = \sum_{\boldsymbol{j}} (n_{\boldsymbol{j}\uparrow} + n_{\boldsymbol{j}\downarrow})$, its eigenstates have well-defined total number $M$. In Figure 4.1(b), I illustrate the subspace density of states $g_M(E)$ at a varying $M$ for a particular instance of the Fermi-Hubbard Hamiltonian (although not actually required for the algorithm as discussed in Section 4.2.3.1, for convenient normalization of variables I rescale the Hamiltonian such that its maximum/minimum eigenvalues are $\pm 1$).

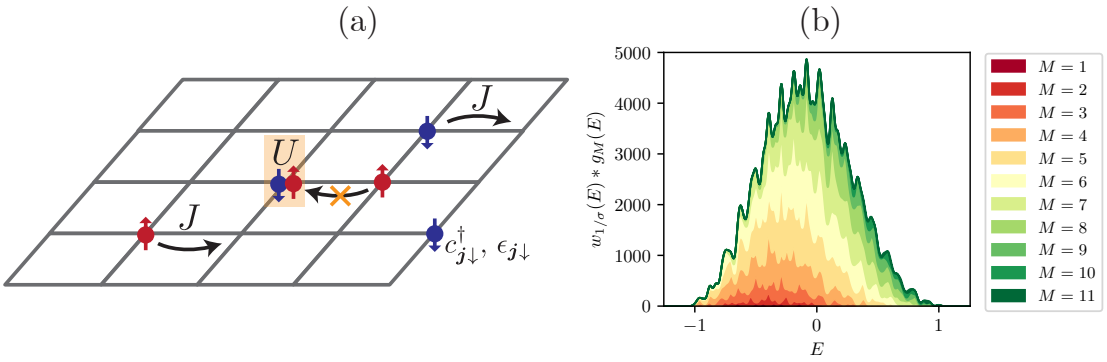



**Figure 4.1: The Fermi-Hubbard model and its fixed-particle-number DOS. (a)** The Fermi-Hubbard model describes spinful fermions hopping between nearest-neighbour sites (assumed throughout this chapter to be on a grid topology, as depicted here) with hopping integral $J$. Opposite-spin fermions can occupy the same site with interaction energy $U$, but the Pauli exclusion principle forbids same-spin fermions from doing so, imposing a maximum particle number $M$. Graphic adapted from Ref. [351]. **(b)** I cumulatively plot the subspace DOS $g_M(E)$ (up to a broadening of lines discussed in Section 4.2.3.1) for a $(3 \times 2)$ open-boundary grid Fermi-Hubbard model (4.6) with $J = -1$, $U = 2$ for a series of different particle numbers $M$. Each $M$ has a different DOS (different colours) and a different normalization factor $|\mathcal{S}|$ which has important implications for computing the partition function $Z_M(\beta)$ or the grand canonical partition function $\mathcal{Z}(\beta, \mu)$. Here, $M = 6$ is the most populous subspace.

## 4.2 Results

In this section, I outline the foundations and analytic details of our approach to computing the DOS. Much like the shadow spectroscopy technique explored in Chapter 3, our approach is based on a Fourier analysis of temporal signals obtained from time evolution on a quantum computer. Therefore, the Fourier transform of the DOS (FDOS) will be relevant throughout, which I define below for systems of fixed particle number. A generalization to systems of variable particle number will be explored in Section 4.2.1.2.

**Statement 1.** *For physical systems with fixed particle number, the FDOS* $G(t) := \mathcal{F}[g(E)](t)$ *can be computed as the trace of the time-evolution operator as*

$$G(t) = \frac{1}{\sqrt{2\pi}} \mathrm{Tr}\left[e^{-iHt}\right]. \tag{4.7}$$

*Proof.*

$$G(t) \equiv \mathcal{F}[g(E)](t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} dE e^{-iEt} g(E) \tag{4.8}$$

$$= \frac{1}{\sqrt{2\pi}} \sum_{k=1}^{d} e^{-iE_k t} \tag{4.9}$$

$$= \frac{1}{\sqrt{2\pi}} \mathrm{Tr}\Big[e^{-iHt}\Big], \tag{4.10}$$

where I have used the unitary, angular-frequency convention for the Fourier transform and the DOS definition of Equation 4.2. I note that for all time signals in this work, $G(t=0) = d/\sqrt{2\pi}$ and $G(-t) = (G(t))^*$, so only forward time evolution is required. It follows that the DOS can be reconstructed via an inverse Fourier transform as

$$g(E) = \mathcal{F}^{-1}[G(t)](E) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} dt e^{iEt} G(t). \tag{4.11}$$

□

### 4.2.1 Estimating the FDOS using DQC1 circuits

#### 4.2.1.1 One Clean Qubit (DQC1) computation

Equation 4.7 may be rewritten in terms of the maximally mixed state $\rho_{\max} = \mathbb{1}/d$ as

$$G(t) = \frac{d}{\sqrt{2\pi}} \mathrm{Tr}\Big[\rho_{\max} e^{-iHt}\Big]. \tag{4.12}$$

Several related, prior works [306–308] estimate the FDOS based on the above equation, by sampling random pure initial states (since the average of all random initial basis states is the maximally mixed state). To derive new results in the following sections, it is essential here to consider another approach, whereby the FDOS can be computed using a purification of the maximally mixed state [352]. When $H$ is an $n$-qubit Hamiltonian, one adds an $n$-qubit ancillary register and creates Bell pairs between the two registers as illustrated in Figure 4.2(a) – indeed, the reduced density matrix of the main register is the maximally mixed state, while the composite system is in a pure state. One then performs a Hadamard test, with the time evolution operator controlled on an ancilla qubit, and thereby deterministically encoding the FDOS into the ancilla qubit's amplitude. Similar

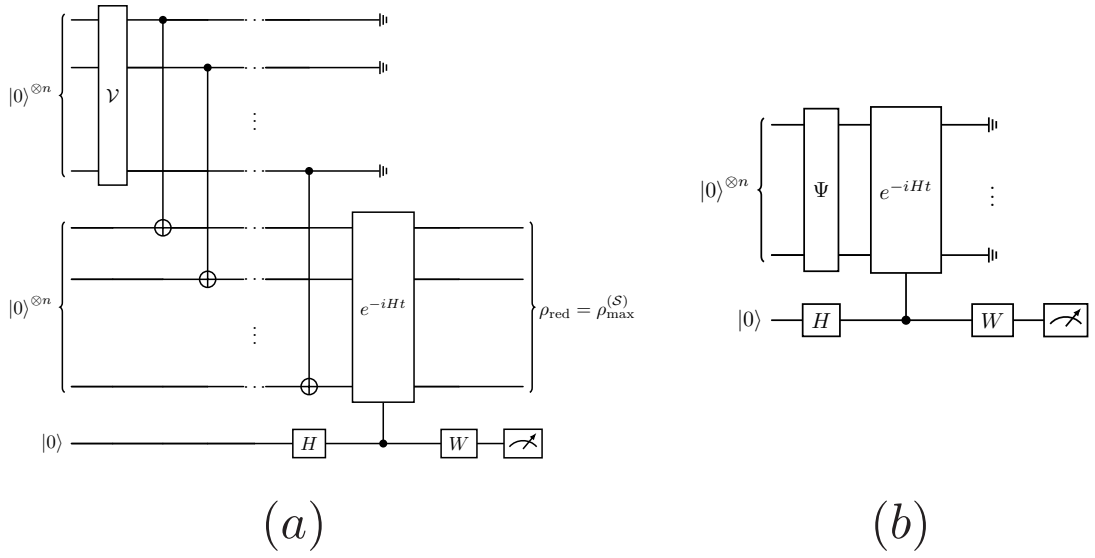



**Figure 4.2: Quantum circuits for evaluating the FDOS. (a)** DQC1 circuits: First, setting $\mathcal{V}$ to be the Hadamard transform $H^{\otimes n}$ allows one to estimate the FDOS over the entire Hilbert space as relevant for Section 4.1.1. This creates Bell pairs between the two registers such that the reduced density matrix of the lower, main register is maximally mixed — as in standard DQC1 computations. Second, setting $\mathcal{V}$ to be a circuit that prepares a uniform superposition $|\mathcal{S}|^{-1/2}\sum_k |\psi_k\rangle$ allows one to estimate the FDOS $G_{\mathcal{S}}(t)$ on a subspace (Section 4.1.2) spanned by $\{|\psi_k\rangle\}_{k\in\mathcal{S}}$. I detail an efficient construction of $\mathcal{V}$ for fermionic Hamiltonians in Section 4.2.1.2. **(b)** Alternatively, the FDOS can be estimated by random state sampling methods outlined in Section 4.2.2. Here the main register is initialised in a randomly chosen initial state and and thus the need for a second register to prepare a maximally mixed state is eliminated entirely. The random intialisation circuit $\Psi$ can be extremely simple — even random single-qubit bit-flips suffice, and we explore further variants in Section 4.3. The real (imaginary) parts of the FDOS are encoded as the ancilla probability by setting $W{=}H$ ($W{=}S^\dagger H$).

circuits have been investigated in the literature in the context of the one clean qubit model (DQC1) [353], and therefore I will henceforth refer to the approach as a DQC1 computation. Finally, repeatedly measuring the ancilla qubit allows one to estimate the FDOS under standard shot noise scaling. In a fully fault-tolerant setting, one could also apply amplitude estimation to obtain a fundamentally improved precision — albeit at the cost of a significantly increased circuit depth.

#### 4.2.1.2 Modified DQC1 computation for general subspaces

Consider now a subspace spanned by a set of energy eigenstates $\{|\psi_k\rangle\}_{k\in\mathcal{S}}$ (e.g. those with fixed particle number). The density of states projected onto this

subspace is given by

$$g_{\mathcal{S}}(E) = \sum_{k\in\mathcal{S}} \delta(E - E_k), \tag{4.13}$$

and its FDOS $G_{\mathcal{S}}(t) \equiv \mathcal{F}[g_{\mathcal{S}}(E)](t)$ can be computed straightforwardly on a quantum computer as follows.

**Result 1.** *The subspace FDOS $G_{\mathcal{S}}(t)$ can be computed via a simple modification to the DQC1 protocol depicted in Figure 4.2(a), whereby one replaces the Hadamard transform with a sub-circuit $\mathcal{V}$ that produces a uniform superposition of all orthogonal states $\{|\phi_j\rangle\}$ in the subspace (in any basis).*

*Proof.* Denoting the projection onto this subspace as $\mathcal{P}_{\mathcal{S}}$, one has

$$G_{\mathcal{S}}(t) \equiv \mathcal{F}[g_{\mathcal{S}}(E)](t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} dE g_{\mathcal{S}}(E) e^{-iEt} \tag{4.14}$$

$$= \frac{1}{\sqrt{2\pi}} \sum_{k\in\mathcal{S}} e^{-iE_k t} \tag{4.15}$$

$$= \frac{1}{\sqrt{2\pi}} \operatorname{Tr}\Big[\mathcal{P}_{\mathcal{S}}(e^{-iHt})\Big] \tag{4.16}$$

$$= \frac{|\mathcal{S}|}{\sqrt{2\pi}} \operatorname{Tr}\Big[\rho_{\max}^{(\mathcal{S})} e^{-iHt}\Big], \tag{4.17}$$

where $\rho_{\max}^{(\mathcal{S})} \equiv \frac{1}{|\mathcal{S}|} \sum_{k\in\mathcal{S}} |\psi_k\rangle \langle\psi_k|$ is the maximally mixed state on the subspace. Since $\rho_{\max}^{(\mathcal{S})} = \frac{1}{|\mathcal{S}'|} \sum_{j\in\mathcal{S}'} |\phi_j\rangle \langle\phi_j|$ for any orthonormal basis $\{|\phi_j\rangle\}$ of the subspace (where $\mathcal{S}'$ denotes the collection of indices in this basis), the result follows in analogy to the full-Hilbert-space case. □

The complexity of constructing the circuit $\mathcal{V}$ may strongly depend on the simulated physical system. However, for the pivotal case of fermionic systems under the Jordan-Wigner encoding — one of the most prolific target problems in quantum simulation — one can construct these circuits explicitly. For these systems, computational basis states have well-defined particle number equal to their Hamming weight; the uniform superposition of fixed-Hamming-weight states is known as a Dicke state, and indeed $\mathcal{V}$ must be a Dicke-state preparation circuit. Fortunately, this can be implemented in $\mathcal{O}(n)$ depth [354]. Indeed, as I show in the following section, the protocol can be performed by randomly preparing

individual initial states spanning the subspace, rather than their uniform mixture — a substantially easier task. This approach can therefore be used to estimate the DOS on any subspace for which the basis states can be efficiently prepared. This may be useful when considering symmetries and conserved quantities other than particle number — for example, magnetic properties of a system might be elucidated by comparing the DOS on subspaces of fixed total spin $m_s$.

### 4.2.2 Estimating the FDOS via random-state initialization

Computing the FDOS via the circuit in Figure 4.2(a) requires $2n+1$ qubits for an $n$-qubit Hamiltonian and potentially a state-preparation circuit of depth $\mathcal{O}(n)$. Similarly, previous works considered Monte Carlo importance sampling methods [306–308] or approximating higher-order 3-design random circuits [309] — where the latter similarly introduces additional circuit depth. In contrast, I now present an approach that only requires one ancillary qubit and introduces negligible overhead in circuit depth. This is achieved via random-state sampling, utilizing a very straightforward 1-design construction.

**Statement 2.** *Suppose we randomly choose from a unitary 1-design $\{U_i\}_i$ according to the probability distribution $p_i$, therefore obtaining a spherical 1-design in the states $|\phi_i\rangle = U_i |0\rangle$. Defining a random variable in terms of these random states given by their Loschmidt echoes at fixed time $t$*

$$\hat{L}(t) = \langle \phi_i | \, e^{-iHt} \, |\phi_i\rangle = \langle \phi_i(0) | \phi_i(t) \rangle, \tag{4.18}$$

*then one may construct an unbiased estimator of the FDOS*

$$\hat{G}(t) = d(2\pi)^{-1/2} \; \hat{L}(t), \tag{4.19}$$

*such that $\mathbb{E}\left[\hat{G}(t)\right] = G(t)$.*

*Proof.* Expanding the expected value one obtains

$$G(t)/d = \mathbb{E}\left[\frac{1}{\sqrt{2\pi}} \langle\phi_i| e^{-iHt} |\phi_i\rangle\right] \tag{4.20}$$

$$= \frac{1}{\sqrt{2\pi}} \sum_i p_i \langle\phi_i| e^{-iHt} |\phi_i\rangle \tag{4.21}$$

$$= \frac{1}{\sqrt{2\pi}} \sum_i p_i \operatorname{Tr}\left[e^{-iHt} |\phi_i\rangle \langle\phi_i|\right] \tag{4.22}$$

$$= \frac{1}{\sqrt{2\pi}} \operatorname{Tr}\left[e^{-iHt} \sum_i p_i U_i |0\rangle \langle 0| U_i^\dagger\right], \tag{4.23}$$

where I have used linearity and basis-invariance of trace in the last step. By definition a 1-design recovers Haar-random averages in the first moment, and thus

$$G(t)/d = \frac{1}{\sqrt{2\pi}} \operatorname{Tr}\left[e^{-iHt} \int_\psi U |0\rangle \langle 0| U^\dagger \mathrm{d}U\right] \tag{4.24}$$

$$= \frac{1}{\sqrt{2\pi}} \operatorname{Tr}\left[e^{-iHt} \rho_{\max}\right], \tag{4.25}$$

where I have used that the integration over the Haar measure yields the maximally mixed state $\rho_{\max}$. □

I note that $\hat{L}(t)$ is the overlap between initial random state $|\phi_i\rangle$ and its time-evolved counterpart $|\phi_i(t)\rangle = e^{-iHt} |\phi_i\rangle$, and can therefore be computed via the standard Hadamard test circuit in Figure 4.2(b).

This conveniently replaces the initialization to the maximally mixed state in Figure 4.2(a) with initialization to $|\phi_i\rangle$ via the random circuit $\Psi$, reducing the required number of qubits from $2n+1$ to $n+1$. Furthermore, Statement 2 implies that unitary 1-design circuits are sufficient and allows one to forgo the significantly more complex random circuits that approximate higher moments of the Haar distribution used in previous works [306–309, 323].

Indeed, constructing a 1-design can be extremely straightforward. One such construction consists of applying bitflips with 50% probability to the individual qubits that are initialized to the computational zero state, and thus $\{U_i\} = \{\mathbb{1}, X\}^{\otimes n}$ and $p_i = 1/2^n$. I numerically explore further 1-designs in Section 4.3, such as continuous $\mathrm{SU}(2)^{\otimes n}$ rotations, and confirm that the choice of 1-design does not

affect the convergence or shot requirements (so long as a new sampling circuit is utilized for each circuit repetition).

As mentioned in Section 2.4.3, most *ab initio* models of genuine utility will have a fermionic and/or bosonic component. This necessitates computing the FDOS on subspaces of fixed number (Section 4.2.1.2), which in turn requires one to sample states uniformly within the relevant subspace. Although I assumed a 1-design property in Statement 2 to emphasize how this chapter's approach improves upon previous 2- and 3-design constructions [306–309, 323], even less structure than a 1-design is sufficient. This observation also enables the present approach to be easily generalized to subspaces of fixed particle number.

**Statement 3.** *Statement 2 holds for any set of states $|\phi_i\rangle$ and probabilities $p_i$ that satisfy $\sum_i p_i |\phi_i\rangle \langle\phi_i| = \rho_{max}$. This allows one to naturally generalise the previous results to subspaces without requiring 1-design properties. In particular, given a set of states within the subspace $\{|\psi_k\rangle\}_{k\in S}$ and a probability distribution $p_i$ that satisfy the property*

$$\sum_{k\in S} p_i |\phi_i\rangle \langle\phi_i| = \rho_{\max}^{(\mathcal{S})}, \tag{4.26}$$

*then estimating the Loschmidt echo of randomly chosen initial states $\hat{L}_{\mathcal{S}}(t) = \langle\phi_i| e^{-iHt} |\phi_i\rangle$ yields the unbiased estimator for the FDOS as*

$$\hat{G}_{\mathcal{S}}(t) = |\mathcal{S}|(2\pi)^{-1/2}\ \hat{L}_{\mathcal{S}}(t), \tag{4.27}$$

*such that $\mathbb{E}\left[\hat{G}_{\mathcal{S}}(t)\right] = L_{\mathcal{S}}(t)$.*

*Proof.* One can re-prove Statement 2 by starting the same way up to Equation 4.22, and then move the summation inside the trace as

$$G(t)/d = \frac{1}{\sqrt{2\pi}} \sum_i p_i \operatorname{Tr}\left[e^{-iHt} |\phi_i\rangle \langle\phi_i|\right] \tag{4.28}$$

$$= \frac{1}{\sqrt{2\pi}} \operatorname{Tr}\left[e^{-iHt} \sum_i p_i |\phi_i\rangle \langle\phi_i|\right]. \tag{4.29}$$

By applying $\sum_i p_i |\phi_i\rangle \langle\phi_i| = \rho_{\max}$ one obtains the desired expression, without the need to assume uniform probabilities as in Statment 2. For the subspace approach,

one assumes the property $\sum_i p_i \left|\phi_i\right\rangle \left\langle\phi_i\right| = \rho_{\max}^{(\mathcal{S})}$, thus obtaining the analogous expression for $G_{\mathcal{S}}(t)$. □

Recall again that for fermionic Hamiltonians under the Jordan-Wigner encoding, computational basis states have well-defined particle number $M$ equal to their Hamming weight. It therfore follows Statement 3 that the set of computational basis states $\{|k\rangle\}_{k\in S}$ of fixed Hamming weight $M$ sampled uniformly ($p_i = |\mathcal{S}|^{-1}$) is sufficient to estimate the FDOS on a subspace of fixed particle number $M$. More generally, one could also initialize in a state of fixed particle number and randomly apply number-conserving transformations, such as matchgates [330], to obtain a desired set of random initial states.

Finally, I remark that the above sampling approaches are efficient, since they admit a standard shot-noise scaling.

**Statement 4.** *The number of samples required to estimate $G_{\mathcal{S}}(t)/|\mathcal{S}|$ in Statement 3 (for the full Hilbert space $\mathcal{S} \equiv \mathcal{H}$ and $|\mathcal{S}| = d$) to precision $\epsilon$ is upper-bounded as $N_s \leq (2\pi)^{-1}\epsilon^{-2}$. Similarly, repeatedly measuring the ancilla qubit in the DQC1 circuit (Figure 4.2(a)) achieves the same bound.*

*Proof.* The variance of the estimator is given by

$$\mathrm{Var}[\hat{G}(t)/|\mathcal{S}|] = (2\pi)^{-1}\ \mathrm{Var}\left[\hat{L}(t)\right] \leq (2\pi)^{-1}\ \mathbb{E}\left[|\hat{L}(t)|^2\right], \tag{4.30}$$

and given that $|\hat{L}(t)|^2 \leq 1$ it follows that $\mathrm{Var}[\hat{G}(t)/|\mathcal{S}|] \leq (2\pi)^{-1}$. Thus, the number of samples required to estimate $G(t)/|\mathcal{S}|$ to precision $\epsilon$ is upper bounded through the variance of the mean estimator as $N_s \leq (2\pi)^{-1}\epsilon^{-2}$. □

Importantly, the random initial state sampling approach in Figure 4.2(b) achieves the same sample complexity as the DQC1 circuits in Figure 4.2(a), and therefore can be substituted for no additional cost (other than potential challenges in varying the executed circuit each shot, which I address in Section 4.3). Although the maximally-mixed-state formulation is a theoretical convenience that is useful for proofs, explicitly preparing a maximally-mixed state on near-term devices is a challenging prospect that has hindered similar protocols [352]. In contrast, trivial

single-qubit operations (i.e. bit-flips) are sufficient for this chapter's approach, rendering the impact of improper state preparation negligible (e.g. preparation of states of correct particle number is assured to within single-qubit gate fidelity). Nonetheless, there may be advantages to using the more complex DQC1 circuits in a fault-tolerant setting. For example, one could apply amplitude estimation [19] to the DQC1 circuits to estimate the ancillary probability, which would enable a fundamentally improved time complexity, such that the runtime is $O(\epsilon^{-1})$. In exchange, however, the circuit depth becomes similarly $O(\epsilon^{-1})$, which would prove significantly more challenging for NISQ and early-fault-tolerant devices.

### 4.2.3 Reconstructing the DOS

In the previous sections, I detailed how to estimate the FDOS at a fixed time $t$ using a quantum computer. Formally, the DOS can be written as the inverse Fourier transform of the FDOS as in Equation 4.11, but with finite resources, a quantum computer can only estimate $G(t)$ at discrete time intervals, and up to some finite time. In the section that follows, I discuss the practicalities of this truncation, and how this can be directly linked to the estimation of equilibrium thermal properties.

#### 4.2.3.1 Windowed time evolution

In this chapter, thermal properties are computed via a signal $G(t)$ (which may or may not be on a subspace), which is computed in discrete steps of $\Delta t$. As outlined in the above sections, a (discrete) inverse Fourier transform can then be used to obtain the DOS $g(E)$. To avoid spectral leakage causing distortion in the estimate of $g(E)$, it is prudent to use a Fourier window function [355], which I denote as $W_\sigma(t)$ (where $\sigma$ is some characteristic temporal width). The inverse convolution theorem then guarantees

$$\mathcal{F}^{-1}[W_\sigma(t)G(t)] = w_{1/\sigma}(E) * g(E), \tag{4.31}$$

where $*$ denotes convolution $f(E) * h(E) = \int d\epsilon f(\epsilon)h(E-\epsilon)$ and $w_{1/\sigma}(E) = \mathcal{F}^{-1}[W_\sigma(t)](E)$ is the inverse Fourier transform of the window function. I note

that for typical choices of window function, $w_{1/\sigma}(E)$ is a point-spread function with characteristic energy width $1/\sigma$, and therefore Equation 4.31 represents the evaluation of the DOS up to a broadening of lines (i.e. 'blurring' of the DOS). I henceforth refer to

$$\tilde{g}^{\sigma}(E) \equiv w_{1/\sigma}(E) * g(E) \tag{4.32}$$

as the *resolution-limited DOS*. While a broad range of window functions are available in the literature that make different tradeoffs [356], in this chapter I will consider a standard Gaussian window of width $\sigma$ and unit normalisation $\int dE w_{1/\sigma}(E) = 1$ as

$$W_{\sigma}(t) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{t^2}{2\sigma^2}\right). \tag{4.33}$$

The sole exception to this will be the 'natural' window induced by noise in Section 4.4.2. To illustrate the effect of the window function (in particular, its temporal width $\sigma$), in Figure 4.3(a) I plot subspace FDOS signals for a Fermi-Hubbard Hamiltonian for increasing Gaussian window widths $\sigma$. In Figure 4.3(b), one can see how this blurs out spectral features in the resulting resolution-limited DOS $\tilde{g}_6^{\sigma}(E)$ reconstructions.

In complete generality, the resolution of the DOS improves with an increasing window width $\sigma$. However, it requires longer-duration time evolution to evaluate $G(t)$, as resolving spectral features in the DOS to an energy scale $\Delta E = \mathcal{O}(1/\sigma)$ necessitates evolution up to a timescale $\mathcal{O}(\sigma)$. Since required circuit depths will typically increase with the duration of time evolution, there is a trade-off between the resolution of the DOS and the required resources (e.g. coherence times). This is indeed the main limitation of this chapter's approach, as time evolution can be relatively challenging even for relatively short timescales using NISQ and early fault-tolerant devices. Nonetheless, even relatively shallow circuits can be very useful in estimating certain thermodynamic properties: for example, it may be possible to determine qualitative features with short temporal widnow widths, such as inspecting the DOS for the presence of an insulator gap.

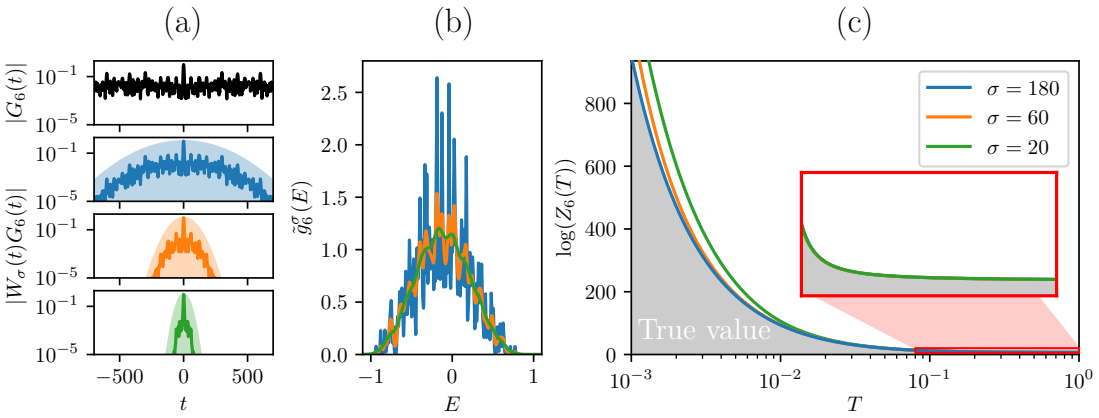



**Figure 4.3: Effects of finite simulation length.** I study a $(3 \times 2)$ grid Fermi-Hubbard model with open boundary conditions, nearest-neighbour couplings, $J = -1$ and $U = 2$, and assuming the fixed-number subspace $M = 6$. **(a)** The FDOS $G_6(t)$ (black) is estimated using a quantum computer and allows reconstruction of the DOS $g_6(E)$ through an inverse Fourier transform. As detailed in Section 4.2.3.1, to cap the total time evolution length, the signal is multiplied in post-processing by a Gaussian Fourier window $W_\sigma(t)$ of temporal width $\sigma = 180$ (blue), $\sigma = 60$ (orange), and $\sigma = 20$ (green) — shaded regions represent the amplitude of the window function. **(b)** Different resolution-limited DOS $\tilde{g}^\sigma(E)$ are obtained from each signal in (a): wider temporal window width (e.g. blue, $\sigma = 180$) leads to finer-resolution spectral features while shorter time evolutions (e.g. green, $\sigma = 20$) blur out spectral features. **(c)** The partition function $Z_6(T)$ from Equation 4.4 is estimated for increasing temperatures $T = 1/\beta$ using the resolution-limited DOS $\tilde{g}^\sigma(E)$ from (b) — the shaded grey region represents the true partition function. Low-temperature physics imposes stricter requirements on the duration of dynamics: shorter temporal window widths (green and orange) yield poorer approximations at low temperatures due to the concentration of the Boltzmann factor $e^{-\beta E}$ at low energies. In contrast, the inset illustrates that even short-time dynamics can be sufficient for accurate prediction of higher temperature properties.

I also highlight that unlike kernel polynomial methods [300, 309], there is no requirement to rescale the Hamiltonian such that its spectrum lies in a certain range, which would require some prior knowledge of the spectrum to reliably perform. Instead, the range and resolution of energies are determined by the duration of dynamics $t_{\max}$ and the sampling period $\Delta t$. In particular, sampling $t$ in $G(t)$ via a uniform grid of $N_t = t_{\max}/\Delta t$ points yields a uniform grid of energies between $-\frac{\pi}{N_t} + \frac{2\pi}{N_t \Delta t}$ and $\frac{\pi}{\Delta t}$; the range of energies can then be increased by decreasing $\Delta t$, while the spectral resolution can be improved by increasing the duration of evolution.

Alternatively, rather than uniformly sampling $t$ in $G(t)$, one could estimate the windowed Fourier transform in Equation 4.31 by Monte Carlo importance

sampling methods. In this approach, one samples the FDOS $G(\hat{t})$ at random times $\hat{t}$ which are drawn from a probability distribution $P(t) = W_\sigma(t)/\{\int_{-\infty}^{\infty} dt W_\sigma(t)\}$. This approach leads to the same spectral resolution limits and to the same shot-noise properties, since FDOS is still evaluated by shot-noise-limited sampling. However, an advantage of this Monte-Carlo approach is that it allows evaluation of $\tilde{g}^\sigma(E)$ for a continuous $E$ in principle, as opposed to the specific frequencies obtained from a discrete Fourier transform at the fixed time points. This approach of importance sampling from a window function is commonly applied in other Fourier-limited methods for computing spectral properties [297].

### 4.2.4 Directly estimating thermodynamic properties

Once the resolution-limited DOS is obtained with the methods of the previous section, one can use it to estimate thermodynamic properties via Equation 4.4. In Figure 4.3(c) I illustrate that a relatively short temporal window width $\sigma$ does not significantly hinder estimating thermodynamic properties at relatively high temperatures (small $\beta$ via $T = 1/\beta$), since it primarily leads to inaccuracies at low temperatures (large $\beta$). This is of course expected from Equation 4.4, since for lower temperatures (larger $\beta$) the Boltzmann factor $e^{-\beta E}$ becomes more evenly spread out in $E$ leading to a diminished relevance of small fluctuations in the density of states when calculating $Z(\beta)$.

However, as I now show, thermodynamic properties in the canonical ensemble define effective window functions, which may be combined with importance sampling to directly compute these properties. Provided one can evolve the system to the sampled times, this eliminates approximations due to convolution in the resolution-limited DOS. Formally, thermodynamic properties are estimated via the integral

$$F(\beta) = \int f_\beta(E) g(E)\, \mathrm{d}E, \tag{4.34}$$

where $f_\beta(E)$ is a function that encodes the desired property. For example, one may directly compute the partition function $F(\beta) = Z(\beta)$ by setting $f_\beta(E) = e^{-\beta E}$, or the internal energy $F(\beta) = U(\beta)$ by setting $f_\beta(E) = Ee^{-\beta E}$.

To directly estimate thermal equilibrium properties without first going via the DOS, note that Parseval's theorem allows one to apply the Fourier transform to both functions without changing the result of the integral as

$$F(\beta) = \int \mathcal{F}[f_\beta(E)](t) G(t)\, \mathrm{d}t. \tag{4.35}$$

This expression can be estimated using importance sampling: one randomly chooses times $\hat{t}$ according to the probability density function

$$P(\hat{t}) = \frac{\mathcal{F}[f_\beta(E)](t)}{\int_{-\infty}^{\infty} dt \mathcal{F}[f_\beta(E)](t)}, \tag{4.36}$$

which can be straightforwardly computed numerically and for relatively simple functions $f_\beta(E)$ analytically. I highlight that a potential difficulty may be that the Fourier transform $\mathcal{F}[f_\beta(E)](t)$ has a long tail, e.g., the function may decrease asymptotically as $O(1/t)$ for increasing $t$, in which case large evolution times $\hat{t}$ may be required beyond capabilities of near-term hardware.

## 4.3 Random state sampling requirements

In Section 4.2.2, I concluded that the random initial state sampling approach in Figure 4.2(b) allows one to compute the FDOS using a comparable number of shots (circuit repetitions) as the more complex DQC1 circuits in Figure 4.2(a). Here I numerically compare this procedure when estimating the FDOS using the following four different sampling methods:

- measuring the ancilla qubit in DQC1 circuits
- exact Haar-random initial state sampling (which would be exponentially expensive)
- product state sampling by single-qubit rotations $e^{-i\theta_{m,1}\sigma_m^x} e^{-i\theta_{m,2}\sigma_m^z} e^{-i\theta_{m,3}\sigma_m^x}$ on each qubit $m$ with randomly chosen $\theta_{m,s}$ (Euler angles)
- computational basis state sampling by uniform randomly applying bitflips $\{\mathbb{1}, X\}$ to each qubit on the computational zero state

The focus in this section is to confirm convergence of the different approaches and verify that they indeed yield the same result (up to possibly slightly different behaviour when state preparation circuits are repeated, discussed below). In doing so, I confirm that these methods require only very simple initial states, circumventing the need for complicated initial-state preparation used in e.g. QPE, which may be a bottleneck for some high-utility problems [287], or the expensive and error-prone process of explicitly preparing a maximally-mixed state, which limits the applicability of some related protocols [352].

Since some of the above random sampling methods are ill-defined for subspace methods, in this section I choose the disordered Heisenberg chain (3.33) as a benchmark model, and estimate the DOS in the full Hilbert space. As a metric for comparison, I define the error in the approximated DOS as

$$\epsilon(f,\sigma) \equiv 1 - \frac{\langle f, \tilde{g}^\sigma \rangle}{\sqrt{\langle f, f \rangle \langle \tilde{g}^\sigma, \tilde{g}^\sigma \rangle}}, \tag{4.37}$$

where $\langle f, g \rangle = \int dE f(E) g(E)$ is the $L^2$ inner product, and $f(E)$ is the density of states estimated using $N_s$ shots.

Here, I compare the performance of different sampling methods using exact time evolution — I investigate the effect of algorithmic errors and gate noise in later sections (Sections 4.4.1, 4.4.2 and 4.5.2). In Figure 4.4 I plot the error $\epsilon(f,\sigma)$ for a range of total per-timestep shot budgets $N_s$, and indeed confirm that all techniques admit standard shot noise scaling as expected from Statement 4. Although not visible on this plot, one should expect the error $\epsilon(f,\sigma)$ to eventually plateau, since I cap the maximal allowed time in $G(t)$ in the signal-processing approach as detailed in Section 4.2.3.1.

Figure 4.4 illustrates that the DQC1 method and all three random-state sampling methods achieve the same performance, so long as a new random initial state is supplied for each shot when sampling random initial states (blue curves). Indeed, there is no advantage in using complex random sampling methods (such as approximate 2-designs [309]), since simple 1-design sampling (e.g. bit-flip sampling) achieves the same performance when a new initial state can be supplied for each shot.

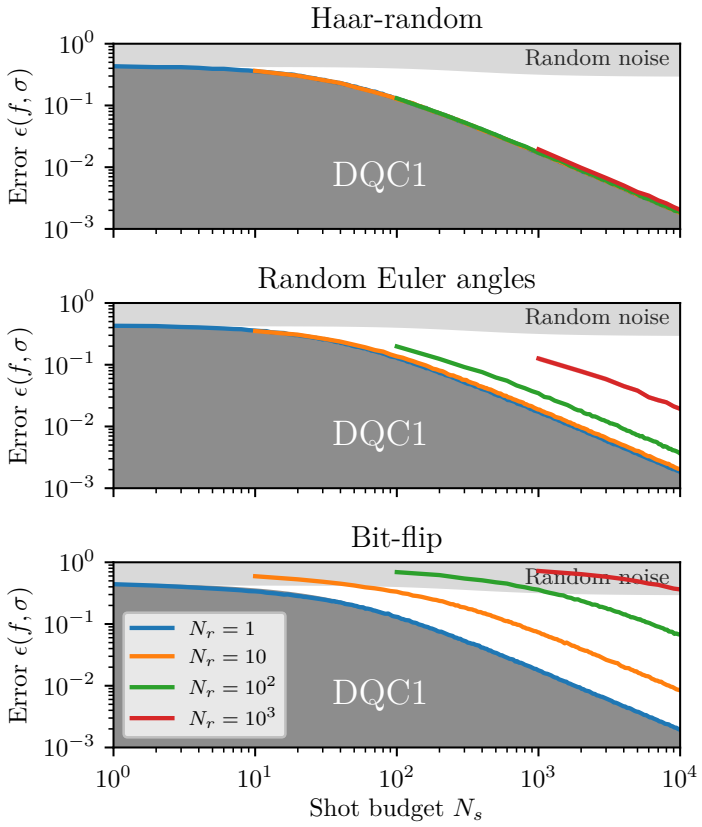



**Figure 4.4: Comparison of random initial-state sampling methods.** Error $\epsilon(f, \sigma)$ as a function of per-timestep shot budget $N_s$ in the DOS of a Heisenberg model (3.33) ($h = 1, J = 1, n = 12$), obtained using different random sampling methods (Haar-random, single-qubit continuous rotations, and random bit flips). Coloured lines represent that the same random initial state was reused $N_r$ times, since loading a new circuit is expensive on most hardware platforms and thus multiple repetitions of a single circuit are desirable. Lower dark grey shaded regions represent the errors obtained by DQC1 computation via the circuit of Figure 4.2(a), whereas the light grey shaded area represents completely random noise. Parallel lines in the above log-log plots confirm standard shot-noise scaling $N_s = O(\epsilon^{-2})$ from Statement 4 for all techniques, i.e. performances are only different by constant absolute factors, and the DQC1 and random initial-state sampling methods coincide as expected when $N_r = 1$. For larger values of $N_r$, the more uniform random sampling methods perform slightly better, however they may require more complex initialisation circuits — the single-qubit, random Euler angle approach is likely to be the most practical one. Confidence intervals are too small to visualize on this plot.

However in practice, on most quantum hardware platforms it is expensive to load a different circuit at each repetition, i.e. it is more efficient to 'bunch' shots together by repeatedly sampling the same initial state, and thus changing the circuit every $N_r$ shots. In this case, the total shot budget is $N_s = N_r \times N_\psi$, where $N_\psi$ random initial states are sampled. Figure 4.4 (orange, green, red) illustrates the performance when an initial random state is resued $N_r > 1$ times. One can clearly see that Haar-random sampling performance are almost uncompromised for up to $N_r = 10^3$, while random Euler-angle single-qubit sampling admits a moderately increased shot noise as $N_r$ increases at fixed $N_s$. In contrast, shot noise in bit-flip sampling is significantly increased for an increasing $N_r$. This illustrates that random-state sampling methods that sample the Hilbert space more uniformly can be advantageous when sampling circuits are repeated, as they may require fewer samples. Single-qubit Euler-angle sampling appears to be a particularly pragmatic alternative here, since it requires only single-qubit rotations but is more robust to repeated sampling circuits than simple bit-flips.

The motivation for studying the convergence in a non-fermionic (fixed-particle-number) model was the larger range of random-state sampling methods available, since methods like the random Euler-angle sampling do not preserve fermion number. This allowed me to showcase how uniformity in state sampling affects the impact of circuit repetitions $N_r$, but analogous results can be shown for fermionic problems. To do so, I perform the same sampling task on a Fermi-Hubbard model (4.6). As in the rest of this chapter, I utilize a $(3 \times 2)$ grid Fermi-Hubbard model with open boundary conditions, nearest-neighbour coupling, and parameters $J = -1$, $U = 2$. The DOS is computed with a window width $\sigma = 60$ (relative to the rescaling described in Section 4.2.3.1). Sampling is performed (a) with the modified fermionic-subspace DQC1 approach of Section 4.2.1.2, and (b) with fixed-Hamming-weight bit-flip sampling as outlined in Section 4.2.2. The results are depicted in Figure 4.5, again confirming that both sampling methods admit the same standard shot-noise-limited scaling, but this time for a fermionic problem. Beyond this simple bit-flip sampling approach, I anticipate that more uniform sampling could

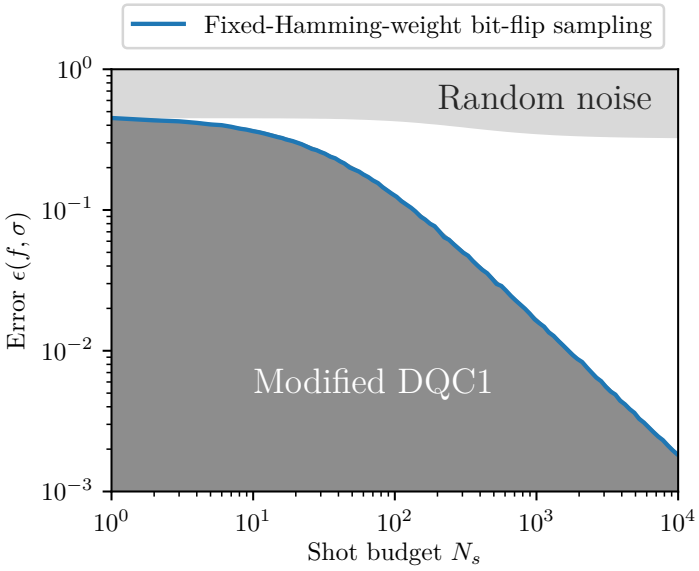



**Figure 4.5: Convergence of sampling methods for fermionic problems.** I plot the error $\epsilon(f,\sigma)$ as a function of per-timestep shot budget $N_s$ in the DOS of a Fermi-Hubbard model (4.6) using the modified DQC1 approach for fermionic subspaces (lower dark grey shaded region) and bit-flip initial state sampling at fixed Hamming weight (blue line). The upper light grey shaded area represents the error that would be obtained from a random white-noise signal. As expected, the errors for the fermionic-subspace DQC1 approach and the random-state-sampling approach coincide, again confirming that both sampling methods admit standard shot-noise scaling $N_s = O(\epsilon^{-2})$. The behaviour is overall analogous to that of the non-fermionic problem in Figure 4.4. Confidence intervals are too small to visualize on this plot.

be obtained by e.g. initializing in a state of definite number and applying random (number-conserving) matchgate circuits [330], yielding the benefits of more uniform sampling methods seen in Section 4.3 when $N_r > 1$.

To further explore this intuition that more uniform random-state sampling is advantageous when randomly-sampled initial states are repeated, I consider the use of a layered hardware-efficient ansatz circuit structure to generate random initial states by randomly sampling ansatz parameters. The circuit, depicted in Figure 4.6(a), consists of $L$ layers each consisting of $X$-$Y$-$Z$ rotations (Tait-Bryan angles), followed by nearest-neighbour $ZZ$ couplings. It is well known that this circuit structure generates a full-rank Lie algebra (see Chapter 5 for a deeper discussion on Lie algebras and quantum circuits), and thus generates an $\epsilon$-approximate 2-design with layers $L \propto \log(1/\epsilon)$ [213]. Consequently, varying $L$

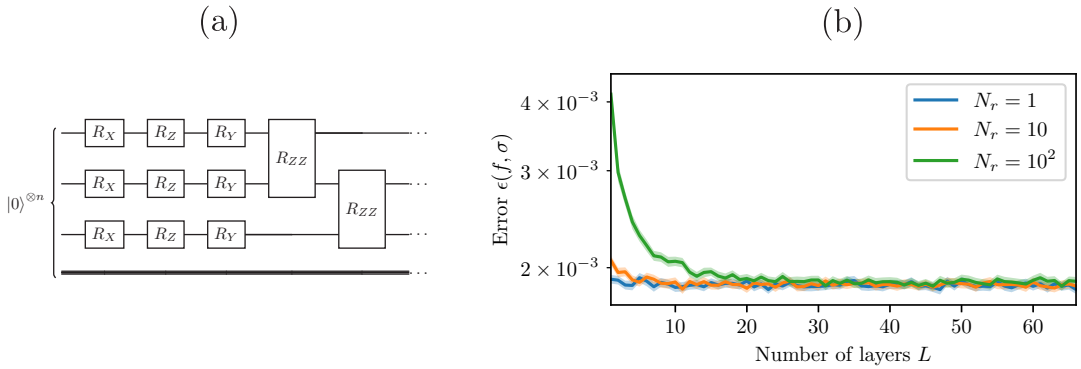



**Figure 4.6: Effect of initial-state-preparation circuit depth on DOS error.** **(a)** Structure of an ansatz layer for random initial state sampling, per the circuit in Figure 4.2. Each layer consists of arbitrary single-qubit rotations (parametrized by Tait-Bryan angles) and nearest-neighbour $ZZ$ rotations. **(b)** Error $\epsilon(f, \sigma)$ in the estimated DOS for an increasing number of layers $L$, at three different sampling circuit repetitions $N_r \in \{1, 10, 10^2\}$. Shaded bars represent bootstrapped 95% confidence intervals. Increasing $L$ interpolates between the 1-design and 2-design limits, the latter of which leads to reduced impact of sampling circuit repetition. Indeed, this confirms that more uniform random sampling of the Hilbert space is preferable when reusing initialisation circuits ($N_r > 1$).

should vary between the 1-design and 2-design limits. In Figure 4.6(b), I compare the error $\epsilon(f, \sigma)$ achieved for $N_s = 10^4$ shots per timestep at varying layer count $L$, for three different sample repetition numbers $N_r$. As expected, in the limit of many layers $L$ the three schemes converge to nearly the same error $\epsilon(f, \sigma)$.

## 4.4 Applicability in the early-fault-tolerant era

In Section 1.2, I alluded to the dawn of the early fault-tolerant era of quantum computing, where small numbers of error-corrected logical qubits can conduct simple computations. In this era, simple algorithms for dynamics such as Trotterization are likely to be feasible, and computations will be bottlenecked by algorithmic error (due to discretization of time evolution) and gate noise (originating from imperfect application of $T$ gates, rather than the underlying errors in entangling gates themselves). This state of affairs is very promising for the methods of this chapter, and thus in this section I conduct numerical simulations to investigate the praticality of this chapter's methods in an early fault-tolerant setting. By

performing explicit simulations of these protocols, I demonstrate a high degree of robustness to both algorithmic error (Section 4.4.1) and gate noise (Section 4.4.2).

### 4.4.1 Resilience to algorithmic error

While simulating time evolution is possibly the most natural application of quantum computers, these simulations are only approximate due to algorithmic errors. In this section, I implement Trotterized time evolution and investigate the effect of these algorithmic errors on the DOS in the subspace of particle number $M = 6$ as $g_6(E)$. In doing so, I demonstrate surprising levels of robustness to algorithmic error — even with low-fidelity evolution, a reasonably accurate estimate of the DOS can be obtained.

I consider the Fermi-Hubbard model (whose Hamiltonian is given in Equation 4.6) for a $(3 \times 2)$ grid with open boundary conditions, nearest-neighbour coupling, and parameters $J = -1$, $U = 2$. I map the Hamiltonian to qubits via a Jordan-Wigner encoding and rescale the spectrum as outlined in Section 4.2.3.1 to finally obtain $H = \sum_m H_m$. The time evolution under this Hamiltonian can be implemented via a first-order Trotterization as

$$\begin{aligned} U(\Delta t) &= \exp\left(-i\Delta t \sum_m H_m\right) \\ &= \prod_m \exp(-iH_m\Delta t) + \mathcal{O}(\Delta t^2), \end{aligned} \tag{4.38}$$

whereby all $\mathcal{O}(\Delta t^2)$ contributions are neglected. Controlled time evolution is then implemented by applying controlled variants of the individual local gates as $|0\rangle_a \langle 0|_a \otimes \mathbb{1} + |1\rangle_a \langle 1|_a \otimes \exp(-iH_m\Delta t)$, where $|0\rangle_a$, $|1\rangle_a$ denote ancilla states — the resulting controlled-Pauli operations can be straightfowardly implemented in the manner outlined in Section 4.4.2.

I first compare the fidelity of the ideal dynamics to the Trotterized dynamics via the unitary fidelity

$$\mathcal{F}_{\text{trotter}}(t) \equiv \frac{1}{d} \operatorname{Tr}\left[\exp(iHt) \prod_{s=1}^{t/\Delta t} \prod_m \exp(-iH_m\Delta t)\right].$$

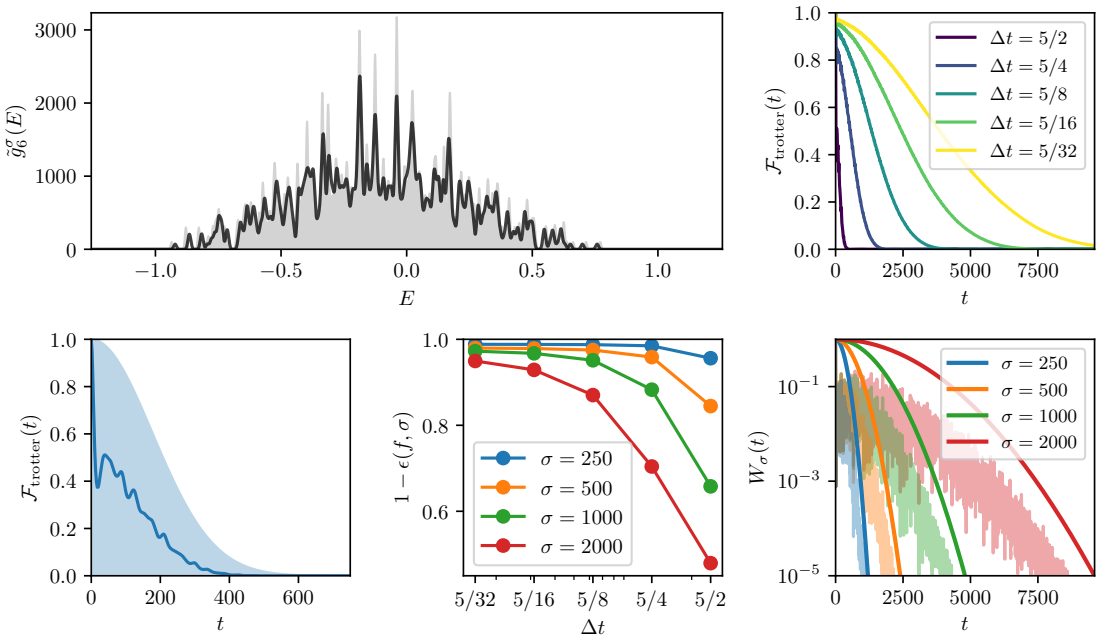



**Figure 4.7: Effect of algorithmic error on DOS calculations. Top left:** The DOS (solid line) of the $M = 6$ subspace of a $3 \times 2$ grid Fermi-Hubbard model ($J = -1$, $U = 2$) is a surprisingly accurate reproduction of the exact DOS (shaded grey region) despite very significant Trotter error, i.e., the unitary fidelity $\mathcal{F}_{\text{trotter}}(t)$ (solid blue line, **bottom left panel**) of the time evolution operator decays rapidly relative to the timescale set by the Fourier window (shaded region, bottom panel). Even with levels of Trotter error that would be too significant for most applications, the spectral content of the FDOS $G(t)$ remains accurate enough to correctly resolve the DOS. **Top right:** For a more systematic study, I compare five different Trotter steps $\Delta t \in \{5/2, 5/4, 5/8, 5/16, 5/32\}$ and depict the rate at which the unitary fidelity $\mathcal{F}_{\text{trotter}}(t)$ decays for each timestep $\Delta t$. **Bottom middle:** Using these timesteps $\Delta t$, I compare the error $\epsilon(f, \sigma)$ in the estimated DOS across a range of window widths $\sigma$. As expected, larger window widths $\sigma$ require shorter timesteps $\Delta t$, since these incorporate the effect of the signal $G(t)$ at later times $t$ where the accumulated effect of algorithmic error is greater. This is necessary to resolve more detailed spectral features, since the relevant spectral resolution is $\Delta E \sim 1/\sigma$. **Bottom right:** With respect to the timescale of the corresponding Fourier windows (solid lines) and windowed signals (transparent lines), the unitary fidelity $\mathcal{F}_{\text{trotter}}(t)$ decays non-negligibly, further confirming that our method is successful even with very significant algorithmic Trotter error.

For fixed timestep $\Delta t$, this leads to an exponential drop in the fidelity as a function of total simulation time $t$, which is depicted for various $\Delta t$ in Figure 4.7(top right).

I consider the modified DQC1 protocol outlined in Section 4.2.1, assuming the main register is in an initial maximally mixed state on the subspace of $M = 6$ (which can be efficiently prepared via $\mathcal{O}(n)$-depth Dicke state circuits) and the ancilla is in the $|0\rangle_a$ state. In Figure 4.7(top left), I show the evaluation of $g_6(E)$ for $\Delta t = 2.5$ and a window width of $\sigma = 250$. Despite the very poor fidelity of Trotter simulation, with $\mathcal{F}_{\text{trotter}}(t)$ in Figure 4.7(bottom left) almost immediately dropping below 0.5 and further decaying quickly relative to the the window timescale $\sigma$, I find that the DOS is reproduced to a surprisingly good level of accuracy. In fact, by comparing the individually resolved peaks in Figure 4.7 (top left, black vs grey lines) at the highest and lowest energies, it is clear that the Trotter error merely shifts the peaks to the left and right but does not significantly affect their peak heights. This can be understood analytically by noting that Trotterization implements time evolution under an effective Hamiltonian with a perturbative error term. By the Baker-Campbell-Hausdorff formula, this effective Hamiltonian can be written as $H_{\text{eff}} = H + \Delta t P + O(\Delta t^2)$, where the perturbation operator $P$ is composed of nested commutators between the individual Hamiltonian terms (i.e. terms of the Hamiltonian Lie algebra — see Chapter 5). Consequently, since the energies observed in the ancillary signal of controlled Trotterization are eigenvalues of $H_{\text{eff}}$, their deviation will be bounded as

$$\Delta E_k \leq \Delta t \|P\|_\infty + O(\Delta t^2). \tag{4.39}$$

These shifts can indeed be observed in Figure 4.7 (top left), since the Fourier window has a fine width $1/\sigma \ll \Delta E_k$, narrower than the energy shifts. This illustrates that in practice, one should rather choose a convolution window (and hence total time evolution length) such that $1/\sigma > \Delta E_k$, in which case the energy shifts due to Trotter error would not be resolvable. As there is no practical motivation to evolve longer than $\mathcal{O}(\sigma)$ satsifying this condition, it is clear that the Trotter error effectively limits the achievable resolution, and reducing it (via smaller $\Delta t$ and/or

higher-order product formulae) is necessary to achieve finer spectral resolution, which I now investigate further. In Figure 4.7(bottom middle), I compare the error $\epsilon(f, \sigma)$ achieved in the approximate DOS for varying window widths $\sigma \in \{250, 500, 1000, 2000\}$, across a range of time-steps $\Delta t \in \{5/2, 5/4, 5/8, 5/16, 5/32\}$. Here one can see another aspect to the relationship between window width and spectral resolution — as noted in Section 4.2.3.1, resolving spectral features in the DOS to an energy scale $\Delta E = \mathcal{O}(1/\sigma)$ necessitates evolution up to a timescale $\mathcal{O}(\sigma)$. However, accumulated algorithmic errors will increase in severity at larger $t$, necessitating smaller time-steps for accurate dynamics at later $t$. Nonetheless, I still find that the algorithmic error requirements are quite forgiving. Comparing the decay in unitary fidelities for these time-steps (Figure 4.7(top right)) to the timescales of the chosen window widths (Figure 4.7(bottom right)), one can see again that reasonably accurate determination of the DOS is possible even with fidelities that decay fast within the window timescale $\sigma$, demonstrating the robustness of our scheme to Trotter error in particular.

### 4.4.2 Resilience to gate noise

While algorithmic errors in the previous section can be suppressed through increasing circuit depths, hardware and gate noise pose inherent limitations on the achievable circuit depths. In this section, I consider the effect of gate noise on the Trotterized methods studied in Section 4.4.1, for a simulated early fault-tolerant device. I assume that in an early fault-tolerant setting the dominant source of error will be the imperfect implementation of continuous rotations (due to the relatively high cost of $T$ gates, e.g. due to imperfect magic state distillation). To model this numerically, I employ a modified version of the relevant error model in Ref. [334]. The high cost of magic state distillation in an early-fault-tolerant setting forces one to use short-depth Clifford+T sequences, which can be well approximated by an effective depolarizing noise model [357], which may be mitigated through more advanced techniques [358]. Following the example of the previous section, time-evolution is implemented via a

first-order Trotterization (4.38), which for controlled evolution of the Fermi-Hubbard model (4.6) under a Jordan-Wigner mapping generates terms of the form

$$|0\rangle_a \langle 0|_a \otimes \mathbb{1} + |1\rangle_a \langle 1| \otimes \exp\Big(-i\Delta t \sigma_i^\nu \sigma_{i+1}^z \dots \sigma_{j-1}^z \sigma_j^\nu\Big), \tag{4.40}$$

with $\nu \in \{x, y, z\}$. The Coulomb terms of the Hubbard Hamiltonian act on a single site $i = j$, and the hopping terms generate more general terms of the form in Equation 4.40. Furthermore, the $\sigma^z$ terms sandwiched between qubits $i$ and $j$ in the hopping terms can be removed by introducing a network of fermionic SWAP (FSWAP) gates, which only consists of local gates of depth $\mathcal{O}(N^{\frac{1}{2}})$ [348, 359, 360]. I do not explicitly model this, instead using it to assume an error model where local depolarizing noise of probability $\lambda$ is applied to the ancilla and qubits $i$ and $j$ for every controlled Pauli rotation. I parametrize this in terms of a total circuit error rate $\xi = \lambda N_{\text{gates}}$, where the Trotter circuit contains $N_{\text{gates}}$ error-susceptible gates due to imperfect application of $T$ gates.

The effect of this noise can be well understood in terms of the Fourier window functions outlined in Section 4.2.3.1. Since the logical qubits are effectively afflicted with Pauli noise ([357, 358]), the state fidelity exponentially decays with increasing circuit depth to a very good approximation [361]. This acts as an effective exponentially-decaying Fourier window applied to the temporal signals $G(t)$ as

$$W_\sigma(t) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{t}{\log(2)\sigma}\right), \tag{4.41}$$

where the normalization has been chosen such that $\mathcal{F}^{-1}[W_\sigma(t)](E) = \frac{(1/\log(2)\sigma)}{\pi((1/\log(2)\sigma)^2+\omega^2)}$ is unit-normalized, ensuring correct normalization of the DOS. The effective window width $\sigma$ is determined by the strength of depolarizing noise and circuit structure. Consequently, gate imperfections in Trotter circuits yield a broadening of spectral features and a reduction in spectral resolution $\Delta E$. This is not an unusual limitation — for most applications, the maximum number of gates is typically limited by the fact that $\xi = N_{\text{gates}}\lambda$ must not significantly exceed 1 [361], which imposes a limit on the duration of dynamics, and indeed imposes a limit on the best achievable

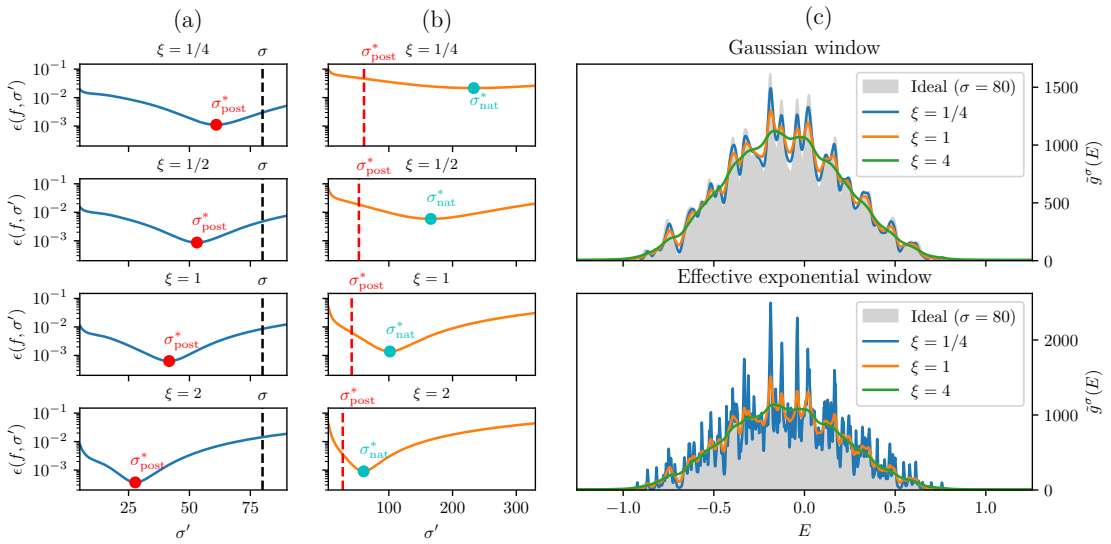



**Figure 4.8: Effect of hardware noise on DOS estimation.** I compute the error $\epsilon(f, \sigma)$ in the DOS of a $3 \times 2$ grid Fermi-Hubbard model ($J = -1$, $U = 2$) for $G_6(t)$ computed under the influence of an early fault-tolerant noise model at a range of different circuit error rates $\xi \in \{1/4, 1/2, 1, 2, 4\}$. **(a)** I compare the estimated DOS (computed with fixed window width $\sigma = 80$, black dotted line) to the ideal DOS that would be obtained with a perfect signal at window width $\sigma'$. Under the influence of noise, the error is minimized at a value of $\sigma' = \sigma^*_{\text{post}}$ *smaller* than $\sigma$, which decreases with increasing error rate $\xi$. This is due to the broadening-of-lines effect of gate noise. **(b)** Gate imperfections result in a natural effective exponential windowing. I compute the DOS using no additional windowing in postprocessing, and compare to the ideal DOS for an exponential window (4.41) for a range of widths $\sigma'$. At each noise level, the (noise-induced, effective) window width $\sigma^*_{\text{nat}}$ that minimizes the error is larger than the corresponding optimal window width $\sigma^*_{\text{post}}$ obtained in (a) for the same dynamics, demonstrating the 'natural' windowing induced by depolarizing noise in the dynamics. **(c)** Comparison of the DOS at differing error rates for Gaussian windowing (top) and no additional windowing (bottom). Increasing the error rate of the Trotter circuit leads to a broadening of lines in the DOS.

spectral resolution. Further reducing per-gate imperfections $\lambda$ in early fault-tolerant systems will allow for an increased spectral resolution.

In Figure 4.8(a), I estimate the DOS at five different circuit error rates $\xi \in \{0.25, 0.5, 1.0, 2.0, 4.0\}$, and plot the deviation $\epsilon(f, \sigma)$ from the ideal windowed DOS using a range of different window widths $\sigma'$. The unique minimum of the curves identifies the window width $\sigma^*_{\text{post}}$ that is most representative of the decay rate in the noisy DOS reconstructions, and indeed $\sigma^*_{\text{post}}$ decreases as one increases $\xi$.

In Figure 4.8(b), I compute the DOS with no additional windowing, relying only on the effective windowing introduced by gate imperfections, and compare

this to the ideal DOS with exponential window from Equation 4.41 applied for increasing widths $\sigma'$. Again, the unique minimum of the curves identifies the width $\sigma^*_{\text{nat}}$ that is most representative of the noisy evolution, which in contrast to the previous example is naturally induced by gate noise. Similarly, $\sigma^*_{\text{nat}}$ decreases with $\xi$. However, this window width is larger than that of the minimum in Figure 4.8(a), indicating that spectral information may be lost by additional windowing. While some additional windowing will be required in the presence of shot noise, the noise level should be taken into account when choosing a window width for postprocessing, since there will be a natural windowing effect due to gate noise.

## 4.5 Applicability in the NISQ era

Error correction on current-generation devices is not yet of a sufficient scale to be useful for quantum dynamics. In the NISQ era, one must contend with the underlying hardware errors of the physical gates themselves, which are typically too severe even for simple dynamics methods like Trotterization — applying even as few as 20 Trotter steps for modestly-sized Ising systems is at the edge of current capabilities [52]. Much like in Chapter 3, I apply ansatz-based variational methods to attempt to circumvent these circuit depth limitations. In Section 4.5.1 I introduce a new method for variational dynamics, with notable advantages compared to the method of Li *et al.* [29] which I used in Chapter 3. I then demonstrate numerically in Section 4.5.2 how this method can be used in tandem with the DOS estimation methods of this chapter to compute thermal properties of a Heisenberg model in the presence of realistic noise for the NISQ era.

### 4.5.1 Variational dynamics via classical shadows and covariance root-finding

In Section 3.3, I outlined the approach to variational dynamics introduced by Li *et al.* [29] and applied it to shadow spectroscopy. However, this approach requires one to compute the metric tensor for the ansatz (Equation 3.36), which requires a very large number of samples and can be sensitive to noise. Additionally, the Li

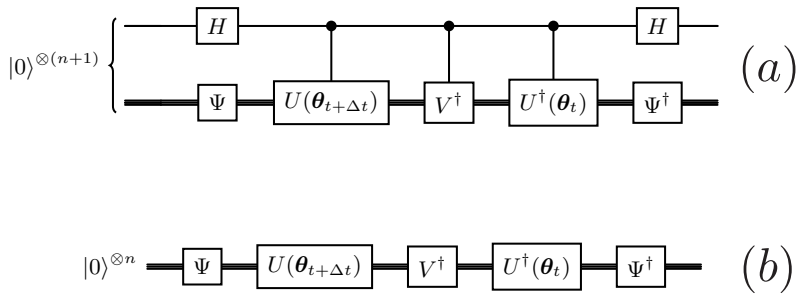



**Figure 4.9: Circuits for variational dynamics compilation.** The depicted circuits enable approximate recompilation of a Trotter step $V$ applied to some dynamical state $U(\boldsymbol{\theta_t})\Psi|0\rangle$, (a) such that $U(\boldsymbol{\theta}_{t+\Delta t})\Psi|0\rangle \approx VU(\boldsymbol{\theta}_t)\Psi|0\rangle$ including global phase or (b) such that $U(\boldsymbol{\theta}_{t+\Delta t})\Psi|0\rangle \approx e^{i\phi}VU(\boldsymbol{\theta}_t)\Psi|0\rangle$ for arbitrary global phase difference $\phi$. In both cases, the state produced by the depicted circuit minimizes the compilation Hamiltonian (4.43) for the desired values of $\boldsymbol{\theta}_{t+\Delta t}$, enabling variational optimization at each time step. I use the CoVaR optimizer [152] to train circuit parameters.

method is sensitive to degeneracies in parameter space, requiring regularization, and it is difficult to ascertain how well-converged the optimization is. In contrast, methods based on recompiling Trotter steps [189–191] can be far simpler, using simple gradient methods on a cost function defined by an optimization 'Hamiltonian' (see e.g. Equation 4.43). Furthermore, the minimum value of this cost function is usually analytically known, providing a proxy to ascertain how well-converged each step is relative to the fiducial dynamics.

There is scope to refine this further. Although cheaper than computing the metric tensor, the gradient computation in these methods is still expensive, and it would be desirable to utilize e.g. the shot-frugal classical shadows approaches summarized in Section 2.1. Since variational dynamics provides a natural warm-start for each optimization step [263], this method also benefits from optimization methods that are particularly more performant than competitors when initialized near a good minimum. Both of these properties align well with the CoVaR method [152], and therefore my novel approach to variational dynamics is as follows. Given a PQC $U(\boldsymbol{\theta})$ and an initial state $|\psi\rangle$ (produced by some sampling circuit $\Psi$ such that $|\psi\rangle = \Psi|0\rangle$), I seek to find parameters $\boldsymbol{\theta}_t$ such that $U(\boldsymbol{\theta}_t)|\psi\rangle \approx e^{-iHt}|\psi\rangle$. That is, $U(\boldsymbol{\theta}_t)$ should approximately reproduce the effects of time evolution *on*

*a specific initial state* $|\psi\rangle$*, including the correct global phase.* This is a stronger requirement than typical variational dynamics schemes [29, 188, 189, 191, 207] due to the global phase requirement, but a much more permissive requirement than full unitary compilation $U(\boldsymbol{\theta}_t) \approx e^{-iHt}$, which would require the effects of time evolution to be reproduced on *all* initial states. Much like Refs. [189, 191], this approach relies upon recompilation of Trotter steps. At $t = 0$, one initializes the parameters $\boldsymbol{\theta}_{t=0}$ such that $U(\boldsymbol{\theta}) = \mathbb{1}$ (typically this is $\boldsymbol{\theta}_{t=0} = \mathbf{0}$). Then, for each subsequent step, given parameters $\boldsymbol{\theta}_t$ at some time $t$, one attempts to find parameters for the next time $t + \Delta t$ such that

$$U(\boldsymbol{\theta}_{t+\Delta t}) |\psi\rangle \approx V(\Delta t) U(\boldsymbol{\theta}_t) |\psi\rangle , \tag{4.42}$$

where $V(\Delta t) \approx e^{-iH\Delta t}$ is a Trotter step over time $\Delta t$, and global phase is accounted for in the (approximate) equality. This can be accomplished by applying the circuit depicted in Figure 4.9(a), and variationally minimizing the compilation Hamiltonian

$$H_{\text{comp}} = -\sum_{j=1}^{n+1} \sigma_j^z \tag{4.43}$$

with respect to $\boldsymbol{\theta}_{t+\Delta t}$. The global phase is accounted for by the $n + 1$th (ancilla) qubit. In cases where global phase is unimportant, one could instead apply the circuit in Figure 4.9(b), which does not require controlled evolution on an ancilla, and thus $H_{\text{comp}} = -\sum_{j=1}^{n} \sigma_j^z$. Therefore, this approach can be directly compared to the Trotter recompilation methods of Refs. [189–192]. The optimization steps of such methods can be relatively costly, relying on either SWAP tests or gradient computation with parameter-shift rules [189–191] which can have highly unfavourable sampling costs.

I circumvent these difficulties by training the circuits with the CoVariance Root finding (CoVaR) approach introduced in Ref. [152]. This allows very shot-frugal optimization of variational dynamics: not only does this not require a SWAP test or expensive gradient computation, but since only covariances of local observables are necessary, each step can be performed simply by measuring a classical shadow [324], necessitating a sample complexity that is merely logarithmic in $n$ for fixed observable

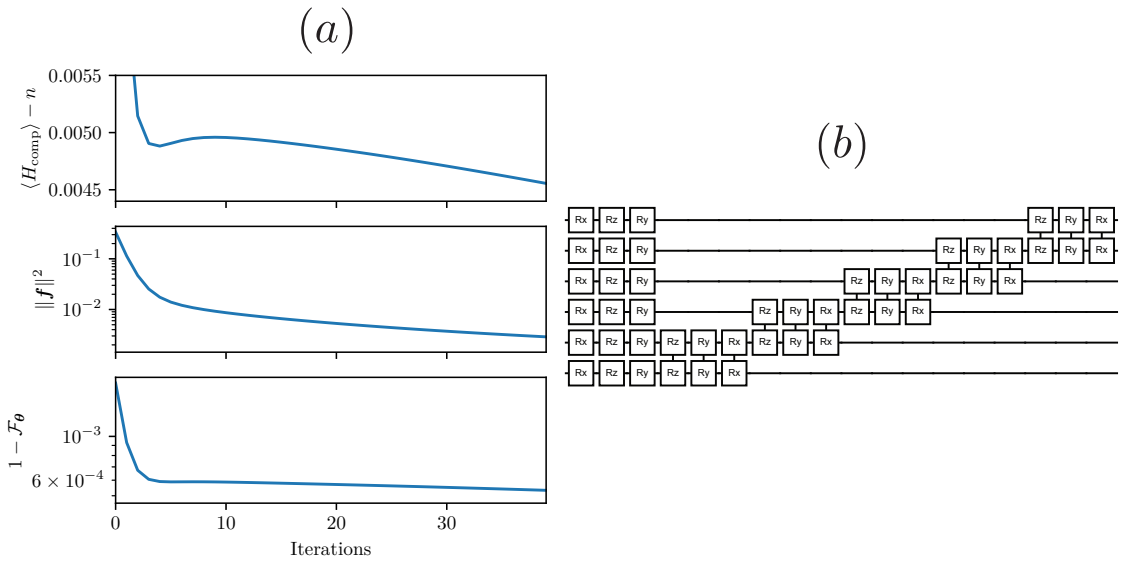



**Figure 4.10: Recompiling Trotter steps with CoVaR. (a)** I recompile a single Trotter step of $\Delta t = 0.2$ for the Heisenberg chain (3.33), with parameters as studied in Section 4.5.2 ($n = 6$ spins, disorder $h = 1$, coupling $J = 1$) and 40 CoVaR steps (one classical shadow required each). I compare the compilation energy $\langle H_{\text{comp}} \rangle$, the covariance norm $\|\boldsymbol{f}\|^2$, and the step infidelity $1 - \mathcal{F}_{\boldsymbol{\theta}}$. The covariance norm $\|\boldsymbol{f}\|^2$ is directly minimized by the method, which leads here to a strict decrease in infidelity $1 - \mathcal{F}_{\boldsymbol{\theta}}$. However, even though it strictly improves fidelity, this optimization takes a path through parameter space that briefly *increases* energy $\langle H_{\text{comp}} \rangle$, which would not occur when training by gradient descent. **(b)** A single layer of the hardware-efficient ansatz used for variational approximation of dynamics in (a) and Section 4.5.2.

locality. Furthermore, since $\Delta t$ will typically be chosen to be small (such that the Trotter approximation holds), $\boldsymbol{\theta}_t$ and $\boldsymbol{\theta}_{t+\Delta t}$ will be close to each other in parameter space. This ensures that CoVaR is initialized in the a vicinity of the solution, where it performs best [152], and also may help circumvent the barren plateaus that plague variational methods in general (Section 2.2.3).

In addition to the shot-frugality of classical shadows and the effectiveness of CoVaR for warm-started problems, this approach has another advantage: it can escape local minima in the optimization landscape that gradient-based methods would get trapped in. In Figure 4.10(a) I compare, for recompilation of a single Trotter step, the compilation 'energy' $\langle H_{\text{comp}} \rangle$, the covariance vector norm $\|\boldsymbol{f}\|^2$ (where $\boldsymbol{f}$ is a vector of covariances $\langle O_j O_k \rangle - \langle O_j \rangle \langle O_k \rangle$ of 3-local observables $O_j$), and the step fidelity $\mathcal{F}_{\boldsymbol{\theta}} \equiv |\langle \psi | U^\dagger(\boldsymbol{\theta}) V(\Delta t) U(\boldsymbol{\theta_t}) | \psi \rangle|^2$, where $V(\Delta t)$ is a second-order Trotter step over time $\Delta t$. The CoVaR method directly minimizes $\|\boldsymbol{f}\|^2$, which also

leads to direct reduction of the infidelity $1 - f_{\boldsymbol{\theta}}$. However, this includes an *increase* in the compilation energy $\langle H_{\text{comp}} \rangle$, taking a path in parameter space that improves fidelity which gradient-based methods would not take due to the energy increase. I note that at large system sizes, the local observables $O_j$ included in $\boldsymbol{f}$ should be chosen randomly at each step [152], but at small system sizes it remains tractable to use all 3-local observables. In Figure 4.10(b), I depict the hardware-efficient ansatz used for variational dynamics throughout this section. Throughout this section, to approximate the dynamics I use this method to recompile second-order Trotter steps of size $\Delta t = 0.2$ using 25 steps, each corresponding to a change of parameters and new classical shadow.

As noted before, this method can be used for global-phase-insensitive recompilation of dynamics (as studied in Refs. [29, 188, 189, 191, 207]) using the circuit structure of Figure 4.9(b). For the purposes of this chapter, I perform the optimization noiselessly on a classical computer, but I anticipate that this approach may be a competitive method of variationally approximating dynamics in general due to the shot-frugal classical shadows approach. At the time of writing this thesis, I have commenced a deeper investigation into this method, which I hope may lead to a further paper.

### 4.5.2 Demonstrating noisy variational DOS estimation

In this section, I utilize the variational dynamics method outlined in the previous section to demonstrate a NISQ-compatible variant of this chapter's techniques. Since the goal is to demonstrate the technique under realistic conditions for current-generation quantum hardware, I now outline a suitable noise model (which I also utilize in Chapter 6 to model noisy preparation of a metrological probe state).

Since this chapter's methods require controlled time-evolution, the overwhelming majority of gates for the circuits of Section 4.5.1 are 2-qubit (from on-site Hamiltonian terms) or 3-qubit (from coupling Hamiltonian terms) gates. This error model burdens these multi-qubit gates with noise as follows. First, I note that although simulating controlled evolution under the Heisenberg Hamiltonian (3.33)

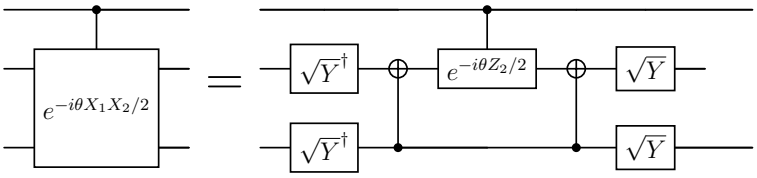



**Figure 4.11: Decomposition of controlled multi-qubit Pauli rotations.** Decomposition of a controlled-Pauli-XX rotation, whereby the entire operation can be controlled by controlling one single-qubit Pauli-Z rotation. The Pauli product is conjugated by single-qubit gates and CNOTs which map it to a single Pauli $Z$ observable, about which a phase rotation is applied. This can be straightforwardly generalized to any controlled multi-qubit Pauli rotation using conjugation by appropriate single-qubit transfomations and CNOTs, but always can be controlled via one single-qubit Pauli-Z rotation.

requires 3-qubit gates (controlled multi-qubit Pauli rotations), these can be easily be decomposed into 1- and 2-qubit gates by the structure outlined in Figure 4.11. Therefore, when applying controlled coupling terms, I apply a 2-qubit error channel between the two target qubits and a (different, pairing-dependent) two-qubit error channel between the ancilla and the nearest target qubit. Since qubit connectivity varies greatly between architectures, I do not explicitly model it here.

Each distinct pair of qubits has a unique error model, in order to capture the effects of differing calibration properties of each qubit. The error models are drawn from a realistic noise model that was learned directly from a 127-qubit IBM Eagle processor in Ref. [52]. The noise models are of the sparse Pauli-Lindblad form outlined in Ref. [272]. Each channel is generated by a site-dependent Lindbladian $\mathcal{L}_m$, where

$$\mathcal{E}_m(\rho) = \exp\left[\mathcal{L}_m\right](\rho), \quad \mathcal{L}_m(\cdot) = \lambda_0 \sum_{k\in\mathcal{K}} \gamma_{km}(P_k \cdot P_k^\dagger - I \cdot I) \tag{4.44}$$

for $\gamma_k \geq 0$, and where $\lambda_0$ is an overall multiplier of the base noise level and $\mathcal{K}$ is the set of 2-qubit Pauli operators for the qubits acted upon by the entangling gate. This corresponds to a diagonal Pauli transfer matrix with diagonal terms $f_j = \prod_{k\in\mathcal{K}} w_k + (1 - w_k)(-1)^{\langle j,k\rangle}$, where $w_k \equiv (1 + e^{-2\lambda\gamma_k})/2$ and $\langle a, b\rangle$ is the binary symplectic product

$$\langle a, b\rangle = \begin{cases} 0 \text{ when } [P_a, P_b] = 0, \\ 1 \text{ when } \{P_a, P_b\} = 0. \end{cases} \tag{4.45}$$

One can easily write this in Kraus operator form

$$\Lambda(\cdot) = \sum_j c_j P_j \cdot P_j^\dagger, \tag{4.46}$$

by the Walsh-Hadamard type transform

$$c_b = \frac{1}{4^n} \sum_a (-1)^{\langle a,b \rangle} f_a. \tag{4.47}$$

In the numerical simulations of this section (and later Chapter 6), I generate the Kraus-operator forms of these noise channels (4.46) for a given noise level $\lambda$ and apply these directly to the density matrix after each entangling gate. Different $\gamma_{km}$ are chosen for each qubit pairing $m$, sampled randomly from a set of amplitudes learned directly from CNOT processes at different sites in a 127-qubit IBM Eagle device [52].

I now use this noise model to illustrate the performance of this chapter's methods using variational techniques in the NISQ era. All Loschmidt echoes used in the following calculation are evaluated under the effect of the above noise model, which is parametrized by a base noise multiplier $\lambda_0$, where $\lambda_0 = 1$ corresponds to noise levels comparable to current-generation hardware. I denote the combined channel containing both the circuit $U(\boldsymbol{\theta})$ and hardware noise as $\mathcal{U}_{\boldsymbol{\theta}} : \mathcal{D}(\mathcal{H}) \to \mathcal{D}(\mathcal{H})$.

For this numerical experiment, I consider dynamics of the Heisenberg model in Equation 3.33 at $n = 6$, for disorder strength $h = 1$ and coupling $J = 1$. This model has been identified as a strong candidate for early quantum advantage [342, 344], and scaled up to much larger system sizes would likely represent a genuinely non-trivial quantum task. I consider 100 initial states $\{|\psi_s\rangle\}_{s=1}^{100}$ sampled randomly from single-qubit rotations (uniform random Euler angles), and using the new variational method of Section 4.5.1, compute approximate angles $\boldsymbol{\theta}_{t,s}$ such that $U(\boldsymbol{\theta}_{t,s}) |\psi_s\rangle \approx e^{-iHt} |\psi_s\rangle$ for each $s$ and relevant time $t$. I consider 60 timesteps of $\Delta t = 0.2$, evaluating each Loschmidt echo with shot noise of $N_s = 200$ shots per time step (100 shots each for real and imaginary parts). For the ansatz circuit, I use the hardware-efficient ansatz depicted in Figure 4.10(b).

In Figure 4.12, I compare this variational evaluation of the DOS for noise rates $\lambda_0 \in \{1, 1/2, 1/10\}$. I compare the approximations to the DOS obtained in

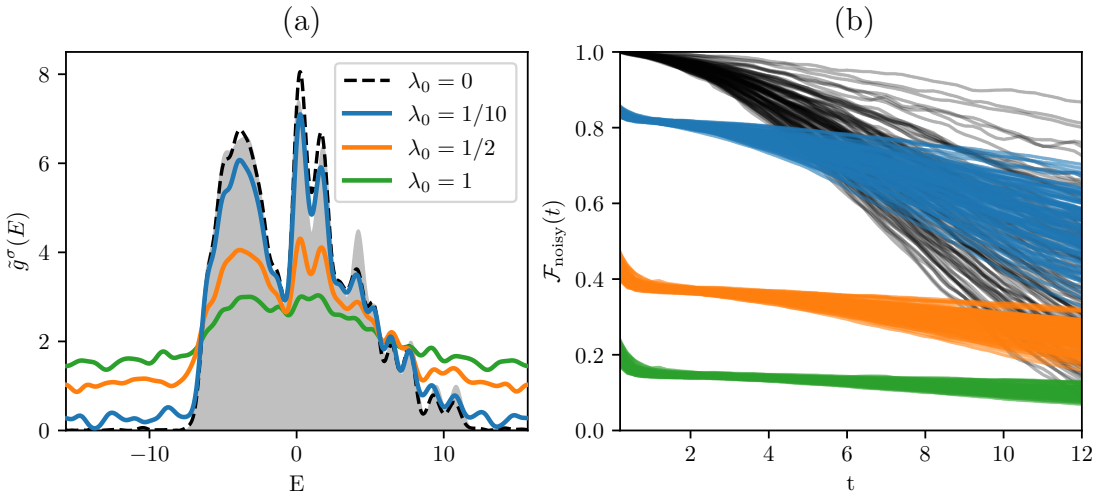



**Figure 4.12: Calculating density of states with noisy variational quantum dynamics. (a)** I demonstrate a NISQ-friendly variant of our methods by calculating the DOS of a Heisenberg model with disordered magnetic fields (3.33) ($n = 6$, $h = 1$, $J = 1$) in the presence of hardware noise, using variational quantum dynamics. I do *not* rescale the Hamiltonian, and use a window width of $\sigma = 2.5$, leading to the ideal DOS depicted by the shaded grey region. 100 randomly sampled initial states are evolved with variational methods and used to estimate $G(t)$ at a per-timestep shot budget of $N_s = 10^4$ ($N_r = 100$), which in the absence of noise replicates the DOS with only minor defects (dotted black line). I compare the DOS obtained at varying overall noise levels $\lambda_0$, resolving the DOS to good fidelity at $\lambda_0 = 1/10$. **(b)** A comparison of the noisy fidelities $\mathcal{F}_{\text{noisy}}(t)$ for the underlying dynamics of the 100 randomly-sampled states, at each noise level.

Figure 4.12(a), finding that although noise comparable to current devices ($\lambda_0 = 1$) is too severe for the requirements of controlled evolution, by $\lambda_0 = 1/10$ one can reproduce a reasonable approximation of the DOS. This suggests that our methods may become more viable on NISQ devices with modestly increasing 2-qubit gate fidelity. In Figure 4.12(b), I plot the state-to-state fidelities

$$\mathcal{F}_{\text{noisy}}(t) \equiv \frac{1}{d} \operatorname{Tr}[\rho_{\text{ideal}}(t)\rho_{\text{actual}}(t)], \tag{4.48}$$

where $\rho_{\text{ideal}}(t) \equiv e^{-iHt} |\psi_s\rangle \langle\psi_s| e^{iHt}$ is the ideal evolved state and $\rho_{\text{actual}}(t) = \mathcal{U}_{\boldsymbol{\theta}_{t,s}}(|\psi_s\rangle \langle\psi_s|)$ is the actual state obtained from the noisy ansatz — indeed, the two most severe noise rates that fail to produce a good DOS have very low fidelities ($\mathcal{F}_{\text{noisy}}(t) < 0.5$).

## 4.6 Conclusions

In this chapter, I introduced a range of new techniques for estimating thermodynamic properties of quantum systems using quantum computers. This is achieved via the DOS, whose Fourier transform (the FDOS) is computed via controlled real-time evolution on a quantum computer. Throughout this chapter, I introduced a framework of new and improved methods to estimate the FDOS. Notably, I introduced a method to compute the DOS not only on the full Hilbert space, but also on Hilbert subspaces, which per Section 4.1.2 is a necessary requirement to compute thermal properties of any fermionic system. Since the constituents of matter are fermions, this is a crucial step: nearly all *ab initio* quantum models of genuine real-world relevance will have a fermionic component. I also introduced deterministic circuits that enable one to *directly* sample the FDOS without resorting to random sampling methods as in Refs. [306–309, 323]. With regard to random sampling, I detailed a broad range of circuits that enable direct sampling of the FDOS, greatly improving on previous unitary 2- and 3-design methods by showing how very simple 1-design initialization circuits could be used. Indeed, even sampling as simple as random bit-flips was shown to be equivalent to more uniform methods when a different circuit was sampled per shot, and I demonstrated the effect of random circuit uniformity on the convergence rate when sampling circuits are repeated. I also detailed how one can directly estimate thermodynamic quantities using these methods via importance sampling of the FDOS.

By simulating this novel approach in a broad range of numerical experiments, I demonstrated its robustness to a number of practical factors. In addition to the aforementioned study of sampling and shot noise, I demonstrated surprising levels of robustness to algorithmic and gate errors that are encountered in early fault-tolerant quantum computers. In the case of algorithmic error, the structure of Trotter error shifts peaks in the DOS by a timestep-dependent amount, meaning that estimation of the DOS up to some commensurate spectral resolution is still robust. In the case of gate error, the effective noise from imperfect $T$ gates leads to an effective additional windowing of the DOS, which results in a broadening of

spectral features rather than complete failure of the method. I then introduced a novel approach to variational dynamics that uses the CoVaR method [152] to train parameters of a variational circuit, and demonstrated how this may enable our methods to be adapted to flawed devices in the NISQ era.

A range of further extensions and generalizations of the work in this chapter are possible. Although I concluded that the random initial state sampling approach (Figure 4.2(b)) has the same sample complexity scaling as the more complex DQC1 circuits in Figure 4.2(a), there may be advantages to using the latter in certain circumstances. As presented in this chapter, both approaches are fundamentally shot-noise-limited. Although the state preparation circuits are simpler in the case of random initial state sampling, in a mature fault-tolerant setting where one can apply very deep controlled operations, it would be possible to use amplitude estimation [19] to asymptotically improve the scaling of the method. While we have presented the work in this chapter in the context of NISQ and early-fault-tolerant systems, it nonetheless represents a strong foundation for future fault-tolerant methods. I anticipate that this target problem will remain compelling well into the fault-tolerant era, as the DOS enables one to probe key thermal equilibrium properties of systems that can be simulated on a quantum computer, and computing it is in general a computationally hard problem [304].

In the nearer term, there are several clear avenues to enhance the algorithm. Compared the the shadow spectroscopy approach I introduced in Chapter 3, implementing the methods of this chapter is more challenging due to the requirement of *controlled* time evolution. This requires more entangling operations, more qubit connectivity, and burdens the ancilla (from which the entire FDOS signal is extracted) with noise from *all* time evolution operations. While this is a common feature of related methods [163, 306–309], there has recently been interest in overcoming this [362, 363]. The DQC1 circuit approach is directly compatible with the techniques of Ref. [362], by which one estimates the phase of the Loschmidt echoes via its imaginary-time logarithm, and this would enable one to estimate the FDOS using only real and imaginary time evolutions (albeit at significantly

increased shot cost). I also note the possibility of introducing novel error mitigation techniques related to the hyperparameters of this chapter's methods. For example, when directly estimating observables via the methods of Section 4.2.4, one could potentially estimate the observable at several different window widths and use extrapolation to estimate the observable's true value at $\sigma \to \infty$. My collaborator on the work of Chapter 3 investigated a similar method for mitigating Trotter errors in shadow spectroscopy, finding it to be highly successful [334].

Throughout this chapter and Chapter 3, when considering application of our methods to the NISQ era, I have utilized ansatz-based variational methods. As outlined in the relevant section of my literature review (Section 2.2.3), there is a great deal of emerging skepticism in the community about the utility of these methods, due to the seemingly fundamental connections between trainability and simulability. The authors of Ref. [134] observe that in every known case where a VQA provably lacks a barren plateau (Definition 1), not only is the problem efficiently classically simulable, but the method of classical simulation relies upon the same mathematical tools used to prove the absence of barren plateaus. One such connection they observe was my own work on Lie-algebraic classical simulations for variational quantum computing, which I present in the following chapter.

***Borat:*** *What type of dog is this?*
***Zookeeper:*** *This is a tortoise.*

— Borat Sagdiyev [364], predicting the outrageous effectiveness of QML for classification

# 5

# $\mathfrak{g}$-sim: Lie-algebraic classical simulations for variational quantum computing

*The research in this chapter was conducted jointly with Martin Larocca, Lukasz Cincio, Marco Cerezo, and Frédéric Sauvage and led to the preprint of Ref. [207], which this chapter closely follows. I led the project as first author, supervised by Fred and Marco. Following a core observation conceived by Marco, Martin and Fred, I significantly expanded upon the analytic framework and methodology, and conducted all investigations myself. Text from the published manuscript is used verbatim in places — in these cases I am the original author of this text.*

## Contents


**5.1 Framework . . . . . 115**
5.1.1 Foundations . . . . . 117
5.1.2 Lie theory . . . . . 119
5.1.3 $\mathfrak{g}$-sim principles . . . . . 122
5.1.4 Generalization to operators not supported by $\mathfrak{g}$ . . . . . 130
**5.2 Guidelines for efficient implementation . . . . . 132**
5.2.1 Efficient evolution . . . . . 133
5.2.2 Sparsity of adjoint representation . . . . . 135
5.2.3 Efficient calculation and eigendecomposition of Pauli-basis adjoint representations . . . . . 137
**5.3 Scalability and demonstration . . . . . 138**
5.3.1 Scalability of $\mathfrak{g}$-sim . . . . . 138
5.3.2 A polynomial-sized Lie algebra . . . . . 140
5.3.3 Example simulation task . . . . . 141

5.3.4 Resource benchmark . . . . . 144
**5.4 Applications . . . . . 146**
5.4.1 Characterizing VQAs . . . . . 147
5.4.2 Pre-training quantum circuits . . . . . 151
5.4.3 Circuit synthesis . . . . . 158
5.4.4 Supervised quantum machine learning . . . . . 170
**5.5 Conclusions . . . . . 174**
5.5.1 Scope for future work . . . . . 176

---

Throughout Chapters 3 and 4, I have utilized variational methods in the hopes of creating NISQ-compatible variants of my algorithms. But the spectre of the trainability-simulability connection (Section 2.2.3) looms large over all efforts in variational quantum computing. This chapter involves a deeper dive into this connection: using the same Lie-algebraic tools that mostly account for the barren plateau phenomenon [213, 225, 227, 228], I construct a framework for classical simulation of VQAs, which I call '$\mathfrak{g}$-sim'. I present a theoretical framework (Section 5.1), present practical guidelines for efficient implementation 5.2, discuss scalability, benchmarking and a novel demonstration exceeding the capabilities of competing methods (Section 5.3), and apply it to a broad variety of paradigmatic tasks in variational quantum computing (Section 5.4). Specifically, I employ $\mathfrak{g}$-sim to train and optimize parametrized quantum circuits, study trainability phenomena at beyond-state-vector-simulation system sizes, design enhanced parameter initialization strategies, solve tasks of quantum circuit synthesis, and train a quantum phase classifier. The results of this chapter were later used as a key ingredient in the trainability-simulability connection famously conjectured in Ref. [134], and form a key part of the ongoing conversations in the community about the foundations of variational quantum computing.

## 5.1 Framework

In Section 2.1, I outlined the role of efficient classical simulations in the development of quantum technology. Between the development of large-scale quantum algorithms, challenging claims of quantum advantage [66, 67, 70–78], techniques to learn from

quantum data [134, 135], randomized benchmarking [117, 118], and training learning-based error mitigation strategies [119–122] it is clear that any novel method for scalable classical simulations will find a practical use. Classical simulations can also teach us about the nature of quantum complexity, an essential pursuit in the study of quantum information. The existence of a scalable classical simulation method typically reveals a distinct way in which a system can be *not* truly quantum, lacking the full exponential complexity of quantum dynamics (whether by a restriction in initial states, operations performed, or observables measured). In terms of the methods already discussed in this thesis, MPS methods reveal the importance of entanglement in quantum complexity [114], stabilizer simulations reveal how even highly-entangled states can lack complexity in the absence of a complete gate set [107, 108], and free-fermion methods demonstrate the importance of hidden symmetries and transformations in the system [109, 110]. One approach to classical simulation that has existed for nearly two decades yet seen little use in the quantum computing community is the role of the underlying Lie-algebraic structure of quantum dynamics [112, 113]. These methods recast quantum dynamics in terms of the underlying DLA $\mathfrak{g}$; since they scale in the dimension of the DLA $\dim(\mathfrak{g})$ rather than the full Hilbert space dimension $2^n$, such simulations become scalable whenever $\dim(\mathfrak{g})$ grows polynomially with the system size $n$ (i.e., $\dim(\mathfrak{g}) \in \mathcal{O}(\text{poly}(n))$). In this section, I present a modernized and extended framework for this approach, which I call $\mathfrak{g}$-sim. Building upon a context for these methods (Section 5.1.1), I first provide a pedagogical review of relevant aspects of Lie theory (Section 5.1.2). I then present the mathematical framework for $\mathfrak{g}$-sim (Section 5.1.3), proving a range of theoretical results. Although the core principles of the framework build upon the well-known results of Somma [112, 113], I further generalize Somma's theoretical results in a number of ways. Furthermore, as the results of Refs. [112, 113] predate much of the quantum computing literature, the work of this chapter also serves to modernize its presentation in line with the current quantum computing community's needs. The theoretical generalizations introduced in this chapter include differentiability of the dynamics for optimization, the ability to simulate

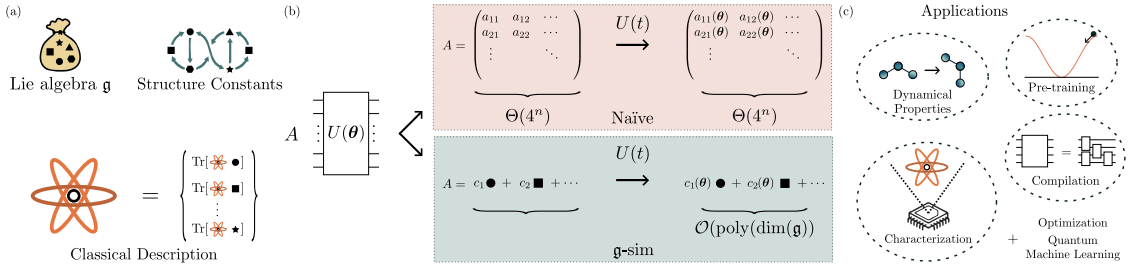



**Figure 5.1: Theoretical framework and applications of $\mathfrak{g}$-sim. (a)** The essential ingredients for $\mathfrak{g}$-sim. First, one needs a description of the Lie algebra $\mathfrak{g}$ associated with the possible unitary dynamics. In this diagram, the basis elements of $\mathfrak{g}$ are depicted as a collection of black geometric shapes. Next, one needs to know how these elements couple to each other under the unitary dynamics. This is determined entirely by the structure constants (5.6). The final requirement is a 'classical description' of the input state, i.e. a vector of expectation values with respect to the basis elements of $\mathfrak{g}$. **(b)** Comparison of naïve and Lie-algebraic approaches to quantum dynamics in the Heisenberg picture. In the naïve approach, an observable $A$ is viewed as a linear operator acting on an $n$-qubit Hilbert space $\mathcal{H}$, with $\dim(\mathcal{H}) = 2^n$. The observable is represented as a $2^n \times 2^n$ matrix whose coefficients evolve according to the von Neumann equation of motion, and thus the approach *always* scales as $\Theta(4^n)$ and is not classically scalable. In the Lie-algebraic approach, the observable is instead viewed as a linear combination of distinct basis terms in $\mathfrak{g}$ (here depicted as black geometric shapes), whose coefficients couple to each other over time via the structure constants. Whenever $\dim(\mathfrak{g}) \in \mathcal{O}(\mathrm{poly}(n))$, $\mathfrak{g}$-sim yields a scalable classical simulation framework. **(c)** Summary of the potential applications of $\mathfrak{g}$-sim. These efficient classical simulations have utility in simulation and optimization of quantum systems, including pre-training of VQA or QML problems, compiling unitary processes to compact quantum circuits, and characterizing QML optimization landscapes.

non-unitary quantum channels, extension to general Lie-group-invariant subspaces, and compatability with arbitrary (mixed) input states. In particular, the latter enables me to conduct a simulation which, to the best of my knowledge, is not possible with any other known simulation method (Section 5.3.3).

I present a high-level overview of the theoretical framework and applications of this chapter in Figure 5.1.

### 5.1.1 Foundations

Throughout this chapter, I consider a qubit system acted upon by quantum circuits (i.e. discrete gate operations). However, this framework straightforwardly generalizes to continuous-time quantum evolution (see Appendices A.5 and B.3 of our full manuscript [207]), as well as to higher-dimensional qudit systems or even infinite-dimension bosonic systems [365].

With this scope in mind, throughout this chapter I will study a system of $n$ qubits with an associated Hilbert space $\mathcal{H} = (\mathbb{C}^2)^{\otimes n}$ of dimension $d = 2^n$. In particular, I consider dynamics determined by unitaries of the form

$$U(\boldsymbol{\theta}) = \prod_{l=1}^{L} \prod_{k=1}^{K} e^{-i\theta_{lk} H_k}, \tag{5.1}$$

where the $\boldsymbol{\theta} = (\theta_{11}, \ldots, \theta_{LK})$ are real-valued classical parameters, and the $H_j$ are the Hermitian *gate generators* of the circuit. This structure is repeated over $L$ layers. While Equation 5.1 is of the layered form common for PQCs in VQA applications, it can also accomodate unitaries with fixed gates (by fixing some of the parameters $\theta_{lk}$) or non-periodic structure (by setting $L = 1$) and essentially allows for arbitrary unitary circuits. It is a helpful representation for two reasons: it is mathematically convenient since it aligns with the frameworks of Refs. [213, 225], and I will predominantly be concerned with applications in VQAs and quantum simulation, where layered circuits are commonplace.

Throughout this chapter I consider tasks of (weak) simulation and optimization of quantum circuits. Given a desired observable $O$, an initial state $\rho^{(\text{in})}$, and a circuit $U(\boldsymbol{\theta})$, *simulation* consists of evaluating the expectation value of $O$ for the evolved state $\rho^{(\text{out})}(\boldsymbol{\theta}) \equiv U(\boldsymbol{\theta})\rho^{(\text{in})}U^\dagger(\boldsymbol{\theta})$. That is, for the purposes of this chapter, 'simulation' refers to evaluating quantities of the form

$$\langle O(\boldsymbol{\theta})\rangle \equiv \text{Tr}\Big[OU(\boldsymbol{\theta})\rho^{(\text{in})}U^\dagger(\boldsymbol{\theta})\Big] = \text{Tr}\Big[O\rho^{(\text{out})}(\boldsymbol{\theta})\Big]. \tag{5.2}$$

In Section 2.2, I detailed how VQA applications require one to *optimize* the system's parameters $\boldsymbol{\theta}$ to minimize a loss function that is typically composed of expectation values of the form in Equation 5.2. The loss function is typically a function of one of more expectation values $O_j$ (e.g. the energy of a Hamiltonian is a linear combination of these) evaluated for a single state (e.g. VQE [138]) or a dataset of multiple states (e.g. QML applications [210]). In most cases, optimization of circuit parameters utilizes gradients of the form $\partial_{\boldsymbol{\theta}}\langle O(\boldsymbol{\theta})\rangle$ [143–145, 366], and therefore the ability to compute these efficiently will also be crucial throughout this chapter.

### 5.1.2 Lie theory

This chapter relies upon the theory of Lie groups and Lie algebras; in this section, I provide a brief review of the relevant concepts and definitions. This introduction is far from comprehensive, and the interested reader could refer to standard introductory textbooks [367, 368], treatments focused on quantum control theory [369, 370], or closely related prior work using Lie-algebraic tools to assess trainability of VQAs [213, 225].

The dynamics and expressiveness of a quantum circuit are characterized by its *dynamical Lie algebra*, defined as follows:

**Definition 2** (Dynamical Lie algebra)**.** *Consider an ansatz of the form in Equation 5.1. Its dynamical Lie algebra* $\mathfrak{g}$ *is defined as the vector space spanned by all the possible nested commutators of* $\{iH_1, \ldots, iH_K\}$*. That is,*

$$\mathfrak{g} = \mathrm{span}_{\mathbb{R}} \langle \{iH_1, \ldots, iH_K\} \rangle_{\mathrm{Lie}} \subseteq \mathfrak{su}(2^n), \tag{5.3}$$

*where* $\langle S \rangle_{\mathrm{Lie}}$ *denotes the Lie closure of S, i.e., the set of elements obtained by repeatedly taking the commutator between elements in S until no new linearly independent elements are obtained. I denote as* $G_\alpha$ *(with* $\alpha = 1, \ldots, \dim(\mathfrak{g})$*) the Hermitian operators such that* $\{iG_\alpha\}_\alpha$ *forms a Schmidt-orthonormal basis of* $\mathfrak{g}$*.*

The dynamical Lie algebra $\mathfrak{g}$ is a Lie subalgebra of the special unitary Lie algebra $\mathfrak{su}(2^n)$, the space of linear anti-Hermitian operators acting on the $n$-qubit system. An observable $A$ is *supported* by the algebra whenever $iA \in \mathfrak{g}$, i.e., whenever $A$ can be expanded in the *basis of observables* $\{G_\alpha\}_\alpha$. In this chapter, I mostly work with algebras whose bases can be constructed analytically (or at least are subalgebras of such algebras), although in general cases one can compute the Lie closure numerically [213, 370].

Using the dynamical Lie algebra, one can construct the extremely useful *dynamical Lie group*, which characterizes the expressiveness of quantum circuits of the form in Equation 5.1, and is defined as follows:

**Definition 3** (Dynamical Lie group)**.** *The dynamical Lie group $\mathcal{G}$ of a circuit of the form Equation 5.1 is determined by its dynamical Lie algebra $\mathfrak{g}$ and defined as*

$$\mathcal{G} = e^{\mathfrak{g}} \equiv \{e^{iA}, \quad iA \in \mathfrak{g}\}. \tag{5.4}$$

Crucially, the dynamical Lie group $\mathcal{G}$ corresponds to all possible unitaries that can be implemented by circuits of the form in Equation 5.1. That is, for every $V \in \mathcal{G}$ there exists (at least) one choice of parameter values $\boldsymbol{\theta}$ for a sufficiently large, but finite, number of layers $L$ such that $U(\boldsymbol{\theta}) = V$ [369]. By its nature, $\mathcal{G}$ contains an uncountably infinite number of elements. However, this can be more manageably parametrized by noting that $\mathfrak{g}$ fully determines the group $\mathcal{G}$ of unitaries that can be realized, and one can therefore define their dimensions in correspondence as $\dim(\mathcal{G})$ (as a smooth differentiable manifold) $= \dim(\mathfrak{g})$. With these objects defined, I note that Heisenberg-like evolution will involve the adjoint action of elements of $\mathcal{G}$ on elements of $\mathfrak{g}$, in the form $O \to U^\dagger O U$ for $O \in i\mathfrak{g}$ and $U \in \mathcal{G}$. The methods of this chapter exploit the fact that the Lie algebra is closed under the adjoint action of its associated Lie group (that is, $U^\dagger O U \in i\mathfrak{g}$), as below.

**Lemma 1** (Action of a group operation)**.** *Suppose $U \in \mathcal{G}$ and $\{iG_\alpha\}_{\alpha=1}^{\dim(\mathfrak{g})}$ forms a Schmidt-orthonormal basis of $\mathfrak{g}$. Then defining $G'_\alpha \equiv U^\dagger G_\alpha U$ for any $G_\alpha$, one has $iG'_\alpha \in \mathfrak{g}$, and thus*

$$G'_\alpha \equiv U^\dagger G_\alpha U = \sum_\beta u_{\alpha\beta} G_\beta, \tag{5.5}$$

*with $u_{\alpha\beta} \in \mathbb{C}$.*

A proof is given in Appendix A.1. Since this chapter's techniques rely upon recasting quantum dynamics in terms of the Lie algebra $\mathfrak{g}$ and the Lie group $\mathcal{G}$, it is not enough to merely have closure: one must also understand how their elements act dynamically (i.e. *what* element of $\mathfrak{g}$ the adjoint action produces). This is characterized how the elements of the algebra transform under commutation with elements of $\mathfrak{g}$ (due to the Baker-Campbell-Hausdorff formula), and under conjugation with elements of $\mathcal{G}$. Since a (matrix) Lie algebra $\mathfrak{g}$ is closed under

nested commutation, its commutation relations are fully characterized by the algebra's *structure constants* $f^{\gamma}_{\alpha\beta} = \mathrm{Tr}[iG_\gamma\,[iG_\alpha, iG_\beta]] \in \mathbb{R}$, through

$$[iG_\alpha, iG_\beta] = \sum_{\gamma=1}^{\dim(\mathfrak{g})} f^{\gamma}_{\alpha\beta} iG_\gamma\,. \tag{5.6}$$

A careful inspection of Equation 5.6 hints at the seed of this chapter's techniques: even though the operators $G_\alpha$ are $2^n \times 2^n$-dimensional matrices, there are less than[1] $\dim(\mathfrak{g})^3$ structure constants that fully characterize the system's dynamics. If $\dim(\mathfrak{g}) \ll 2^n$, then recasting the dynamics in terms of the Lie algebra $\mathfrak{g}$ and its structure constants $f^{\gamma}_{\alpha\beta}$ can be scalable even if the full-Hilbert-space approach is not. Doing so requires the tools of representation theory, and in particular the following definition:

**Definition 4** (Adjoint representation of $\mathfrak{g}$)**.** *Given an operator $iG_l$ in the basis of $\mathfrak{g}$, its adjoint representation is obtained via the map $\Phi^{\mathrm{ad}}_{\mathfrak{g}} : \mathfrak{g} \mapsto \mathbb{R}^{\dim(\mathfrak{g})\times\dim(\mathfrak{g})}$ and is defined by*

$$\left(\Phi^{\mathrm{ad}}_{\mathfrak{g}}(iG_\gamma)\right)_{\alpha\beta} \equiv f^{\gamma}_{\alpha\beta} = \mathrm{Tr}[iG_\gamma\,[iG_\alpha, iG_\beta]]\,. \tag{5.7}$$

If the underlying algebra is compact and the basis observables $G_\alpha$ are Schmidt-orthonormal, then the adjoint representation is faithful [112] (i.e. the map $\Phi_{\mathfrak{g}}$ is injective). The adjoint representation is linear, and therefore Definition 4 implies that knowledge of the adjoint representations of the $\dim(\mathfrak{g})$ basis observables is sufficient to obtain the representations of any element of $\mathfrak{g}$. That is, for any $A = \sum_\alpha (\boldsymbol{w})_\alpha G_\alpha$ (with $\boldsymbol{w} \in \mathbb{R}^{\dim(\mathfrak{g})}$) supported by the algebra, it follows that $\Phi^{\mathrm{ad}}_{\mathfrak{g}}(A) = \sum_\alpha (\boldsymbol{w})_\alpha \Phi^{\mathrm{ad}}_{\mathfrak{g}}(G_\alpha)$.

It is the Lie *group* $\mathcal{G}$ that determines the dynamics, and so this creates a foundation to construct its adjoint representation:

[1]The structure constant tensor $f^{\gamma}_{\alpha\beta}$ is antisymmetric under exchange of $\alpha$ and $\beta$ due to commutator properties.

**Definition 5** (Adjoint representation of $\mathcal{G}$)**.** *The adjoint representation of the dynamical Lie algebra $\mathfrak{g}$ induces the adjoint representation $\Phi_{\mathcal{G}}^{\mathrm{Ad}}$ of the dynamical Lie group $\mathcal{G}$. This representation is a linear map $\Phi_{\mathcal{G}}^{\mathrm{Ad}} : \mathcal{G} \mapsto \mathbb{GL}(\mathbb{R}^{\dim(\mathfrak{g})}) \subset \mathbb{R}^{\dim(\mathfrak{g})\times\dim(\mathfrak{g})}$ from the group $\mathcal{G}$ to the group of invertible linear operators $\mathbb{GL}(\mathbb{R}^{\dim(\mathfrak{g})})$, defined as*

$$\Phi_{\mathcal{G}}^{Ad}(U = e^{iA}) = e^{i\bar{\boldsymbol{A}}}, \tag{5.8}$$

*for all $iA \in \mathfrak{g}$ (or any $U \in \mathcal{G}$), and with $\bar{\boldsymbol{A}} \equiv \Phi_{\mathfrak{g}}^{\mathrm{ad}}(A)$.*

I now show that the adjoint action of a unitary $U \in \mathcal{G}$ on some element of $\mathfrak{g}$ is characterized by the adjoint representation $\Phi_{\mathcal{G}}^{\mathrm{Ad}}$, also known as the Hadamard lemma [365].

**Lemma 2** (Adjoint representation of unitary evolution)**.** *Suppose once again that $U \in \mathcal{G}$, $\{iG_\alpha\}_{\alpha=1}^{\dim(\mathfrak{g})}$ forms a Schmidt-orthonormal basis of $\mathfrak{g}$, and consider results of the group operations defined in Equation 5.5. Then the corresponding matrix elements $u_{\alpha\beta}$ are those of the adjoint representation of $U$,*

$$u_{\alpha\beta} = \left(\Phi_{\mathcal{G}}^{\mathrm{Ad}}(U)\right)_{\alpha\beta}. \tag{5.9}$$

A proof is given in Appendix A.2. As I detail in the following section, these adjoint representations therefore allow one to perform (Heisenberg) evolution of any operator in the algebra under the action of any $U \in \mathcal{G}$ with a time complexity scaling with $\mathcal{O}(\mathrm{poly}(\dim(\mathfrak{g})))$ as opposed to $\Theta(4^n)$.

### 5.1.3 $\mathfrak{g}$-sim principles

In this section I present the main results which comprise the foundation of $\mathfrak{g}$-sim including simulations with observables supported by $\mathfrak{g}$ (Section 5.1.3.1), simulations with observables that are products of terms supported by $\mathfrak{g}$ (Section 5.1.3.2), gradient-based optimization (Section 5.1.3.3), and the simulation of non-unitary quantum channels (Section 5.1.3.4). Later sections will extend this framework to more general Lie-group-invariant subspaces and circuit compilation cost functions.

#### 5.1.3.1 Simulation with observables supported by $\mathfrak{g}$

I begin by considering simulation problems where the observable of interest is supported by the Lie algebra: that is, the evaluation of Equation 5.2 when $O \in i\mathfrak{g}$. Here, it is convenient to define $\dim(\mathfrak{g})$-dimensional vectors of expectation values that captures the description of the states $\rho^{(\text{in/out})}$ over the basis observables $G_\alpha$:

$$\left(\boldsymbol{e}^{(\text{in/out})}\right)_\alpha \equiv \text{Tr}\left[G_\alpha \rho^{(\text{in/out})}\right]. \tag{5.10}$$

I refer to these vectors $\boldsymbol{e}^{(\text{in/out})}$ as *classical representations* of the state $\rho^{(\text{in/out})}$. These classical representations evolve as follows.

**Result 2** (Simulation of observables in the algebra)**.** *Consider a circuit of the form in Equation 5.1, and let $O$ be an observable with support in $i\mathfrak{g}$ such that $O = \sum_\alpha (\boldsymbol{w})_\alpha G_\alpha$ for $\boldsymbol{w} \in \mathbb{R}^{\dim(\mathfrak{g})}$. Then, given an input classical representation $\boldsymbol{e}^{(\text{in})}$, one can compute*

$$\langle O(\boldsymbol{\theta}) \rangle = \boldsymbol{w} \cdot \boldsymbol{e}^{(\text{out})}, \tag{5.11}$$

*with the output classical representation obtained as*

$$\boldsymbol{e}^{(\text{out})} = \left( \prod_{l=1}^{L} \prod_{k=1}^{K} e^{-i\theta_{lk} \bar{\boldsymbol{H}}_k} \right) \cdot \boldsymbol{e}^{(\text{in})}, \tag{5.12}$$

*where $\bar{\boldsymbol{H}}_k \equiv \Phi_{\mathfrak{g}}^{\text{ad}}(H_k)$.*

*Proof.* Result 2 follows trivially from Lemmas 1 and 2. □

I note that while Result 2 follows the approach of Somma [112, 113] in spirit and proof method, it differs subtly in that it allows for *arbitrary* initial states for which the classical representation is known, whereas the results of Ref. [112] only accommodate the highest-weight states of the algebra as initial states. Result 2 indicates that, in order to compute $\langle O(\boldsymbol{\theta}) \rangle$, it suffices to have a decomposition of $O$ in the algebra, the adjoint representations $\bar{\boldsymbol{H}}_k$ of the gate generators, and the vector of expectation values of the basis observables $\{G_\alpha\}_\alpha$ for the input state. I stress that although Equation 5.12 strongly resembles unitary evolution in the Hilbert space $\mathcal{H}$, it differs subtly. The classical representation vectors $\boldsymbol{e}^{(\text{in/out})}$ are not Hilbert-space

state vectors in the usual sense, but vectors of expectation values of observables, and consequently the phases (signs) are indeed physical. Since the $\bar{\boldsymbol{H}}_k$ are purely imaginary Hermitian matrices, the gate representations $e^{-i\theta_m \bar{\boldsymbol{H}}_k}$ are real-valued, and describe linear coupling between observables induced by unitary evolution.

An immediate consequence of Result 2 is that one could compute expectation values of observables supported by the algebra with a time complexity scaling as $\mathcal{O}(LK\dim(\mathfrak{g})^3)$. However, using the novel guidelines for efficiency presented later in Section 5.2.1, I was able to reduce the asymptotic complexity required by a factor of $\dim(\mathfrak{g})$ compared to Refs. [112, 113]. This leads to my first main theoretical contribution, which improves on the time complexity of $\mathfrak{g}$-sim.

**Theorem 2.** *Computing expectation values of observables supported by the algebra using* $\mathfrak{g}$-sim *has a time complexity in* $\mathcal{O}(LK\dim(\mathfrak{g})^2)$ *for circuits of the form in Equation 5.1.*

*Proof.* The proof follows constructively from Algorithm 1 (presented later in Section 5.2.1). □

#### 5.1.3.2 Simulation with products of observables supported by $\mathfrak{g}$

Here, I consider the task of simulating the expectation values of products of observables supported by the Lie algebra, and first focus on simulating correlators of the form $\langle O^{(1)}O^{(2)}(\boldsymbol{\theta})\rangle = \mathrm{Tr}\Big[OU(\boldsymbol{\theta})\rho^{(\mathrm{in})}U^\dagger(\boldsymbol{\theta})\Big]$ for $O^{(1)}, O^{(2)} \in i\mathfrak{g}$, even when $O^{(1)}O^{(2)} \notin \mathfrak{g}$. For these, I define a $(\dim(\mathfrak{g}) \times \dim(\mathfrak{g}))$-dimensional matrix of expectation values (a second-order classical representation) that captures the description of the states $\rho^{(\mathrm{in/out})}$ over products of the basis observables

$$(\boldsymbol{E}^{(\mathrm{in/out})})_{\alpha\beta} \equiv \mathrm{Tr}\Big[G_\alpha G_\beta \rho^{(\mathrm{in/out})}\Big]. \tag{5.13}$$

These second-order classical representations evolve according to the following result.

**Result 3** (Simulation of product of two operators in the algebra)**.** *Consider a circuit of the form in Equation 5.1, and let* $O^{(1)}$ *and* $O^{(2)}$ *be observables with support in*

*$i\mathfrak{g}$ such that $O^{(1)} = \sum_\alpha (\boldsymbol{w}^{(1)})_\alpha G_\alpha$ and $O^{(2)} = \sum_\beta (\boldsymbol{w}^{(2)})_\beta G_\beta$ for $\boldsymbol{w}^{(1)}, \boldsymbol{w}^{(2)} \in \mathbb{R}^{\dim(\mathfrak{g})}$. Then, given $\boldsymbol{E}^{(\text{in})}$, one can compute*

$$\langle O^{(1)} O^{(2)}(\boldsymbol{\theta}) \rangle = (\boldsymbol{w}^{(1)})^T \cdot \boldsymbol{E}^{(\text{out})} \cdot \boldsymbol{w}^{(2)}, \tag{5.14}$$

*with the superscript $T$ denoting the matrix transpose, and with the matrix of output expectation values obtained as*

$$\boldsymbol{E}^{(\text{out})} = \left( \prod_{l=1}^{L} \prod_{k=1}^{K} e^{-i\theta_{lk} \bar{\boldsymbol{H}}_k} \right) \boldsymbol{E}^{(\text{in})} \left( \prod_{l=1}^{L} \prod_{k=1}^{K} e^{-i\theta_{lk} \bar{\boldsymbol{H}}_k} \right)^T. \tag{5.15}$$

A proof is given in Appendix A.3. I note once again that while this result is similar to that of Ref. [112, 113], I have again constructed it in such a way that it accommodates arbitrary initial states with a known classical representation. Combining Results 2 and 3 we can simulate expectation values of observables of the form

$$O = \sum_{\alpha=1}^{\dim(\mathfrak{g})} (\boldsymbol{w}^{(1)})_\alpha G_\alpha + \sum_{\beta,\gamma=1}^{\dim(\mathfrak{g})} (\boldsymbol{W}^{(2)})_{\beta\gamma} G_\beta G_\gamma \tag{5.16}$$

for $\boldsymbol{w}^{(1)} \in \mathbb{R}^{\dim(\mathfrak{g})}$ and $\boldsymbol{W}^{(2)} \in \mathbb{R}^{\dim(\mathfrak{g}) \times \dim(\mathfrak{g})}$. In Section 5.2.1, will I show how matrix exponentiation can be removed as the computational bottleneck. Thus, at this point it is worth mentioning that evolution of observables (Result 2) involves only matrix-vector multiplication, whilst evolution of correlators (Result 3) is more expensive as it requires matrix-matrix multiplication. This can be made precise by the following corollary.

**Corollary 1.** *Computing expectation values of products of observables supported by the algebra using* $\mathfrak{g}$-sim *has a time complexity in $\mathcal{O}(LK \dim(\mathfrak{g})^\omega)$, where $\omega \leq 2.371552$ is the matrix multiplication constant [337].*

As noted in Chapter 3, for practical implementations matrix multiplication will be closer to the naïve algorithm ($\omega = 3$), since substantially better asymptotic complexities are obtained via galactic algorithms. Going further, $\mathfrak{g}$-sim can be extended to compute expectation values of correlators of any order (i.e., $M$-th order correlator $O^{(1)} \dots O^{(M)}$). However, simulating an $M$-th order correlator for an arbitrary initial state involves an $\mathcal{O}(\dim(\mathfrak{g})^{2M})$ tensor contraction that becomes impractical for large-order $M$, as noted below.

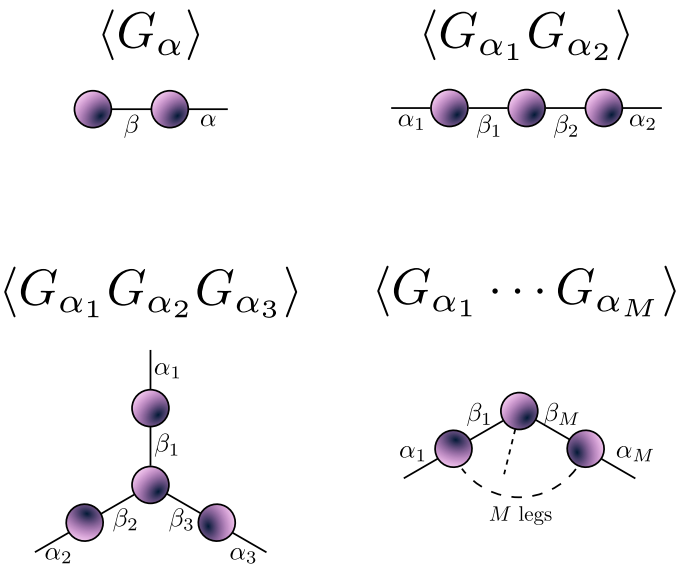



**Figure 5.2: Tensor contractions for $\mathfrak{g}$-sim evolution.** Tensor diagrams for the contractions required to compute dynamics with $\mathfrak{g}$-sim at first order $\langle G_\alpha \rangle$, second order $\langle G_{\alpha_1} G_{\alpha_2} \rangle$, third order $\langle G_{\alpha_1} G_{\alpha_2} G_{\alpha_3} \rangle$, and arbitrary $M$-th order $\langle G_{\alpha_1} \dots G_{\alpha_M} \rangle$.

**Remark 1** (Generalization to higher-order correlators)**.** *By following the approach of the proof of Result 3 (taking the correlator expectation value, expanding the channel, using trace cyclicity and inserting identities, applying Lemma 2), these results generalize straightforwardly to $M$th order correlators $\langle G_{\alpha_1} \dots G_{\alpha_M} \rangle$. The evolution requires contraction of an $M$-th order tensor as depicted in Fig. 5.2, which is naïvely $\mathcal{O}(\dim(\mathfrak{g})^{2M})$. Thus, despite being classically efficient up to fixed $M$, computing high-order correlators quickly becomes impractical as $M$ increases. For this chapter's applications, I require only $M \leq 2$.*

However, I note that for certain initial states, the evaluation of higher-order correlators can be simplified. Given an initial generalized coherent state (highest-weight state of $\mathfrak{g}$), one may factorize the $M$-th order correlator $\langle G_{i_1} \dots G_{i_M} \rangle$ using Wick's theorem, resulting in a highly simplified matrix product form for the correlator tensor that can be evaluated more efficiently. This method is outlined in Appendix F of Ref. [112].

#### 5.1.3.3 Evaluation of gradients

As discussed in Section 2.2, in addition to simulating parametrized quantum circuits, in a variational quantum computing setting one aims to *optimize* the parameters to minimize a loss function. In order to benefit from gradient-based training schemes one needs to efficiently compute derivatives of observable expectation values with respect to the circuit parameters.

Since these techniques were originally envisioned for simulating fixed unitary dynamics and not for their optimization, there were no existing methods using $\mathfrak{g}$-sim-like methods to compute partial derivatives. However, the evolution in Equation 5.12 can be easily differentiated analytically. Naïvely, this would lead to a time complexity of $\mathcal{O}(L^2 K \dim(\mathfrak{g})^3)$ to compute the gradient of an observable, which although polynomially scaling would be sufficiently complicated to preclude many of the applications in Section 5.4. However, using the refined approaches for efficient implementation in Section 5.2.1, the complexity can be reduced:

**Theorem 3** (Gradient calculations in $\mathfrak{g}$-sim)**.** *Computing the gradient of the expectation value of an observable supported by the algebra using* $\mathfrak{g}$-sim *has a time complexity in* $\mathcal{O}(LK \dim(\mathfrak{g})^2)$ *for circuits of the form in Equation 5.1.*

*Proof.* The proof follows constructively from Algorithm 2 (presented later in Section 5.2.1). □

#### 5.1.3.4 Simulation of non-unitary quantum channels

Given that much of variational quantum computing is aimed at the NISQ era, the ability to include noise in $\mathfrak{g}$-sim simulations is extremely desirable. In the treatment I have given thus far, the framework has been entirely unitary, greatly limiting the types of error that can be captured. One possibility is variation in the gate parameters $\theta_{lk}$ in Equation 5.1: over- and under-rotations can be captured by perturbing the angles $\theta_{lk} \to \theta_{lk} + \epsilon_{lk}$, and fluctuations or uncertainties in gate rotations can be captured by averaging over realizations of this. Nonetheless, a useful model of noise in quantum computers will typically require the simulation

of non-unitary quantum channels, which I bring within the scope of $\mathfrak{g}$-sim in this section. Although a straightforward extension of $\mathfrak{g}$-sim to mixed-unitary channels composed of unitaries in $\mathcal{G}$ is obvious (one could simply replace $U \cdot U^\dagger$ with $\sum_m c_m U_m \cdot U_m^\dagger$ in Equation 5.5 and all proofs would still hold), this restricts the available noise channels to those generated by $\mathfrak{g}$, which is highly restrictive for most $\dim(\mathfrak{g}) \in \text{poly}(n)$ algebras. A less obvious, but much more general construction is possible. Its construction is based on the normalizer of $\mathfrak{g}$, defined as follows:

**Definition 6** (Normalizer of $\mathfrak{g}$)**.** *Given a DLA $\mathfrak{g}$, its normalizer $N(\mathfrak{g})$ is is the set of all operators that leave the subspace invariant under conjugation:*

$$N(\mathfrak{g}) \equiv \{A \in \mathcal{D}(\mathcal{H}) \,|\, A^\dagger X A \in \mathfrak{g},\, \forall X \in \mathfrak{g}\}, \tag{5.17}$$

*where $\mathcal{D}(\mathcal{H})$ is the set of bounded linear operators on the Hilbert space $\mathcal{H}$.*

It follows trivially from Lemma 1 that $\mathcal{G} \subset N(\mathfrak{g})$, but there can (and typically will) be elements of of the normalizer $N(\mathfrak{g})$ that are not in $\mathcal{G}$, which enables simulation of more general processes than just mixed-unitary channels composed of elements of $\mathcal{G}$. Consider now a noisy channel form $\tilde{\Lambda}$ of the quantum circuit in Equation 5.1, where each gate is followed by a (possibly unique) noise channel

$$\tilde{\Lambda} = \bigcirc_m (\tilde{\Lambda}_m \circ \mathcal{U}_m), \quad \text{with } \mathcal{U}_m(\cdot) := U_m \cdot U_m^\dagger \quad \text{and } U_m = e^{-i\theta_m H_m}. \tag{5.18}$$

where I have replaced the layer and generator indices with a combined index $m$ for notational clarity. Each noise channel can be written in the form

$$\tilde{\Lambda}_m(\cdot) = \sum_s E_{s,m} \cdot E_{s,m}^\dagger, \tag{5.19}$$

where the $E_{s,m}$ are the Kraus operators of the channel and thus $\sum_s E_{s,m}^\dagger E_{s,m} = \mathbb{1}$. In the case where all $E_{s,m} \in N(\mathfrak{g})$, then $\mathfrak{g}$ has closure under the action of each $\tilde{\Lambda}_m$ and thus the full noisy channel $\tilde{\Lambda}$. Therefore, its conjugation effects can be entirely captured by an adjoint representation:

**Definition 7** (Adjoint representation of $N(\mathfrak{g})$)**.** *Given a normalizer element $A \in N(\mathfrak{g})$, its adjoint representation is obtained via the map $\Phi^{\mathrm{Ad}}_{N(\mathfrak{g})} : N(\mathfrak{g}) \mapsto \mathbb{R}^{\dim(\mathfrak{g})\times\dim(\mathfrak{g})}$ and is defined by*

$$\left(\Phi^{\mathrm{Ad}}_{N(\mathfrak{g})}(A)\right)_{\alpha\beta} \equiv \mathrm{Tr}\Big[G_\alpha A^\dagger G_\beta A\Big]. \tag{5.20}$$

By linearity in Equation 5.20, it follows trivially that $\Phi^{\mathrm{Ad}}_{N(\mathfrak{g})}(\tilde{\Lambda}_m) = \sum_s \Phi^{\mathrm{Ad}}_{N(\mathfrak{g})}(E_{s,m})$. These ingredients are sufficient to simulate the noisy channel $\tilde{\Lambda}$ using $\mathfrak{g}$-sim, as detailed in the following theorem.

**Theorem 4** (Noisy simulation of observables in the algebra)**.** *Consider a noisy circuit of the form in Equation 5.18, for which all Kraus operators $E_{s,m}$ are in the normalizer $N(\mathfrak{g})$. Under the application of this circuit, the classical representation $\boldsymbol{e}^{(\mathrm{in})}$ evolves as*

$$\boldsymbol{e}^{(\mathrm{out})} = \left(\prod_m \Phi^{\mathrm{Ad}}_{N(\mathfrak{g})}(\tilde{\Lambda}_m) e^{-i\theta_m \bar{\boldsymbol{H}}_m}\right) \cdot \boldsymbol{e}^{(\mathrm{in})}, \tag{5.21}$$

*where $\bar{\boldsymbol{H}}_m = \Phi^{\mathrm{ad}}_{\mathfrak{g}}(H_m)$ and $\Phi^{\mathrm{Ad}}_{N(\mathfrak{g})}(\tilde{\Lambda}_m) = \sum_s \Phi^{\mathrm{Ad}}_{N(\mathfrak{g})}(E_{s,m})$.*

*Proof.* Consider the circuit gate-by-gate. The action of the unitary component $e^{-i\theta_m \bar{\boldsymbol{H}}_m}$ follows from Theorem 2, so it suffices to consider the action of the noise channel $\tilde{\Lambda}_m$. Taking $\rho$ to be the state immediately prior to applying the noise channel, the $\alpha$-th component of the classical representation immediately after applying the noise channel is

$$(\boldsymbol{e})_\alpha \equiv \mathrm{Tr}\Big[\tilde{\Lambda}_m(\rho) G_\alpha\Big] = \sum_s \mathrm{Tr}\Big[\rho E^\dagger_{s,m} G_\alpha E_{s,m}\Big], \tag{5.22}$$

where I have used linearity and trace cyclicity after expanding the channel in its Kraus form (5.19). By assumption $E_{s,m} \in N(\mathfrak{g})$, therefore it follows that $E_{s,m} G_\alpha E^\dagger_{s,m} = \sum_\beta d_{\alpha\beta,s} G_\beta$ for some coefficients $d_{\alpha\beta,s}$. Identifying these coefficients via the adjoint representation (5.20) as $d_{\alpha\beta,s} = \mathrm{Tr}\Big[G_\beta E_{s,m} G_\alpha E^\dagger_{s,m}\Big] = \left(\Phi^{\mathrm{Ad}}_{N(\mathfrak{g})}(E_{s,m})\right)_{\alpha\beta}$, the action of the per-gate noise channel $\tilde{\Lambda}_m$ follows linearly. The full noisy circuit evolution is then obtained by composing the per-gate evolutions over $m$, yielding Equation 5.21 and concluding the proof. □

Result 3 and Theorem 3 also generalize straightforwardly in this manner, allowing one to compute correlators and gradients under noisy evolution. At this point, the question arises of which Kraus operators will be contained in $N(\mathfrak{g})$, and whether their adjoint representations (5.20) can be efficiently computed. Many algebras of interest admit a Pauli basis (a full classification is given in Ref. [371]), which fortunately leads to the following corollary.

**Corollary 2** (Simulation of arbitrary Pauli noise channels)**.** *Suppose the chosen basis of* $\mathfrak{g}$ *is a Pauli basis (i.e. the* $G_\alpha$ *are Pauli strings) and* $\dim(\mathfrak{g}) \in \text{poly}(n)$*. Then one can efficiently simulate any Pauli noise channel in* $\mathfrak{g}$-sim*.*

*Proof.* Conjugating an $n$-qubit Pauli operator by another $n$-qubit Pauli operator results in a sign flip at most, satisfying the requirements of Theorem 4. Furthermore, the adjoint representations (5.20) of the Kraus operators admit a $\text{poly}(n)$ number of elements, each of which can be computed efficiently according to the Gottesman-Knill theorem [107, 108]. □

As noted previously, a very general class of noise processes can be twirled to this form [340, 341]. Indeed, Pauli noise channels have proven sufficiently descriptive for real NISQ-era quantum devices that they have enabled highly effective error mitigation techniques [52, 272].

### 5.1.4 Generalization to operators not supported by $\mathfrak{g}$

The presentation of $\mathfrak{g}$-sim in this chapter has thus far been restricted to the simulation of observables supported by $\mathfrak{g}$ and products thereof. More generally, one might wish to include observables outside of the algebra entirely. This seems, at first, too lofty a goal: computing the action $A \to UAU^\dagger$, even for $U \in \mathcal{G}$ where the underlying $\dim(\mathfrak{g}) \in \text{poly}(n)$, would naively scale as $\mathcal{O}(2^{2n})$. Yet, some further generalization can be achieved by exploiting the structure of $\mathcal{G}$ to partition operator space into manageable, smaller invariant subspaces, per the following definition.

**Definition 8** ($\mathcal{G}$-invariant subspace)**.** *Given an operator space $\mathcal{L}$, a subspace $\mathcal{L}_\lambda \subset \mathcal{L}$ is said to be $\mathcal{G}$-invariant if*

$$UAU^\dagger \in \mathcal{L}_\lambda, \forall A \in \mathcal{L}_\lambda \text{ and } \forall V \in \mathcal{G}. \tag{5.23}$$

*Furthermore, $\mathcal{L}_\lambda$ is said to be irreducible (thus dubbed an 'irrep') when there exists no non-trivial $\mathcal{L}_\mu \subset \mathcal{L}_\lambda$ that is also $\mathcal{G}$-invariant.*

Since, for the purposes of this thesis, $\mathcal{G}$ is a compact group acting on a finite dimensional space, it follows that operator space can be decomposed into a direct sum of irreps, as

$$\mathcal{L} \cong \bigoplus_\lambda \mathcal{L}_\lambda. \tag{5.24}$$

If one has $\dim(\mathfrak{g}) \in \text{poly}(n)$, and an irrep with dimension $\dim(\mathcal{L}_\lambda) \in \text{poly}(n)$, then one can efficiently simulate the action of $U \in \mathcal{G}$ on observables in $\mathcal{L}_\lambda$. This is formalized in the following theorem.

**Theorem 5.** *Let $\{B_\alpha^{(\lambda)}\}_{\alpha=1}^{\dim(\mathcal{L}_\lambda)}$ be a Schmidt-orthnormal basis of $\mathcal{L}_\lambda$. Let $\Phi_\lambda^{\text{ad}}$ be a generalization of the adjoint representation* (5.7) *to $\mathcal{G}$-invariant subspaces, whereby*

$$\left(\Phi_\lambda^{\text{ad}}(iG_\gamma)\right)_{\alpha\beta} \equiv f_{\alpha\beta}^{\lambda\gamma} = \text{Tr}\left[iB_\beta^{(\lambda)}\left[iG_\gamma, iB_\alpha^{(\lambda)}\right]\right]. \tag{5.25}$$

*Then the corresponding classical representation ($\boldsymbol{e}_\lambda^{\text{in/out}}$) evolves under the action of $U \in \mathcal{G}$ as*

$$\boldsymbol{e}_\lambda^{\text{out}} = \exp\left(i\Phi_\lambda^{\text{ad}}(K)\right) \cdot \boldsymbol{e}_\lambda^{\text{in}}, \tag{5.26}$$

*where $U = e^{iK}$ for some Hermitian generator $K$.*

We present a proof in the recent revision of our manuscript [207]. All results presented in this chapter can be generalized from observables supported by $\mathfrak{g}$ to those supported by $\mathcal{G}$-invariant subspaces in this manner. For example, noisy simulation in the manner of Theorem 4 can be extended to the normalizers of any (tractable dimension) $\mathcal{L}_\lambda$. We have recently completed a full revision of our paper that reframes all of these results in terms of general $\mathcal{G}$-invariant subspaces; regrettably, this was completed too recently to be fully included in the scope of this thesis, and

the remainder of this chapter will follow the treatment of our original manuscript. Nonetheless, the extension of $\mathfrak{g}$-sim to general $\mathcal{G}$-invariant subspaces greatly expands the class of observables that can be studied, significantly maturing our Lie-algebraic simulation framework between the seminal work of Somma [112, 113].

## 5.2 Guidelines for efficient implementation

The $\mathfrak{g}$-sim framework outlined in Section 5.1 allows scalable classical simulations of noisy and noiseless circuits for a variety of classes of observables and their gradients, whenever $\dim(\mathfrak{g}) \in \mathcal{O}(\text{poly}(n))$. While I have already presented a variety of generalizations upon the work of Somma [112, 113], another contribution of this chapter is to improve upon the prior asymptotic complexity as well as provide guidelines for practical computer implementation. The main improvements detailed in this section are as follows:

- Pre-computing eigendecompositions of gate generator adjoint representations reduces the time complexity of $\mathfrak{g}$-sim evolution from $\mathcal{O}(\dim(\mathfrak{g})^3)$ to $\mathcal{O}(\dim(\mathfrak{g})^2)$. Further combining this with reverse-mode differentiation [372] reduces the time complexity of gradient calculations from $\mathcal{O}(L^2K^2\dim(\mathfrak{g})^3)$ to $\mathcal{O}(LK\dim(\mathfrak{g})^2)$ (5.2.1).

- Relevant matrices are extremely sparse when $\mathfrak{g}$ has a Pauli basis, allowing speedups by use of sparse methods, even for subalgebras whose generators are not sparse (5.2.2).

- Eigendecompositions of gates with Pauli generators can be computed in just $\mathcal{O}(\dim(\mathfrak{g}))$ in the representation of a Pauli basis, instead of the usual $\mathcal{O}(\dim(\mathfrak{g})^3)$ (5.2.3).

In practice, I found that these optimizations significantly accelerated my simulations; without them, most of the applications of Section 5.4 would not have been possible.

### 5.2.1 Efficient evolution

#### 5.2.1.1 Observables

Evaluating each of the $LK$ terms $e^{-i\theta\bar{\boldsymbol{H}}_k}$ appearing in Eq. (5.12) of the simulation routine has time complexity scaling as $\mathcal{O}(\dim(\mathfrak{g})^3)$ using standard matrix exponentiation routines (assuming the matrix multiplication constant $\omega = 3$). However, given the full-rank eigendecompositions

$$\bar{\boldsymbol{H}}_k = \bar{\boldsymbol{Q}}_k \mathrm{diag}(\epsilon_k^{(1)}, \ldots, \epsilon_k^{(\dim(\mathfrak{g}))}) \bar{\boldsymbol{Q}}_k^T \tag{5.27}$$

of the (adjoint representation of the) gate generators $\bar{\boldsymbol{H}}_k$, the action of these parametrized gates is given as

$$\Phi_{\mathcal{G}}^{\mathrm{Ad}}(e^{-i\theta H_k}) = \bar{\boldsymbol{Q}}_k \mathrm{diag}(e^{-i\theta\epsilon_k^{(1)}}, \ldots, e^{-i\theta\epsilon_k^{(\dim(\mathfrak{g}))}}) \bar{\boldsymbol{Q}}_k^T, \tag{5.28}$$

which can be applied to a vector $\boldsymbol{e} \in \mathbb{R}^{\dim(\mathfrak{g})}$ with $\mathcal{O}(\dim(\mathfrak{g})^2)$ time complexity. Notably, these decompositions only need to be performed *once* per algebra and per gate generator used, the number of which is typically no more than linear in the system size $n$. Pre-computing these eigendecompositions naïvely scales as $\mathcal{O}(\dim(\mathfrak{g})^3)$, however, in Section 5.2.3 I detail a procedure by which the eigendecompositions for $\mathfrak{g}_0$ can be computed in just $\mathcal{O}(\dim(\mathfrak{g}))$. I provide an explicit algorithm for simulation incorporating these pre-computed eigendecompositions in Algorithm 1.

#### 5.2.1.2 Gradients

In Section 5.1.3.3, I highlighted the importance of computing gradients of expectation values with respect to the circuit parameters using $\mathfrak{g}$-sim. Here, I improve upon the time complexity of doing so. Differentiating Equation (5.11) with respect to $\theta_m$, one finds

$$(\boldsymbol{\nabla}_{\boldsymbol{\theta}}\langle O\rangle)_m = \frac{\partial\langle O\rangle}{\partial\theta_m} = \boldsymbol{w}^T \left(\frac{\partial\Phi_{\mathcal{G}}^{\mathrm{Ad}}(U(\boldsymbol{\theta}))}{\partial\theta_m}\right) \boldsymbol{e}^{(\mathrm{in})}, \tag{5.29}$$

where one may compute

$$\frac{\partial\Phi_{\mathcal{G}}^{\mathrm{Ad}}(U(\boldsymbol{\theta}))}{\partial\theta_m} = -i\left(\bar{\boldsymbol{U}}_{m+1:M}\bar{\boldsymbol{H}}_k\bar{\boldsymbol{U}}_{1:m}\right), \tag{5.30}$$

**Algorithm 1** Efficient evolution of observable expectation values using $\mathfrak{g}$-sim.

**Input:** Circuit parameters $\boldsymbol{\theta}$, input observables $\boldsymbol{e}^{(\text{in})}$, circuit generator eigendecompositions (Eq. (5.28)) $\bar{\boldsymbol{Q}}_k$ and $\epsilon_k^{(g)}$
  $\boldsymbol{e}^{(\text{out})} \leftarrow \boldsymbol{e}^{(\text{in})}$
  **for** $l$ in $1, \ldots, L$ **do**
    **for** $m$ in $1, \ldots, M$ **do**
      $\boldsymbol{e}^{(\text{out})} \leftarrow \bar{\boldsymbol{Q}}_k^T \boldsymbol{e}^{(\text{out})}$
      **for** $g$ in $1, \ldots, \dim(\mathfrak{g})$ **do**
        $(\boldsymbol{e}^{(\text{out})})_g \leftarrow e^{-i\theta_{lk}\epsilon_k^{(g)}} (\boldsymbol{e}^{(\text{out})})_g$
      **end for**
      $\boldsymbol{e}^{(\text{out})} \leftarrow \bar{\boldsymbol{Q}}_k \boldsymbol{e}^{(\text{out})}$
    **end for**
  **end for**
**Output:** Vector $\boldsymbol{e}^{(\text{out})}$ of basis observables in $\mathfrak{g}$
**Asymptotic complexity:** $\mathcal{O}(M \dim(\mathfrak{g})^2)$

where $\bar{\boldsymbol{U}}_{a:b} \equiv e^{-i\theta_b \bar{\boldsymbol{H}}_b} e^{-i\theta_{b-1}\bar{\boldsymbol{H}}_{b-1}} \ldots e^{-i\theta_a \bar{\boldsymbol{H}}_a}$, $m$ is a combined index for $l$ and $k$, and $M = LK$. Naïvely evaluating Eq. (5.30) requires $\mathcal{O}(M^2)$ matrix-vector multiplications (each $\mathcal{O}(\dim(\mathfrak{g})^2)$), however by exploiting a recurrent property of the gradients, one may compute the gradient with only $\mathcal{O}(M)$ matrix-vector multiplications (i.e., with the same asymptotic complexity as simulating the circuit in the first place). This procedure, which follows the 'reverse-mode' gradient calculation of Ref. [372] in conjunction with the efficient spectral approach of Algorithm 1, is outlined in Algorithm 2. For all gradient-based optimizations in this chapter, I use the L-BFGS-B algorithm: although this does approximate the Hessian, it requires only the first-order gradient as input.

Higher order derivatives can also be computed, such as the Hessian

$$(\boldsymbol{\nabla}^2_{\boldsymbol{\theta}} \langle O \rangle)_{jk} = \frac{\partial^2 \langle O \rangle}{\partial \theta_j \partial \theta_k} = -\boldsymbol{h}^T \bar{\boldsymbol{U}}_{k+1:N} \bar{\boldsymbol{H}}_k \bar{\boldsymbol{U}}_{j:k} \bar{\boldsymbol{H}}_j \bar{\boldsymbol{U}}_{1:j} \boldsymbol{e}^{(\text{in})}. \tag{5.31}$$

However, since the circuit generators $\bar{\boldsymbol{H}}_k$ are typically non-invertible, Algorithm 2 cannot be generalized to evaluate Equation 5.31 and higher-order derivatives. Although still polynomial in complexity, evaluating a $P$-th order gradient for $P \geq 2$ will be a factor $\mathcal{O}(M^P)$ slower than efficiently evaluating the gradient.

**Algorithm 2** Calculating gradients in $\mathfrak{g}$-sim.

**Input:** Circuit $\boldsymbol{\theta}$, input expectation value vector $\boldsymbol{e}^{(\text{in})}$, circuit generator adjoint representations $\bar{\boldsymbol{H}}_k$ and eigendecompositions $\bar{\boldsymbol{Q}}_k$, $\epsilon_k^{(g)}$, coefficients $\boldsymbol{w}$ of observable $O$

$\boldsymbol{\lambda} \leftarrow \bar{\boldsymbol{U}}\boldsymbol{e}^{(\text{in})}$ ▷ Use the main loop of Algorithm 1.

$\boldsymbol{\phi} \leftarrow \boldsymbol{w}$

$(\boldsymbol{\nabla}_{\boldsymbol{\theta}}\langle O\rangle)_M \leftarrow -i\boldsymbol{\phi}^T\bar{\boldsymbol{H}}_M\boldsymbol{\lambda}$

**for** $m$ in $N, \ldots, 2$ **do**

  $\boldsymbol{\lambda} \leftarrow \bar{\boldsymbol{Q}}_m\boldsymbol{\lambda}$

  $\boldsymbol{\phi} \leftarrow \bar{\boldsymbol{Q}}_m\boldsymbol{\phi}$

  **for** $g$ in $1, \ldots, \dim(\mathfrak{g})$ **do**

    $\lambda_g \leftarrow e^{i\theta_m\epsilon_m^{(g)}}\lambda_g$

    $\phi_g \leftarrow e^{i\theta_m\epsilon_m^{(g)}}\phi_g$

  **end for**

  $\boldsymbol{\lambda} \leftarrow \bar{\boldsymbol{Q}}_m^T\boldsymbol{\lambda}$

  $\boldsymbol{\phi} \leftarrow \bar{\boldsymbol{Q}}_m^T\boldsymbol{\phi}$

  $(\boldsymbol{\nabla}_{\boldsymbol{\theta}}\langle O\rangle)_{m-1} \leftarrow -i\boldsymbol{\phi}^T\bar{\boldsymbol{H}}_m\boldsymbol{\lambda}$

**end for**

**Output:** Gradient vector $\boldsymbol{\nabla}_{\boldsymbol{\theta}}\langle O\rangle$

**Asymptotic complexity:** $\mathcal{O}(M\dim(\mathfrak{g})^2)$

### 5.2.2 Sparsity of adjoint representation

When the relevant basis of $\mathfrak{g}$ consists of Pauli strings and the gate generators are also Pauli strings, the adjoint representations $\bar{\boldsymbol{H}}_l$ of the gate generators and their corresponding parametrized gates are of an extremely sparse form, having fewer than $2\dim(\mathfrak{g})$ nonzero entries per $(\dim(\mathfrak{g}) \times \dim(\mathfrak{g}))$-dimensional matrix. Thus, substantial speedups can be obtained by leveraging sparse linear algebra libraries. This is the case for most, but not all, simulations presented in this chapter.

However, I sometimes must apply gates whose generators are not Pauli strings, such as $e^{-i\theta\sum_{j=1}^{n-1}\sigma_j^x\sigma_{j+1}^x}$ for VQE on the LTFIM (Sec. 5.4.2.2), or $e^{-i\theta\sum_{j=1}^{n-1}\sigma_j^z\sigma_{j+1}^z}$ for QAOA (Sec. 5.4.2.3). For gates like these, the adjoint representation of the generator does *not* admit an efficient sparse representation, and will in general be dense. Figure 5.3 depicts this difference in sparsity.

For gates not generated by single Pauli strings, but rather sums of commuting

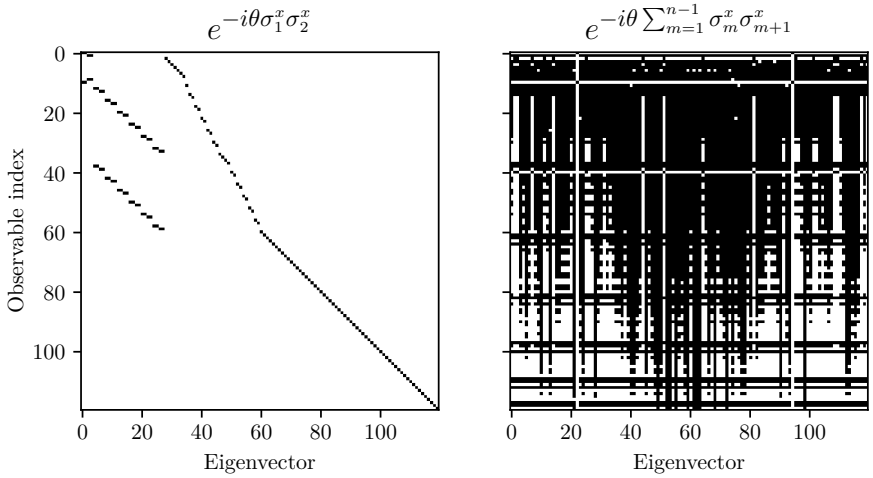



**Figure 5.3: Sparsity of gates generated by Pauli strings versus general gates.** I plot sparsity maps for the eigenvector matrix $\bar{\boldsymbol{Q}}$ where $\Phi_{\mathcal{G}}^{\mathrm{Ad}}(\exp(-i\theta H)) = \bar{\boldsymbol{Q}}\mathrm{diag}(e^{-i\theta\epsilon^{(1)}},\ldots,e^{-i\theta\epsilon^{(\dim(\mathfrak{g}))}})\bar{\boldsymbol{Q}}^T$, for $H=\sigma_1^x\sigma_2^x$ (left) and $H=\sum_{m=1}^{n-1}\sigma_m^x\sigma_{m+1}^x$ (right), for a system of $n=8$ qubits. Black entries in the map indicate non-zero matrix elements, and white entries indicate zero elements (up to machine precision). The gate $\exp(-i\theta\sigma_1^x\sigma_2^x)$ has a sparse eigenvector matrix since it is generated by a single Pauli string, whereas the gate $\exp(-i\theta\sum_{m=1}^{n}\sigma_m^x\sigma_{m+1}^x)$ has a dense eigenvector matrix since it is generated by a sum of Pauli strings.

Pauli strings, one can factorize the gate and still use sparse methods. For example,

$$e^{-i\theta\sum_{j=1}^{n-1}\sigma_j^x\sigma_{j+1}^x} = \prod_{j=1}^{n-1} e^{-i\theta\sigma_j^x\sigma_{j+1}^x} \tag{5.32}$$

$$\implies \underbrace{\Phi_{\mathcal{G}}^{\mathrm{Ad}}\left(e^{-i\theta\sum_{j=1}^{n-1}\sigma_j^x\sigma_{j+1}^x}\right)}_{\text{dense}} = \prod_{j=1}^{n-1} e^{-i\theta\overbrace{\Phi_{\mathfrak{g}}^{\mathrm{ad}}(\sigma_j^x\sigma_{j+1}^x)}^{\text{sparse}}}, \tag{5.33}$$

since the adjoint representation is a Lie algebra homomorphism. However, in such cases the dense nature of the eigenvector matrix can make GPU-accelerated dense linear algebra methods highly appealing. For example, I compared these different approaches in my implementation of $\mathfrak{g}$-sim on the typical hardware available to me at the Oxford ARC facility (a single core of an Intel Xeon Platinum 8268 for CPU methods vs a NVIDIA Tesla V100 for GPU methods). I found that for systems with sparse gates only, GPU acceleration provided an overall speedup of a factor of 3–5 versus a single CPU core, whereas for systems with dense gates, GPU acceleration provided an overall speedup of a factor of 60–100. The choice of sparse or dense methods and CPU or GPU hardware must be made judiciously depending

on problem type and the need for parallelizability; I use a careful mixture of both in the applications section of this chapter (Section 5.4).

### 5.2.3 Efficient calculation and eigendecomposition of Pauli-basis adjoint representations

It is often the case that algebraic methods for quantum dynamics are bottlenecked by the evaluation of structure constants. Even when the structure constants can be evaluated in polynomial time, it is common for the polynomial degree to be sufficiently high that computation remains impractical for even modest-sized systems [373]. Furthermore, expensive manipulations of the adjoint representations, such as the eigendecompositions needed to circumvent matrix exponentiation (Algorithm 1), can add a drastic up-front cost to simulations even if they only need to be performed once per algebra. Fortunately, I devised highly efficient methods to compute certain adjoint representations and their eigendecompositions.

Executing $\mathfrak{g}$-sim requires one to compute the adjoint representations (Definition 4) of the gate generators $H_k$, given by

$$\left(\bar{\boldsymbol{H}}_k\right)_{\alpha\beta} = -\operatorname{Tr}\left[H_k\left[G_\alpha, G_\beta\right]\right]. \tag{5.34}$$

Naïvely computing an adjoint representation via Equation 5.34 would scale as $\mathcal{O}(\dim(\mathfrak{g})^2)$. However, if the basis operators $G_\alpha$ of $\mathfrak{g}$ are Pauli strings

$$G_\alpha = \bigotimes_{m=1}^{n} \sigma_m^{s_{\alpha,m}}, \quad s_{\alpha,m} \in \{x, y, z\}, \tag{5.35}$$

as is the case for the algebra used predominantly throughout this chapter (5.38), and $H_k$ is also a Pauli string, then for fixed $k$ and $\alpha$ ($\beta$), Equation 5.34 has a nonzero value for at most one value of $\beta$ ($\alpha$). That is, each row and column have at most one nonzero entry. Thus, one can construct the adjoint representation for $\bar{\boldsymbol{H}}_k$ in just $\mathcal{O}(\dim(\mathfrak{g}))$ by iterating over $\alpha \in \{1, \ldots, \dim(\mathfrak{g})\}$, inverting Equation 5.34 to find the Pauli string $\sigma$ such that $\operatorname{Tr}[H_k[G_\alpha, \sigma]]$ (which is possible since $H_k$ and $G_\alpha$ are Pauli strings), checking if $\sigma = G_\beta$ for some $\beta$ (which can be done in $\mathcal{O}(n)$ by using a hashmap), and populating the corresponding element if so.

Similarly, although naïvely computing the eigendecomposition (5.27) would be $\mathcal{O}(\dim(\mathfrak{g})^3)$, one may compute it in just $\mathcal{O}(\dim(\mathfrak{g}))$ in the case where the $H_k$ and $G_\alpha$ are Pauli strings. Since the $\bar{\boldsymbol{H}}_k$ are Hermitian and at most one element of each row/column is nonzero, the adjoint representations take the form (up to some permutation of rows and columns)

$$\bar{\boldsymbol{H}}_k = 2\left(\bigoplus_{m_1=1}^{(\dim(\mathfrak{g})-\dim(\ker(\bar{\boldsymbol{H}}_k)))/2} \bar{\boldsymbol{\sigma}}^y\right) \oplus \left(\bigoplus_{m_2=1}^{\dim(\ker(\bar{\boldsymbol{H}}_k))/2} \bar{\mathbf{0}}\right), \tag{5.36}$$

where the factor of 2 originates from the Pauli commutation relations, and $\bar{\boldsymbol{\sigma}}^y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$ and $\bar{\mathbf{0}} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$ are the Pauli Y matrix and the zero matrix, respectively. Since the eigendecomposition of $\bar{\boldsymbol{\sigma}}^y$ is known, one may compute the eigendecomposition of $\bar{\boldsymbol{H}}_k$ in $\mathcal{O}(\dim(\mathfrak{g}))$ by iterating over the $\bar{\boldsymbol{\sigma}}_y$ summands, populating the corresponding eigenvectors and eigenvalues, and using any set of vectors spanning $\ker(\bar{\boldsymbol{H}}_k)$ as the remaining eigenvectors with zero eigenvalues.

## 5.3 Scalability and demonstration

In this section, I demonstrate the capabilities of $\mathfrak{g}$-sim and contrast it to related simulation techniques. First, I collect and summarize the conditions for its scalability in Section 5.3.1. To facilitate this scalability, in Section 5.3.2 I introduce the algebra of interest $\mathfrak{g}_0$ that forms the basis of many of the later applications. Next, in Section 5.3.3 I report results for noiseless and noisy simulations for a system of $n = 200$ qubits and discuss its advantages relative to related free-fermion methods. This is followed by a numerical benchmarking procedure in Section 5.3.4, to more systematically verify polynomial resource scalings.

### 5.3.1 Scalability of $\mathfrak{g}$-sim

As outlined in Section 5.1, $\mathfrak{g}$-sim recasts quantum evolution as linear algebra problems on vectors in $\mathbb{R}^{\dim(\mathfrak{g})}$ and matrices in $\mathbb{R}^{\dim(\mathfrak{g})\times\dim(\mathfrak{g})}$. While these techniques can be applied to *any* quantum system, most dynamical Lie algebras have dimension $\dim(\mathfrak{g}) \in \Theta(4^n)$ (since randomly sampled anti-Hermitian operators are known to generate the whole special unitary algebra $\mathfrak{su}(d)$ [374]), and thus this framework

does not in general yield an asymptotic advantage. Nonetheless, there are special cases where $\dim(\mathfrak{g}) \in \mathcal{O}(\text{poly}(n))$, such that $\mathfrak{g}$-sim can be used to perform classically efficient simulations despite the exponential dimension of $\mathcal{H}$.

For convenience, I explicitly reiterate the conditions required for classically efficient simulations via $\mathfrak{g}$-sim.

1. The Lie closure of the gate generators must lead to an algebra $\mathfrak{g}$ with $\dim(\mathfrak{g}) \in \mathcal{O}(\text{poly}(n))$.

2. One must know a Schmidt-orthonormal basis of $\mathfrak{g}$, as well as the structure constants.

3. Observables of interest must be supported by $\mathfrak{g}$ or products of terms in $\mathfrak{g}$ up to some fixed order $M$.

4. The expectation values of the basis elements, or their products up to some fixed order $M$, over the initial state must be known.

Some brief comments regarding these requirements are warranted. First, recall that while most evolutions lead to exponentially sized algebras, there are special cases where the algebra grows polynomially with the number of qubits. These include systems with permutation symmetries [229, 373, 375] and systems with free-fermion mappings [213, 376], even when the explicit construction of the free-fermion mapping is very complicated or not known at all. One must also construct, or obtain, the structure constants: for $\mathfrak{g}_0$, the methods of Section 5.2.3 enable one to compute all of the structure constants with time complexity scaling as $\mathcal{O}(n^4)$, and for permutation-symmetric systems the authors of Ref. [373] present an algorithm to compute the structure constants with time complexity scaling as $\mathcal{O}(n^7)$. In the sections that follow, I consider a free-fermion-mappable model (Equation 5.37) where $\dim(\mathfrak{g}) \in \mathcal{O}(n^2)$, and where all the structure constants can be obtained with a time complexity of $\mathcal{O}(n^5)$. Finally, I note that the expectation values of the basis elements (or their products) can be efficiently computed on a classical computer for certain families of initial states, e.g., product states, stabilizer states,

or the so-called highest weight states [112]. For more general input states, one can always estimate these expectation values by using a quantum computer, preparing $\rho^{(\text{in})}$, and making measurements. For this reason, $\mathfrak{g}$-sim is in the complexity class $\text{CSIM}_{\text{QE}}$ as defined by Ref. [134], whereas the framework of Somma [112, 113] is in the complexity class CSIM. Either way, the procedure yields a form of "classical description" [373] of the input state that one can act upon with $\mathfrak{g}$-sim.

### 5.3.2 A polynomial-sized Lie algebra

As a specific example of an algebra with $\dim(\mathfrak{g}) \in \text{poly}(n)$, throughout the remainder of this chapter's demonstrations I consider a special model with free-fermion mappings, and thus $\dim(\mathfrak{g}) \in \mathcal{O}(n^2)$. The algebra under consideration is given by

$$\mathfrak{g}_0 = \text{span}_{\mathbb{R}} \left\langle \left( \bigcup_{\mu,\nu=x,y} \{ i\sigma_j^\mu \sigma_{j+1}^\nu \}_{j=1}^{n-1} \right) \cup \{ i\sigma_j^z \}_{j=1}^n \right\rangle_{\text{Lie}} . \tag{5.37}$$

As shown in Ref. [377], a basis for $\mathfrak{g}_0$ is given by the set of Pauli strings

$$i\{ \sigma_j^z, \widehat{\sigma_i^x \sigma_j^x}, \widehat{\sigma_i^y \sigma_j^y}, \widehat{\sigma_i^x \sigma_j^y}, \widehat{\sigma_i^y \sigma_j^x} \,|\, 1 \leq i < j \leq n \} , \tag{5.38}$$

where I use the notation $\widehat{A_i B_j} = A_i \sigma_{i+1}^z \cdots \sigma_{j-1}^z B_j$. This algebra is a representation of $\mathfrak{so}(2n)$, and hence its dimension is $\dim(\mathfrak{g}_0) = n(2n-1)$ [377, 378]. All the representation elements can be obtained with a time complexity of $\mathcal{O}(n^5)$; in the GitHub repository of Ref. [379] I report these basis elements as well as their pre-computed structure constants.

Finally, I note that the expectation values of the basis elements (or their products) can be efficiently computed on a classical computer for certain families of initial states, e.g., product states, stabilizer states, or the so-called highest weight states [112]. Interestingly, as I explore in the following section, this can potentially include initial states that are magic states for the class of circuits under consideration [380, 381], allowing $\mathfrak{g}$-sim to evolve some initial states that typical free-fermionic methods cannot. For more general input states, one can always estimate these expectation values by using a quantum computer, preparing $\rho^{(\text{in})}$, and making measurements on all the algebra basis elements (i.e. it is in

the complexity class $\mathrm{CSIM}_{\mathrm{QE}}$ [134]). In any case, this procedure yields a form of classical description of the input state that $\mathfrak{g}$-sim can use.

Furthermore, the subalgebras of $\mathfrak{g}_0$ encompass many other models of interest:

- The Hamiltonian algebra for the XY model with open boundary conditions and free coefficients forms a proper subalgebra $\mathfrak{g}_{\mathrm{XY}} \subset \mathfrak{g}_0$ with $\dim(\mathfrak{g}_{\mathrm{XY}}) = n(n-1)$ [382], given by

$$\mathfrak{g}_{\mathrm{XY}} = \mathrm{span}_{\mathbb{R}} \left\langle \left( \{ i\sigma_j^x \sigma_{j+1}^x \}_{j=1}^{n-1} \right) \cup \left( \{ i\sigma_j^y \sigma_{j+1}^y \}_{j=1}^{n-1} \right) \right\rangle_{\mathrm{Lie}} . \tag{5.39}$$

- The Hamiltonian algebra for the transverse field Ising model (TFIM) with open boundary conditions and non-free coupling coefficients forms a proper subalgebra $\mathfrak{g}_{\mathrm{TFIM}} \subset \mathfrak{g}_0$ with $\dim(\mathfrak{g}_{\mathrm{TFIM}}) = n^2$ [378], given by

$$\mathfrak{g}_{\mathrm{TFIM}} = \mathrm{span}_{\mathbb{R}} \left\langle \left( i \sum_{j=1}^{n-1} \sigma_j^x \sigma_{j+1}^x \right) \cup \left( i \sum_{j=1}^{n} \sigma_j^z \right) \right\rangle_{\mathrm{Lie}} . \tag{5.40}$$

- For a path graph $P_n$ on n vertices, the circuit of the quantum approximate optimization algorithm [23] has an associated algebra

$$\mathfrak{g}_{\mathrm{QAOA},P_n} = \mathrm{span}_{\mathbb{R}} \left\langle \left( i \sum_{j=1}^{n-1} \sigma_j^z \sigma_{j+1}^z \right) \cup \left( i \sum_{j=1}^{n} \sigma_j^x \right) \right\rangle_{\mathrm{Lie}} , \tag{5.41}$$

  which is isomorphic to the one of the TFIM, $\mathfrak{g}_{\mathrm{QAOA},P_n} \cong \mathfrak{g}_{\mathrm{TFIM}}$. The two are related by a Hadamard transform. The algebras for the cycle graph $C_n$ and complete graph $K_n$ are also of polynomial size [378].

### 5.3.3 Example simulation task

To demonstrate the power of $\mathfrak{g}$-sim, I first showcase its application to a quantum simulation task which, to the best of my knowledge, *would not be possible with any other known classical simulation method.* I seek to simulate the dynamics of the transverse field XY (TFXY) model with random magnetic fields, whose Hamiltonian reads

$$H_{\mathrm{TFXY}} = \sum_{j=1}^{n-1} (\sigma_j^x \sigma_{j+1}^x + \sigma_j^y \sigma_{j+1}^y) + \sum_{j=1}^{n} b_j \sigma_j^z , \tag{5.42}$$

where the coefficients $b_j$ are randomly drawn from a Gaussian distribution $N(0, \xi^2)$. Hamiltonian dynamics in $\mathfrak{g}$-sim can be implemented either by directly exponentiating the adjoint representation of the Hamiltonian (5.8) or by simulating an appropriate quantum circuit (5.12). Here, I apply the latter so that I can also demonstrate the noisy simulation techniques of Section 5.1.3.4 by inserting noise channels between gates. In each case, I apply a first-order Trotterization of the dynamics with a total of 300 Trotter steps.

In the first demonstration, I initialize the system in the magic state

$$|\psi\rangle = \left[\frac{|0000\rangle + |0011\rangle + |1100\rangle + e^{i\tau}|1111\rangle}{2}\right]^{\otimes n/4}, \tag{5.43}$$

for $n = 200$ qubits, computing the dynamics of the observables in $i\mathfrak{g}_0$. Although this is a magic state for free-fermion circuits [380], the $\mathfrak{g}$-sim classical representation can be trivially computed, as it is a product state between 4-qubit subspaces. I emphasize that this demonstration highlights the differences between $\mathfrak{g}$-sim and methods based upon FLO or Wick's theorem. This initial state has fermionic linear optical extent growing exponentially in $n$ [381], and would therefore be intractable with these methods. To the best of my knowledge, the methods of this chapter are the only known means capable of classically simulating these dynamics at this system size.

In Figure 5.4(a) I depict the dynamics of this system in the absence of noise, showing the propagation of correlations across the system by plotting the $\langle \sigma_j^y \sigma_{j+1}^x \rangle$ correlators. Despite being intractable with competing methods, this simulation was completed in 16 minutes on a single CPU core. In Figure 5.4(b), I repeat this simulation under the presence of noise using the methods outlined in Section 5.1.3.4. Following every entangling gate, I apply a random 2-qubit Pauli noise channel

$$\rho \to (1-p)\rho + \sum_{\substack{\nu_1,\nu_2 \in \{1,x,y,z\} \\ (\nu_1,\nu_2) \neq (1,1)}} w_{\nu_1,\nu_2} \sigma_j^{\nu_1} \sigma_k^{\nu_2} \rho \; \sigma_k^{\nu_2} \sigma_j^{\nu_1} \tag{5.44}$$

$$w_{\nu_1,\nu_2} = p^{-1} c_{\nu_1,\nu_2} \sum_{\substack{\nu_1',\nu_2' \in \{1,x,y,z\} \\ (\nu_1',\nu_2') \neq (1,1)}} c_{\nu_1',\nu_2'}, \tag{5.45}$$

where the $c_{\nu_1,\nu_2} \sim \mathcal{U}(0,1)$ are uniformly distributed random variables. For the simulation depicted here, I choose an overall per-gate fault probability of $p = 3 \times 10^{-4}$.

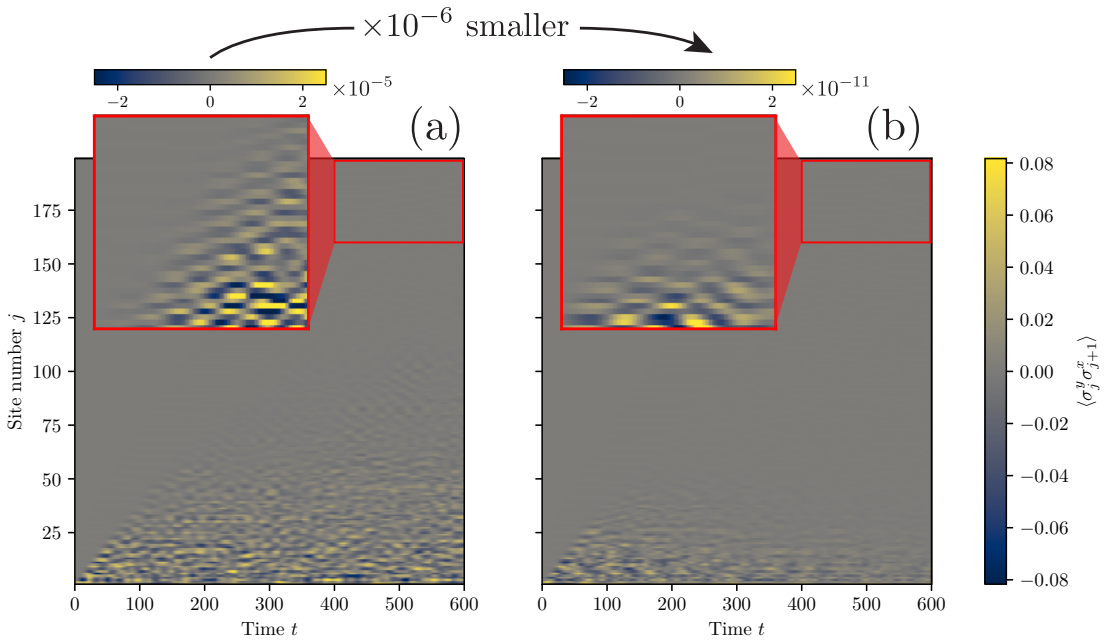



**Figure 5.4: Noiseless and noisy dynamics of a 200-qubit magic state using $\mathfrak{g}$-sim.** I simulate the Trotterized dynamics of an initial magic state (5.43) under a TFXY spin-chain Hamiltonian (5.42), for $n = 200$ qubits and $\tau = 2.81$ using $\mathfrak{g}$-sim. To illustrate the dynamics, I depict here correlators of the form $\langle \sigma_j^y \sigma_{j+1}^x \rangle$. **(a)** Simulating the dynamics in the absence of noise, one can see that the correlations propagate across the chain. This simulation ran in 16 minutes on a single CPU core. **Inset:** Small oscillations in the correlators reach the far end of the chain, confirming that the light cone of the dynamics reaches from the first to last qubit. **(b)** Including random noise in the dynamics (random 2-qubit Pauli channels after each entangling gate with a fault probability of $3 \times 10^{-4}$), one can see that the dissipation weakens the correlations by 6 orders of magnitude, and qualitatively changes the form of the oscillations.

Under the presence of noise, the correlations dissipate, although are not entirely eliminated: oscillations can still be seen at the far end of the chain, albeit 6 orders of magnitude smaller. This highlights the exact nature of $\mathfrak{g}$-sim — many other methods for efficient classical simulation of noisy dynamics would be unlikely to capture this, as a common strategy involves truncating low-weight contributions to the dynamics that are mostly consumed by noise [279]. An *exact* classical simulation method supporting arbitrary Pauli noise channels is comparatively unusual, making $\mathfrak{g}$-sim a valuable addition to the growing arsenal of classical simulation techniques.

When performing simulations with $\mathfrak{g}_0$, one might be inclined to think of $\mathfrak{g}$-sim as effectively being the same as FLO — after all, FLO works due to the poly$(n)$ size of the underlying Lie algebra. Nonetheless, the ability to perform these simulations

highlights that despite the similarities, $\mathfrak{g}$-sim with $\mathfrak{g}_0$ *is a distinct method from FLO*. These methods have different scope, each being able to compute properties that the other cannot, and are fundamentally different methods with different limitations. $\mathfrak{g}$-sim therefore represents a genuinely new addition to the toolbox of classical simulation methods, even when applied to systems with free-fermion mappings.

### 5.3.4 Resource benchmark

With a non-trivial application of $\mathfrak{g}$-sim demonstrated, I now consider its performance and scalability more systematically with a benchmarking task, aiming to verify its polynomial resource scaling numerically. For all benchmark computations, I take the circuits $U(\boldsymbol{\theta})$ to be composed of a single layer of gates generated by the set of operators $\left(\bigcup_{\mu,\nu=x,y}\{\sigma_j^\mu\sigma_{j+1}^\nu\}_{j=1}^{n-1}\right)\cup\{\sigma_j^z\}_{j=1}^n$, where $\sigma_j^\mu$ (with $\mu=x,y,z$) denotes the Pauli operators acting on the $j$-th qubit. In Figure 5.5(a), I depict the circuit for $n=3$.

For the benchmarking task, I evaluate expectation values of random observables in the algebra $\mathfrak{g}_0$ for the state obtained by applying $U(\boldsymbol{\theta})$, with parameters uniformly sampled in the interval $[0,2\pi]$, to an input state $\rho^{(\mathrm{in})}=|0\rangle\langle 0|^{\otimes n}$. Since all the elements of the basis of $\mathfrak{g}_0$ (given in Equation 5.37) are Pauli operators, one can trivially compute the expectation values in $\boldsymbol{e}^{(\mathrm{in})}$:

$$\mathrm{Tr}\left[\rho^{\mathrm{in}}G_\alpha\right]=\begin{cases}1 & \text{if} \quad G_\alpha=\sigma_j^z\ (j=1,\ldots,n)\\ 0 & \text{otherwise}\end{cases} \quad . \tag{5.46}$$

In Figure 5.5(b), I show benchmark results in terms of memory requirements. The memory requirements for state vectors are computed assuming double-precision arithmetic on a $2^n$-dimensional Hilbert space $\mathcal{H}$. Meanwhile, the memory requirements for $\mathfrak{g}$-sim are computed by profiling the actual memory footprint of the observable vector $\boldsymbol{e}$ and algebra data (adjoint representations of generators $\bar{\boldsymbol{H}}_k$ and their eigendecompositions). A further comparison of these memory requirements to other methods, such as tensor-network methods, was depicted previously in Figure 2.1. As expected, it can clearly be seen that memory requirements scale as $\Omega(2^n)$ for state vector methods, and as $\mathcal{O}(\mathrm{poly}(n))$ for $\mathfrak{g}$-sim.

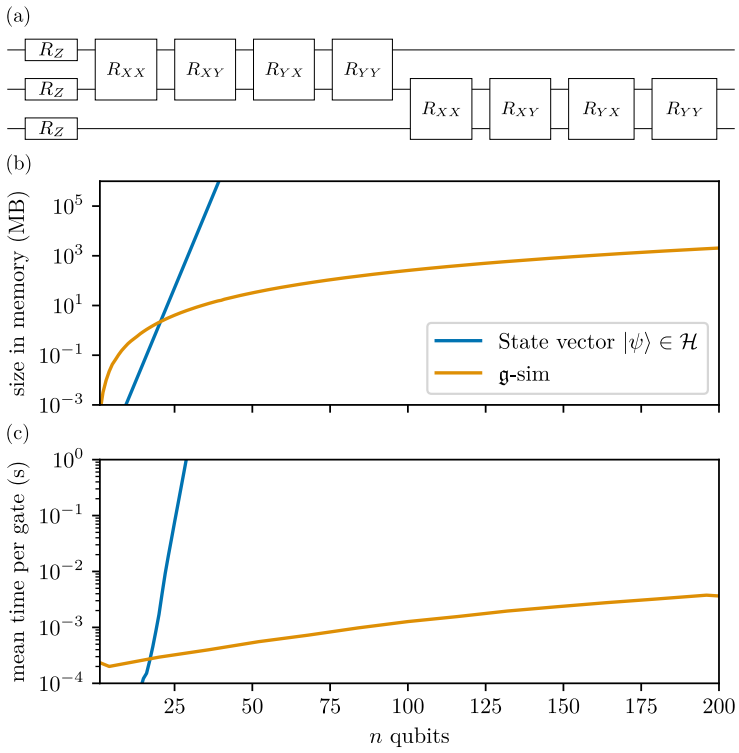



**Figure 5.5: Performance benchmarks of 𝔤-sim. (a)** Circuit used for the benchmark at $n = 3$ qubits. The set of generators are presented in Equation 5.37. Here, $R_Z$ denotes a rotation about the $Z$ axis, and $R_{\mu\nu}$ a rotation generated by the Pauli operator $\sigma^\mu\sigma^\nu$. I compare the memory **(b)** and compute time **(c)** requirements for state vector simulations (blue) and 𝔤-sim (orange) for the unitary shown in panel (a).

As for the computational benchmarks, these are computed on a single CPU core of an Intel Xeon Platinum 8268 at the Oxford ARC facility — using TensorFlow Quantum [105] for state vector methods, and my efficient Python implementation for 𝔤-sim. I stress that unlike tensor network methods (see Section 3.2.2 for an elaboration), the time complexity of applying a gate is independent of circuit depth, so I report timings per gate. I collect these timings using the `timeit` Python package in batches of repeated simulations (to minimize the influence of measurement distortion due to environmental factors) and average over batches. I bootstrap 95% confidence intervals on these averages, but they are too small to be visualized. In Figure 5.5(c), I show the results of these compute time benchmarks, where it can be seen that the compute time again scales exponentially in $n$ for state vector methods, but only polynomially for 𝔤-sim. The exponential scaling of state-vector simulations makes simulations with $n \geq 34$ qubits infeasible on a

single node at the Oxford ARC facility. However, using $\mathfrak{g}$-sim I am able to simulate systems of up to 200 qubits, and likely beyond, on a single CPU core.

These demonstrations highlight the capabilities and scalability of the $\mathfrak{g}$-sim framework, and in particular its differences and advantages relative to similar methods. In the section that follows, I proceed explore its application to archetypal practical problems in variational quantum computing.

## 5.4 Applications

Having presented the framework for $\mathfrak{g}$-sim, I now explore its application to practical problems. These include a study of the landscape of VQAs highlighting the overparametrization phenomenon (Section 5.4.1), designing improved initialization of quantum circuits parameters (Section 5.4.2), unitary synthesis and quantum compilation (Section 5.4.3), and demonstration of the training of a quantum-phase classifier (Section 5.4.4). Collectively, these illustrate the breadth of applications that can be addressed with $\mathfrak{g}$-sim within the context of variational quantum computing. Furthermore, the work of Section 5.4.3 further distinguishes the theoretical framework of $\mathfrak{g}$-sim from that of Refs. [112, 113], as these tools had not previously been extended to quantum compilation.

I note that in all cases, I implement $\mathfrak{g}$-sim based on either $\mathfrak{g}_0$ or subalgebras $\mathfrak{g} \subseteq \mathfrak{g}_0$ obtained by removing some of the generators from Equation 5.37. In each section, I specify which subalgebra of $\mathfrak{g}_0$ I will be working with, but the simulation itself is always performed using the full $\mathfrak{g}_0$ algebra (as its given basis is a Pauli basis, leading to the benefits outlined in Section 5.2.1). I further remark that by definition, I will always be simulating circuits composed of single- and two-qubit gates with local connectivity — that is, circuits that could be implemented on most quantum hardware without incurring large compilation overhead. Since many variational ansätze are utilized throughout this section, I summarize them all in Table 5.1.

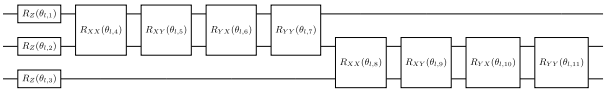
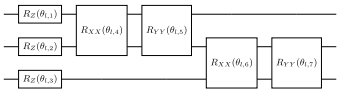
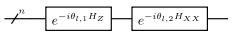
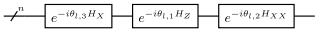
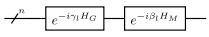
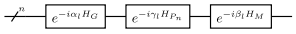
| | Applications | $\mathfrak{g}$-sim efficient | Ansatz layer |
|---|---|---|---|
| TFXY 2-local | Benchmarking (5.2.1)<br>TFXY VQE (5.4.1)<br>Random compilation (5.4.3.3)<br>Phase classifier (5.4.4) | Yes |  |
| TFXY Trotter | Dynamics compilation (5.4.3.5) | Yes |  |
| TFXY HVA | LTFIM pre-training (5.4.2.2) | Yes |  |
| LTFIM HVA | LTFIM pre-training (5.4.2.2) | No |  |
| QAOA standard | QAOA pre-training (5.4.2.3) | For $H_G \in \{P_n, C_n\}$ |  |
| QAOA pre-trained | QAOA pre-training (5.4.2.3) | For $H_G \in \{P_n, C_n\}$ |  |

**Table 5.1:** Layers of all variational ansätze used in this chapter, with reference to their applications, whether or not they are classically efficient in $\mathfrak{g}$-sim (for simulating dynamics of observables supported by $\mathfrak{g}$), and circuit diagrams for single layers of each ansatz. In diagrams where individual qubits are depicted, the example layer is given for $n = 3$ for diagrammatic compactness.

### 5.4.1 Characterizing VQAs

In Section 2.2, I discussed how VQAs aim to enable near-term quantum advantage by means of a hybrid quantum-classical training loop, where some of the problem difficulty is offloaded to an optimization problem on a classical co-processor. Even in the absence of barren plateaus, however, the optimization problem can be difficult due to the presence of local minima, saddle points, and other optimization challenges, and is an NP-hard classical problem in general [241]. Much remains to be learned about the scalability of VQAs, but analytic study of optimization landscapes is difficult and limited in scope, while computational study is hindered by the exponential resource costs of state vector simulation (see Section 2.1). Therefore, scalable classical simulations — including, but not limited to $\mathfrak{g}$-sim — allow one to characterize trainability of VQAs at system sizes that would otherwise be intractable. A previous study has used free-fermion simulations for this purpose [383], but using $\mathfrak{g}$-sim expands the set of systems that can be studied. As a warm-up problem, in

this section I demonstrate the onset of overparametrization in VQE problems for the transverse field XY model at up to $n = 50$ qubits.

#### 5.4.1.1 Overparametrization in VQE

In Section 2.2.4, I discussed the phenomenon of overparametrization in VQAs. In Ref. [225], it was shown that the model capacity of circuits of the form in Equation 5.1 can be quantified by $\dim(\mathfrak{g})$. In turn, overparametrization is characterized by a 'computational phase transition' happening at a critical number of parameters $N_p^{(\text{crit})} \leq \dim(\mathfrak{g})$, below which the circuit is hard to train and above which it becomes easy to train. Exact values of this critical threshold $N_p^{(\text{crit})}$ are state-dependent [225], and often hard to assess analytically as they rely on the conjugation relationship between the initial state and the Lie algebra $\mathfrak{g}$ of the circuit. Thus, exact details of this phenomenon are best probed numerically. However, initial numerical demonstrations of the phenomenon were only provided for systems of 2–10 qubits [225]. Here I use $\mathfrak{g}$-sim to demonstrate the phenomenon in problems of up to 50 qubits, which is well beyond the reach of exact state vector simulations.

#### 5.4.1.2 Simulation results

To probe overparametrization, I consider a VQE task where the goal is to prepare the ground state of the TFXY model (5.42). As an ansatz for $U(\boldsymbol{\theta})$ I consider a Hamiltonian variational ansatz [384, 385] of the form in Equation 5.1, with gate generators taken from the set $\{\sigma_j^x\sigma_{j+1}^x, \sigma_j^y\sigma_{j+1}^y\}_{j=1}^{n-1} \cup \{\sigma_j^z\}_{j=1}^n$. This ansatz is depicted in Table 5.1. One can see by inspection that the dynamical Lie algebra associated with this set of generators is precisely $\mathfrak{g}_0$ in Equation 5.37, such that the structure constants reported in my GitHub repository [379] can be used. Given that $\dim(\mathfrak{g}_0) = n(2n-1)$ and the gates in $U(\boldsymbol{\theta})$ can be parallelized, it follows that overparametrization is achievable with linear circuit depth. To solve the VQE problem, the parameters are optimized using the L-BFGS algorithm with the gradient evaluated according to Algorithm 2, to minimize the energy

$$E_{\text{TFXY}}(\boldsymbol{\theta}) = \langle \mathbf{0} | U^\dagger(\boldsymbol{\theta}) H_{\text{TFXY}} U(\boldsymbol{\theta}) | \mathbf{0} \rangle . \tag{5.47}$$

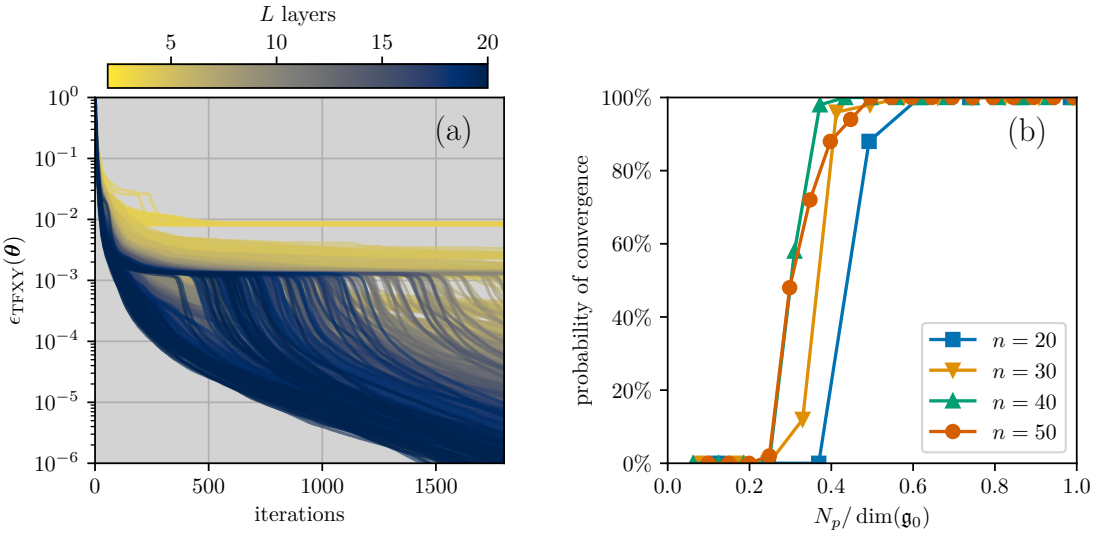



**Figure 5.6: Overparametrization in large system sizes. (a)** Convergence traces at $n = 50$ qubits of the approximate training error $\epsilon_{\mathrm{TFXY}}(\boldsymbol{\theta})$. **(b)** The probability of converging to $\epsilon_{\mathrm{TFXY}} < 10^{-4}$ for uniform random initialization of $\boldsymbol{\theta}$ as measured by 50 samples, at varied circuit depths, with corresponding number of parameters $N_p$ reported as a fraction of $\dim(\mathfrak{g}_0)$, and for $n$ ranging from 20 to 50 qubits.

Note that the measurement operator $H_{\mathrm{TFXY}}$ is, by definition, fully supported within $\mathfrak{g}_0$. Moreover, since the initial state is the all-zero state (for which Pauli expectation values are trivial to compute), one can use the results in Equation 5.46 to construct the vector $\boldsymbol{e}^{(\mathrm{in})}$. Per the summary in Section 5.3.1, this means that all the ingredients for $\mathfrak{g}$-sim are in place, and the problem can be efficiently simulated.

For each of the system sizes probed, from $n = 20$ to 50 qubits, I perform optimization over circuits with a varied number of layers $L$ chosen to result in a number $N_p$ of circuit parameters spanning the range $[1, \dim(\mathfrak{g}_0)]$, and random magnetic field amplitude $\xi = 0.1$. In each case, optimizations are repeated over 50 randomized initial parameter values and fields. Results of this numerical experiment are reported in Figure 5.6. To illustrate the difference in under- and overparametrized optimization, I display all the optimization traces at a fixed system size of $n = 50$ qubits in Figure 5.6(a). There, I observe the convergence behavior of the approximate training error $\epsilon_{\mathrm{TFXY}}(\boldsymbol{\theta}) = (E_{\mathrm{TFXY}}(\boldsymbol{\theta}) - E_{\min})/\|H_{\mathrm{TFXY}}\|_{\mathrm{HS}_1}$, where $E_{\min}$ is the lowest energy discovered by VQE across all depths and $\|\cdot\|_{\mathrm{HS}_1} = \|\cdot\|_{\mathrm{HS}}/2^n$ for $\|\cdot\|_{\mathrm{HS}}$ the Hilbert-Schmidt norm (i.e., I normalize the training error relative to the energy scale

of the Hamiltonian). The optimization traces exhibit a clear change in the hardness of the optimization problem as the depth of the circuits is varied. Below a critical threshold of $L \approx 6$ layers, one can see severe trainability issues where none of the optimization manages to converge to the ground state energy. However, for circuits with $L > 6$ layers a sudden transition to trainability is observed and solutions converge towards the minimum energy. This phase transition in computational complexity indicates the onset of overparametrization [225].

With a specific instance of overparametrization observed, I proceed to study the phenomenon more systematically across system sizes. In Figure 5.6(b), I study this phenomenon across varying system sizes $n$, from 20 to 50 qubits, and report the probability of convergence to $\epsilon_{\text{TFXY}} < 10^{-4}$ as a function of the number of circuit parameters in units of $\dim(\mathfrak{g}_0)$. I observe that the transition to the trainable region (large convergence probability) occurs consistently across all system sizes at $N_p^{(\text{crit})} \approx 0.3 \dim(\mathfrak{g})$ irrespective of $n$. This supports the analytic results of Ref. [225], and demonstrates the phenomenon of overparametrization at system sizes beyond what could be simulated with state vector simulations. Overall, analysis of VQA trainability at scales exceeding the capability of state-vector simulations, that requires repetitions over many optimizations and many circuit sizes, is made possible by $\mathfrak{g}$-sim.

In addition to serving as a warm-up problem to demonstrate $\mathfrak{g}$-sim, overparametrization is crucial throughout the remainder of this work. It is provably achieved for circuits with $N_p \geq \dim(\mathfrak{g})$ parameters, and thus the tractably overparmeterizable models are those with $\dim(\mathfrak{g}) \in \mathcal{O}(\text{poly}(n))$. In cases where the relevant loss function depends entirely on observables supported by the algebra of the circuit (e.g., VQE with Hamiltonian variational ansatz [384–387] or QAOA), circuits overparametrizable in polynomial depth *are exactly those that can be efficiently simulated with* $\mathfrak{g}$*-sim*. Hence, this phenomenon ensures that for any problem that is efficiently simulable in $\mathfrak{g}$-sim, one can guarantee good trainability using a circuit of only polynomial gate-count. Crucially, this ensures a correctly trained prior model for my pre-training scheme (Sec. 5.4.2), the ability to efficiently approximately

compile unitaries in $\mathcal{G}$ to linear-depth circuits (Section 5.4.3), and good trainability of QML models (Section 5.4.4).

### 5.4.2 Pre-training quantum circuits

Although overparametrization guarantees trainability from *arbitrary* initial parameter values, in general cases where $\dim(\mathfrak{g}) \in \Theta(4^n)$ (i.e., cases where $\mathfrak{g}$-sim is no longer scalable) it does not result in a scalable strategy: trying to access the overparametrized regime would require constructing and training exponentially deep circuits, which is intractable for large problem sizes. Nonetheless, we need not have good trainability from arbitrary initial parameters for a successful strategy — it may be the case that with an *adequate* choice of initial parameters, one could still successfully train an underparametrized circuit. Furthermore, efficient trainability requires the ability to efficiently compute gradients, and therefore in either regime one must consider the impact of barren plateaus. Similarly, however, barren plateaus indicate exponentially vanishing gradients *on average*, but do not imply that the entire optimization landscape is exponentially flat (in fact, probabilistic barren plateaus are always accompanied by 'narrow gorges' in which minima are surrounded by regions of large gradient [220]). Hence, improvement in the trainability of quantum circuits — in both the geometry of the optimization landscape and the mitigation of barren plateaus — can be achieved by means of 'pre-training' or 'warm-starting' an ansatz such that its initial parameter values are sufficiently close to the optimal solution.

In this section, I demonstrate the use of $\mathfrak{g}$-sim for the initialization of VQAs, contributing to the growing body of techniques for warm-starting VQAs [251–263]. The central idea (expanded in Section 5.4.2.1) is to perform pre-training on a related auxiliary problem that induces a scalable Lie algebra, and to transfer the resulting solution as initial parameters for the circuit addressing the full (non-classically-simulable) target problem. The full problem, now warm-started with problem-informed parameters, can then be trained further on a quantum device. One should expect that the closer the auxiliary problem is to the target one, the more efficient

such a transfer will be. As a first example (Section 5.4.2.2), I study ground state preparation of the longitudinal-transverse field Ising model (LTFIM), pre-trained by first solving the transverse field Ising model (TFIM) efficiently using $\mathfrak{g}$-sim. The LTFIM only differs from the TFIM by a few additional generators (the longitudinal fields) and I find substantial improvement both in terms of the fidelity of the ground states prepared and the magnitude of the initial gradients when utilizing pre-training. More surprisingly, I also show that in optimization problems with QAOA (Section 5.4.2.3), for which target and auxiliary tasks differ substantially, improved performance can still be achieved over a majority of (but not all) problem instances.

#### 5.4.2.1 Pre-training strategy with $\mathfrak{g}$-sim

I consider again here a VQE task where the goal is to prepare the ground state of an Hamiltonian $H$ that can be decomposed as $H = \sum_j c_j h_j$, where $c_j$ are real valued coefficients. The Hamiltonian Lie algebra is defined as $\mathfrak{g}_{\text{target}} = \langle \{ih_j\} \rangle_{\text{Lie}}$ — that is, it is the algebra generated by the individual terms in $H$. The goal is to construct a circuit $U(\boldsymbol{\theta})$ to prepare the ground state of $H$, whose gate generators are some elements of $\mathfrak{g}_{\text{target}}$.

In the following sections, I will assume that $\dim(\mathfrak{g}_{\text{target}}) \notin \mathcal{O}(\text{poly}(n))$ (else the full VQE problem could be efficiently simulated with $\mathfrak{g}$-sim), meaning that $U(\boldsymbol{\theta})$ should not be constructed from a generating set of $\mathfrak{g}_{\text{target}}$. Hence, the scheme is as follows:

1. Identify a subset of the operators in $i\mathfrak{g}_{\text{target}}$ (denoted as $\{h_k\}_k$) such that their Lie closure $\mathfrak{g}_{\text{aux}} = \langle \{ih_k\} \rangle_{\text{Lie}}$, is a subalgebra $\mathfrak{g}_{\text{aux}} \subset \mathfrak{g}_{\text{target}}$ with $\dim(\mathfrak{g}_{\text{aux}}) \in \mathcal{O}(\text{poly}(n))$. Construct a proxy Hamiltonian $H_{\text{aux}} = \sum_k c_k h_k$. As the examples below will reveal, the choice of $c_k$ is informed by the task at hand.

2. On a classical computer, use $\mathfrak{g}$-sim to solve VQE on $H_{\text{aux}}$ using an ansatz generated by terms of $\mathfrak{g}_{\text{aux}}$ and with a number of parameters allowing for overparametrization.

3. Extend the trained ansatz with *new* gates generated by (some) of the terms in $\mathfrak{g}_{\text{target}} \setminus \mathfrak{g}_{\text{aux}}$. These new gates are initialized to the identity.

4. On a quantum computer, solve the VQE for $H$ starting from the ansatz constructed in step 3.

Although presented in the context of VQE, this warm-starting strategy could be utilized as-is for QML problems, so long as the classical representations of the training states have been obtained.

#### 5.4.2.2 Pre-training VQE for the LTFIM

The LTFIM is a paradigmatic spin-chain model providing a prototypical example of a quantum phase transition. Its Hamiltonian is given by

$$H_{\text{LTFIM}} = h_{xx} \sum_{j=1}^{n-1} \sigma_j^x \sigma_{j+1}^x + h_z \sum_{j=1}^{n} \sigma_j^z + h_x \sum_{j=1}^{n} \sigma_j^x, \tag{5.48}$$

where $h_{xx}, h_z, h_x \in \mathbb{R}$. It is well known that the algebra $\mathfrak{g}_{\text{target}}$ obtained from the terms in $H_{\text{LTFIM}}$ has exponential dimension $\mathfrak{g}_{\text{LTFIM}} \in \Theta(4^n)$ [213]. This renders it intractable in $\mathfrak{g}$-sim, induces a barren plateau for deep Hamiltonian variational circuits [213], and precludes efficient overparametrization [225], thus making application of VQE to identify the ground state of Equation 5.48 highly non-trivial.

Instead, by dropping the longitudinal field terms in $H_{\text{LTFIM}}$ one can obtain the TFIM Hamiltonian, given by

$$H_{\text{TFIM}} = h_{xx} \sum_{j=1}^{n-1} \sigma_j^x \sigma_{j+1}^x + h_z \sum_{j=1}^{n} \sigma_j^z \,. \tag{5.49}$$

Notably, taking the Lie closure of the terms in $H_{\text{TFIM}}$ one obtains $\mathfrak{g}_{\text{aux}} \subseteq \mathfrak{g}_0$. Thus, this successfully identifies a subset of operators in $i\mathfrak{g}_{\text{target}}$ leading to a an algebra whose dimension is in $\mathcal{O}(\text{poly}(n))$, making it overparametrizable [225] and classically tractable in $\mathfrak{g}$-sim.

I now apply the pre-training strategy to ground state preparations of $H_{\text{LTFIM}}$. Following Section 5.4.2.1, I begin by constructing an ansatz with $L = \dim(\mathfrak{g}_0)$ and gates generated by terms appearing in $H_{\text{TFIM}}$:

$$U(\boldsymbol{\theta}) = \prod_{l=1}^{\dim(\mathfrak{g}_0)} e^{-i\theta_{l,1} \sum_{j=1}^{n-1} \sigma_j^x \sigma_{j+1}^x} e^{-i\theta_{l,2} \sum_{j=1}^{n} \sigma_j^z}. \tag{5.50}$$

I initialize the parameters of the ansatz with uniform random values, and train them to prepare the ground state of $H_{\mathrm{TFIM}}$ using $\mathfrak{g}$-sim. Note that since the dynamical Lie algebra $\mathfrak{g}_{\mathrm{aux}}$ associated with Equation 5.50 is a subalgebra of $\mathfrak{g}_0$, I use the structure constants I pre-computed for $\mathfrak{g}_0$. This can be seen by noting that $e^{-i\theta_{l,1}\sum_{j=1}^{n-1}\sigma_j^x\sigma_{j+1}^x} = \prod_{j=1}^{n-1} e^{-i\theta_{l,1}\sigma_j^x\sigma_{j+1}^x}$, and that each gate generator is an element $\sigma_j^x\sigma_{j+1}^x$ of $\mathfrak{g}_0$. As the generators in this case are sums of Pauli strings, simulating this efficiently is best done using the dense methods outlined in Section 5.2.2.

Once the parameters in Equation 5.50 are trained to prepare the ground state of $H_{\mathrm{TFIM}}$, I modify the circuit by inserting new gates generated by the remaining term of $H_{\mathrm{LTFIM}}$, yielding the ansatz

$$U(\boldsymbol{\theta},\boldsymbol{\phi}) = \prod_{l=1}^{\dim(\mathfrak{g}_0)} \Big[ e^{-i\theta_{l,1}\sum_{j=1}^{n-1}\sigma_j^x\sigma_{j+1}^x} e^{-i\theta_{l,2}\sum_{j=1}^{n}\sigma_j^z} e^{-i\phi_l\sum_{j=1}^{n}\sigma_j^x} \Big]. \tag{5.51}$$

The new gates are then initialized to the identity by setting $\boldsymbol{\phi} = (0, 0, \ldots)$, while the other gates retain their pre-trained values. Finally, I proceed by training the full ansatz of Equation 5.51 to minimize the expectation value of $H_{\mathrm{LTFIM}}$. Since $\dim(\mathfrak{g}_{\mathrm{LTFIM}}) \in \Theta(4^n)$, this last step must be performed using state vector simulations (I use TensorFlow Quantum [105]), thus limiting the system sizes that can be probed to those described in Section 2.1.

In Figure 5.7, I report a comparison of the pre-training strategy discussed versus uniform random initialization of the circuit parameters, for Hamiltonians with $h_z = -1$ and $h_{xx} = 1$. As can be seen in Figures 5.7(a, b), pre-training yields improvements by several orders of magnitude in terms of the error

$$\epsilon_{\mathrm{LTFIM}}(\boldsymbol{\theta},\boldsymbol{\phi}) = \frac{\langle \mathbf{0}| U^\dagger(\boldsymbol{\theta},\boldsymbol{\phi}) H_{\mathrm{LTFIM}} U(\boldsymbol{\theta},\boldsymbol{\phi}) |\mathbf{0}\rangle - E_{\min}}{\|H_{\mathrm{LTFIM}}\|_{\mathrm{HS}_1}}, \tag{5.52}$$

where $E_{\min}$ is the ground state energy of $H_{\mathrm{LTFIM}}$ obtained by exact diagonalization. In Figure 5.7(a), I compare these errors as a function of the longitudinal field strength $|h_x|$. The prepared state resulting from the pre-training strategy (with errors depicted as an orange dashed line) becomes a poorer approximation to the ground state as $|h_x|$ increases, and even sometimes at par with random initialization

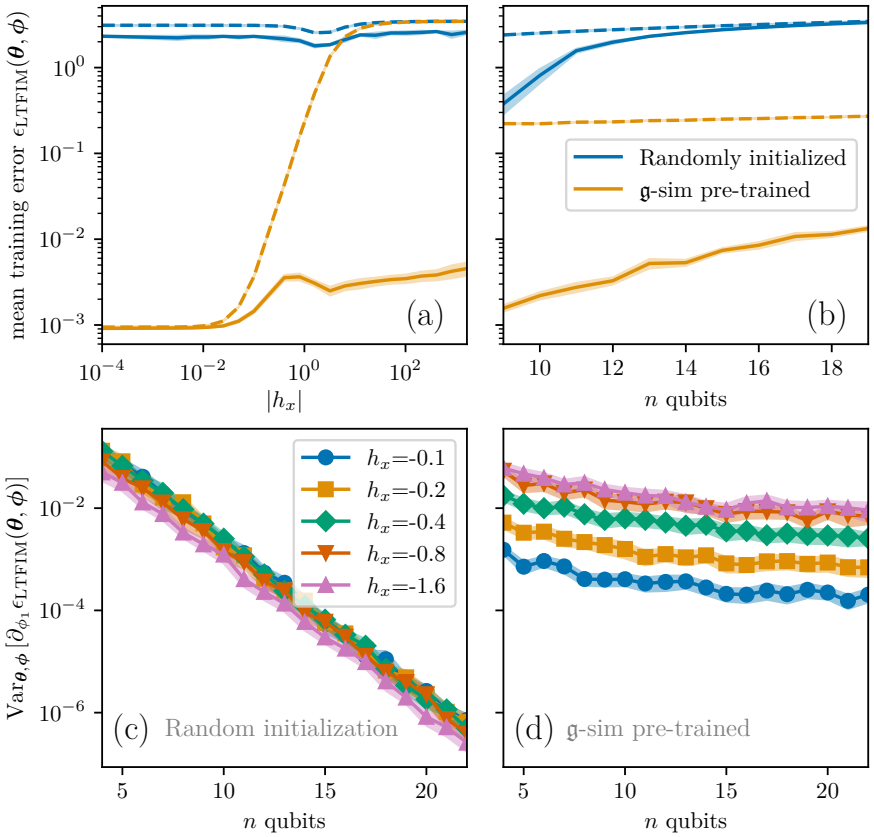



**Figure 5.7: Pre-training VQE for the LTFIM using $\mathfrak{g}$-sim. (a, b)** Mean training error $\epsilon_{\text{LTFIM}}(\boldsymbol{\theta}, \boldsymbol{\phi})$ (Equation 5.52) for VQE with random initialization (blue) and pre-trained with $\mathfrak{g}$-sim (orange), (a) at $n = 12$ qubits and varying longitudinal field strengths $h_x$ and (b) at field strength $h_x = -1.0$ and varying system sizes $n$. Dashed lines represent initial ansatz configurations, while solid lines represent trained ansätze. **(c, d)** Comparison of gradient variances at varying system size $n$ and longitudinal field strengths $h_x$ for (c) uniform random parameter initialization and (d) $\mathfrak{g}$-sim pre-training. Shaded bars represent bootstrapped 95% confidence intervals.

(blue dashed line). Still, after the final step of optimization (with errors depicted as plain lines) I find that pre-training consistently enables accurate ground state preparation of $H_{\text{LTFIM}}$. Across all the values of $h_x$ studied, one can see that the pre-training strategy is the most favorable initialization. In Figure 5.7(b), I report final errors scaling with the system size $n$, noting that the randomly initialized circuits quickly become untrainable as $n$ increases while the pre-trained ansatz only exhibits mild decline in trainability.

In addition to these improved training errors, I observe a mitigation of the barren plateau effect. Due to the exponential dimension of $\mathfrak{g}_{\text{LTFIM}}$, in the case of

random initialization one can guarantee gradient variance scaling as [213]

$$\mathrm{Var}_{\boldsymbol{\theta},\boldsymbol{\phi}}\left[\partial_{\phi_m}\epsilon_{\mathrm{LTFIM}}(\boldsymbol{\theta},\boldsymbol{\phi})\right] \in \mathcal{O}\left(\frac{1}{2^n}\right), \tag{5.53}$$

thus necessitating $\Theta(2^n)$ circuit repetitions to distinguish small gradient values from statistical shot noise. I verify this numerically in Figure 5.7(c) which, over varied values of the field $h_x$, showcases exponentially vanishing gradients. However, in the case of $\mathfrak{g}$-sim pre-training, one can see in Figure 5.7(d) that the gradient variances vanish at a much slower rate in $n$, effectively mitigating appearance of the barren plateau effect and thus extending the scalability of VQE on this system.

#### 5.4.2.3 Pre-training QAOA

As I reviewed in Section 2.2.2, the quantum approximate optimization algorithm (QAOA) is a VQA that attempts to solve combinatorial optimization problems [23]. In this section, I consider its use to approximate solutions of MaxCut problems. It has been reported that for most choices of target graph $G$, the Lie algebra $\mathfrak{g}$ associated with Equation 2.7 will have $\dim(\mathfrak{g}) \in \Omega(2^n)$ [213, 378]. Hence, to apply this pre-training strategy, one needs to identify an algebra with polynomial dimension that is related to the original problem. For the circuit of Equation 2.7, for $n$ vertices this is known to be the case for the path graph $G = P_n$, cycle graph $G = C_n$ and complete graph $G = K_n$ [378]. For the purposes of this chapter, I focus on the path graph $P_n$, for which the Lie algebra $\mathfrak{g}_{\mathrm{QAOA},P_n} = \mathrm{span}\langle\{iH_{P_n}, iH_M\}\rangle_{\mathrm{Lie}}$ is such that (up to a change of basis, where the Pauli $\sigma^x$ and $\sigma^z$ are interchanged) $\mathfrak{g}_{\mathrm{QAOA},P_n} \subset \mathfrak{g}_0$, and therefore has $\dim(\mathfrak{g}_{\mathrm{QAOA},P_n}) \in \mathcal{O}(\mathrm{poly}(n))$. Hence, I again use the structure constants of $\mathfrak{g}_0$, and start by training such circuit for $p = \dim(\mathfrak{g}_{\mathrm{QAOA},P_n})$.

The pre-trained parameters $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ can then be used to initialize the modified ansatz

$$U_G(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}) = \prod_{l=1}^{p}\left[e^{-i\beta_l H_M}e^{-i\gamma_l H_{P_n}}e^{-i\alpha_l H_G}\right] \tag{5.54}$$

for any general graph $G \neq P_n$, with the new parameters initialized to $\boldsymbol{\alpha} = (0, 0, \ldots)$. I then train the parameters $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$ to minimize the expectation value of $H_G$ — in

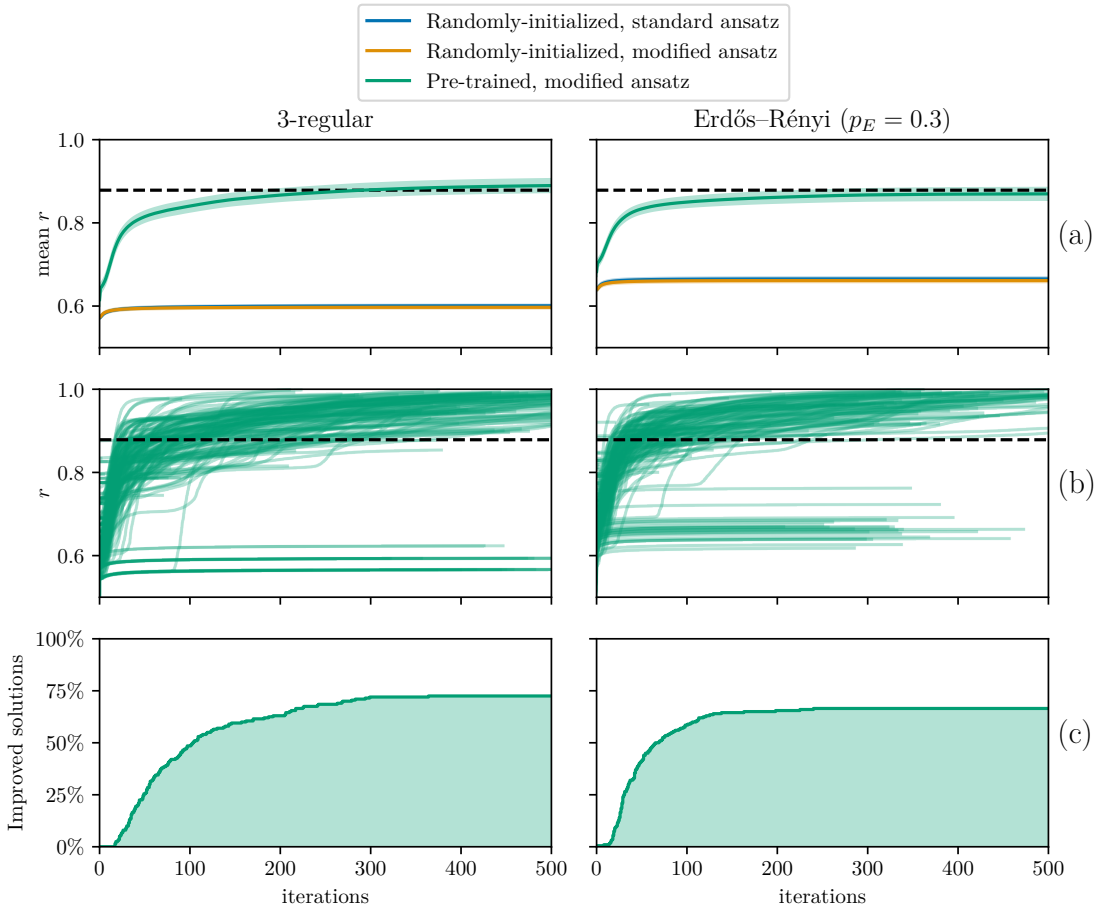



**Figure 5.8: Pre-training QAOA using $\mathfrak{g}$-sim.** Performances of QAOA on MaxCut at $n = 16$ qubits for the standard randomly-initialized QAOA ansatz (blue), the modified randomly-initialized ansatz (orange), and the modified ansatz pre-trained with $\mathfrak{g}$-sim (green). Comparisons are made across ensembles of 200 random 3-regular graphs (left panels) and 200 Erdös-Réyni graphs with edge probability $p_E = 0.3$ (right panels). I compare **(a)** the mean approximation ratios $r$ (shaded bars are bootstrapped 95% confidence intervals), **(b)** individual training trajectories, and **(c)** the fraction of solutions outperforming the GW threshold (horizontal dashed black line).

a scalable approach, one would use a quantum device for this step, however for ease of modelling I again perform this step with TensorFlow Quantum [105].

I study this pre-training procedure at $n = 16$ qubits on random 3-regular graphs and Erdös-Rényi graphs with edge probability $p_E = 0.3$. I present the results of this numerical experiment in Figure 5.8, showing that pre-training significantly outperforms randomly-initialized QAOA circuits. In terms of the mean approximation ratio $r$ (Figure 5.8(a)), both randomly-initialized strategies vastly under-perform the GW threshold $r_{\text{GW}}$ (horizontal dashed line), while the pre-

training strategy achieves comparable average performance. More strikingly, when looking at details of the approximation ratios $r$ of the individual graph problems (Figure 5.8(b)), I find that a majority of individual graphs achieve an approximation ratio $r > r_{\mathrm{GW}}$, and the rest for which it is at par with random initialization.

Overall, the fraction of solutions that improve on the GW threshold (Figure 5.8(c)) converges to 72.5% for 3-regular graphs, and 66.5% for $p_E = 0.3$ Erdös-Rényi graphs. On the other hand, no randomly-initialized circuits achieved better than this threshold. This suggests that $\mathfrak{g}$-sim pre-training is advantageous for QAOA on a substantial fraction of random graphs, even with this relatively crude initialization strategy. Further improvements could likely be achieved by more carefully aligning the path graph $P_n$ along $G$, adaptively transforming the ansatz and cost, or resorting to another polynomially-size algebra for the pre-training.

### 5.4.3 Circuit synthesis

Until now, I have been concerned exclusively with tasks of state-preparation. I now address more difficult problems of *unitary compilation*. In this section, I seek to use $\mathfrak{g}$-sim to identify the parameters of an ansatz circuit $U(\boldsymbol{\theta}) \in \mathcal{G}$ of the form in Equation 5.1 to implement a target unitary $V \in \mathcal{G}$. Although this section's methods require both the target and circuit to belong to a unitary Lie group whose associated Lie algebra is of polynomial dimension, there is still a clear purpose to this task. While naïve applications of $\mathfrak{g}$-sim are restricted to certain initial states and observables, circuit synthesis enables one to surpass this. Using the methods of this section, one could use $\mathfrak{g}$-sim to classically compile *any* unitary in $\mathcal{G}$ to a polynomial-gate-count circuit, and then implement it on a quantum computer to evaluate the evolution of *any* observable or state, even those not supported by $\mathfrak{g}$. More generally, the unitary to be synthesized could be part of a larger protocol such that the initial state or observable may not even be known beforehand. While some applications of $\mathfrak{g}$-sim (e.g. supervised QML as outlined in Section 5.4.4) require classical representations of states (and therefore may require some data to be collected from a quantum device), the methods of this section are

entirely classical. Finally, I note that there may be utility in compiling random unitaries corresponding to polynomially sized Lie algebras, as sampling from the output of some of these circuits have strong hardness guarantees that can be used for compelling demonstrations of quantum advantage [388].

I outline the scheme for unitary compilation with $\mathfrak{g}$-sim in Section 5.4.3.1. I then probe its convergence properties for random targets in Section 5.4.3.3, displaying polynomial optimization resource requirements. However, as discussed in Section 5.4.3.4, given that this is grounded in a representation-theoretic framework, issues of faithfulness in adjoint mappings can complicate compilation procedures. I present and demonstrate a strategy to overcome this. With these tools in hand, I demonstrate a highly practical use of unitary compilation with $\mathfrak{g}$-sim: refining circuits to outperform naïve Trotterization for dynamical evolution in Section 5.4.3.5.

#### 5.4.3.1 Variational compilation of unitaries in $\mathfrak{g}$-sim

In Section 2.2.2, I described variational circuit compilation, wherein one typically seeks to train the parameters of an ansatz circuit $U(\boldsymbol{\theta})$ to approximate some target unitary $V$. Existing techniques for compiling variational circuits to target unitaries usually seek to minimize the Hilbert-Schmidt test cost function (Equation 2.9), which has computational complexity with $\mathcal{O}(2^{2n})$ and is not classically scalable in general. Utilizing this would require evaluation on a quantum computer for even modest system sizes. In pre-existing frameworks, evaluation of Equation 2.9 would typically rely on the Hilbert-Schmidt test [196, 389] or on estimation via random-state sampling [192, 197]. In any case, these assume that one can implement the target $V$ on the compiling device in the first place, which is often unrealistic: if one already has a quantum computer capable of implementing $V$, then the need for variational compilation is often moot.

In contrast, given a description of $V \in \mathcal{G}$ with dimension of the associated Lie algebra $\dim(\mathfrak{g}) \in \mathcal{O}(\text{poly}(n))$, the present approach to compilation can be performed entirely on a classical computer with $\mathcal{O}(\text{poly}(n))$ resources. To that

intent, I propose the loss function

$$\begin{aligned}\mathcal{L}_{\mathfrak{g}}(U(\boldsymbol{\theta}), V) &\equiv \frac{1}{2\dim(\mathfrak{g})}\|\bar{\boldsymbol{U}}(\boldsymbol{\theta}) - \bar{\boldsymbol{V}}\|_{\mathrm{HS}}^2 \\ &= 1 - \frac{1}{\dim(\mathfrak{g})}\,\mathrm{Re}\left[\mathrm{Tr}\left(\bar{\boldsymbol{U}}^T(\boldsymbol{\theta})\bar{\boldsymbol{V}}\right)\right], \end{aligned} \tag{5.55}$$

where $\|\cdot\|_{\mathrm{HS}}$ is the Hilbert-Schmidt norm and $\bar{\boldsymbol{U}}(\boldsymbol{\theta}) \equiv \Phi_{\mathcal{G}}^{\mathrm{Ad}}(U(\boldsymbol{\theta}))$, $\bar{\boldsymbol{V}} \equiv \Phi_{\mathcal{G}}^{\mathrm{Ad}}(V)$ are the adjoint representations of $U(\boldsymbol{\theta})$ and $V$, respectively.

Intuitively, $\mathcal{L}_{\mathfrak{g}}$ measures how accurately $U(\boldsymbol{\theta})$ approximates the evolution of the expectation values $\langle G_\alpha \rangle$ under evolution by $V$ for all initial states. Although the Hilbert-Schmidt loss $\mathcal{L}_{\mathrm{HST}}$, in Equation 2.9, and its adjoint-space counterpart $\mathcal{L}_{\mathfrak{g}}$ (5.55) are similar in form, I highlight that $\mathcal{L}_{\mathfrak{g}}$ is sensitive to global phase differences between $\bar{\boldsymbol{U}}(\boldsymbol{\theta})$ and $\bar{\boldsymbol{V}}$, whereas $\mathcal{L}_{\mathrm{HST}}$ is not sensitive to global phase differences between $U(\boldsymbol{\theta})$ and $V$. This is because global phase differences in the full Hilbert space are nonphysical, whereas global phase differences in the adjoint representation correspond to a sign flip on all basis observables, which is indeed physical. As I discuss in Section 5.4.3.4, related subtleties can compromise our ability to perform faithful compilation if not handled carefully. I defer this point for now, first assessing convergence of optimizations with the loss in Equation 5.55 for random target unitaries in Section 5.4.3.3, and resolving the faithfulness issue in Section 5.4.3.4.

#### 5.4.3.2 Compilation cost function gradients

Throughout this thesis, the importance of gradient-based optimization of VQAs has been clear. For unitary compilation with $\mathfrak{g}$-sim, the use of gradient-based optimizers requires the efficient calculation of $\partial\mathcal{L}_{\mathfrak{g}}/\partial\theta_m$; in this section, I outline how this can be done efficiently. Differentiating Equation 5.55, one obtains

$$\frac{\partial\mathcal{L}_{\mathfrak{g}}}{\partial\theta_m} = \frac{1}{\dim(\mathfrak{g})}\,\mathrm{Im}\left[\mathrm{Tr}\left[\bar{\boldsymbol{V}}^T\bar{\boldsymbol{U}}_{m+1:M}\bar{\boldsymbol{H}}_m\bar{\boldsymbol{U}}_{1:m}\right]\right], \tag{5.56}$$

where I have used properties of the ansatz structure (5.1), and $\bar{\boldsymbol{U}}_{n:m}^T \equiv \bar{\boldsymbol{U}}_n^T\bar{\boldsymbol{U}}_{n+1}^T\ldots\bar{\boldsymbol{U}}_m^T$. Noting that the matrix terms in Equation 5.56 have the same recurrent form as Equation 5.30, one may compute the full gradient vector in $\mathcal{O}(N\dim(\mathfrak{g})^3)$ by a

straightforward modification of Algorithm 2 (with matrix-matrix multiplication instead of matrix-vector multiplication).

For efficient implementation, care should be taken with the order of operations when evaluating Equations 5.55 and 5.56, particularly endeavouring to avoid any dense-dense matrix multiplication. For example, in Equation 5.56, one should store the products $(\bar{\boldsymbol{U}}_{m+1:m})^T\bar{\boldsymbol{V}}$ and $\bar{\boldsymbol{U}}_{1:m}$ as dense matrices, use sparse representations of the $\bar{\boldsymbol{U}}_m$ when applying the iterative step in Algorithm 2, and use sparse representations of $\bar{\boldsymbol{H}}_m$ when evaluating each element of the gradient. I found that with this order of operations, the evaluation is substantially accelerated by the use of sparse-dense matrix multiplication algorithms, and further improved by the use of GPU acceleration (e.g. SpDM with cuSPARSE). Finally, the trace in Equation 5.56 of the product of two dense matrices ($\bar{\boldsymbol{V}}^T\bar{\boldsymbol{U}}_{m+1:M}$ and $\bar{\boldsymbol{H}}_m\bar{\boldsymbol{U}}_{1:m}$) can be evaluated in a dense $\mathcal{O}(\dim(\mathfrak{g})^2)$ operation as

$$\mathrm{Tr}\Big[\bar{\boldsymbol{V}}^T\bar{\boldsymbol{U}}_{m+1:M}\bar{\boldsymbol{H}}_m\bar{\boldsymbol{U}}_{1:m}\Big] = \sum_{\alpha\beta}\Big[((\bar{\boldsymbol{U}}_{m+1:m})^T\bar{\boldsymbol{V}})\circ(\bar{\boldsymbol{H}}_m\bar{\boldsymbol{U}}_{1:m})\Big]_{\alpha\beta}\,, \tag{5.57}$$

where $\circ$ denotes the Hadamard product. This avoids costly $\mathcal{O}(\dim(\mathfrak{g})^3)$ dense-dense matrix multiplication entirely, which would otherwise be by far the most expensive operation in this procedure.

#### 5.4.3.3 Compiling random unitaries

I first benchmark my compilation scheme against random unitaries in $\mathcal{G}_0 = e^{\mathfrak{g}_0}$, which is the most general and demanding task for this scheme. Any unitary in $\mathcal{G}$ may be written in the form

$$V = e^{-iT\sum_\alpha(\vec{w})_\alpha G_\alpha}, \tag{5.58}$$

where I fix $|\vec{w}|_2 = 1$ such that $T$ parametrizes the duration of the effective corresponding dynamics. To generate the weights uniformly, I sample a matrix from a Haar distribution over the orthogonal group $O(\dim(\vec{w}))$ and set the weight vector $\vec{w}$ equal to one of its columns (potentially padding with zeros to account for elements of the basis not present in the sum). In general, Equation 5.58 involves highly

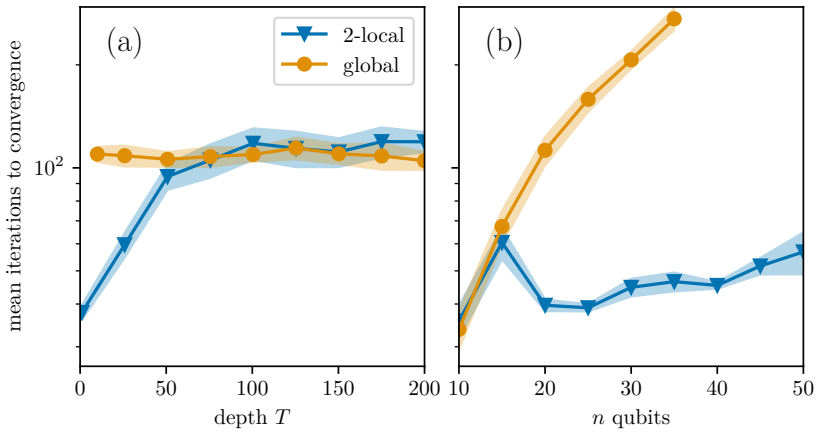



**Figure 5.9: Scaling properties of circuit compilation with 𝔤-sim.** I compare the mean number of iterations to converge to a loss (defined in Equation 5.55) $\mathcal{L}_{\mathfrak{g}} < 10^{-3}$ for random target unitaries (defined in Equation 5.58) with varied levels of locality, including 2-local and global generators. Shaded bars depict bootstrapped 95% confidence intervals. **(a)** Results for varying duration $T$ at $n = 20$ qubits. **(b)** Results for varying system size $n$ at duration $T = 10$. In both cases, the scaling of the iterations number is found to be polynomial in $n$ and $T$.

non-local interactions, since many elements of $\mathfrak{g}_0$ are non-local Pauli strings (see Equation 5.38). By excluding greater-than-$k$-local $G_\alpha$ from Equation 5.58, one can also refine study to the highly relevant case of $k$-local Hamiltonian dynamics, and I consider compilation of unitaries based on both local and nonlocal interactions.

I test this scheme by compiling global and 2-local dynamics, both with the 2-local ansatz of Figure 5.5(a). I build the ansatz with $L = \lceil 2\dim(\mathfrak{g})/K \rceil$ layers where $K$ is the number of generators, which is guaranteed to be overparametrized. In Figure 5.9, I show that the compilation of random unitaries $V \in \mathcal{G}_0$ performs and scales well. At a fixed $n = 20$ qubits (Figure 5.9(a)), the convergence requirements (measured as the number of optimization iterations required for convergence to a fixed error tolerance) plateau to a constant value irrespective of $T$, suggesting that my approach allows fixed-circuit-depth Hamiltonian simulation for arbitrary evolution time. I note that fixed-depth dynamics were obtained in Ref. [382] on related systems, but with different methods (based on an explicit Cartan diagonalization) and only targeting local dynamics. At fixed $T = 10$ (Figure 5.9(b)), the number of steps to converge appears constant in $n$ for 2-local targets, and polynomial in $n$ for global targets. However, for smaller values of $T$, a different convergence behavior is seen

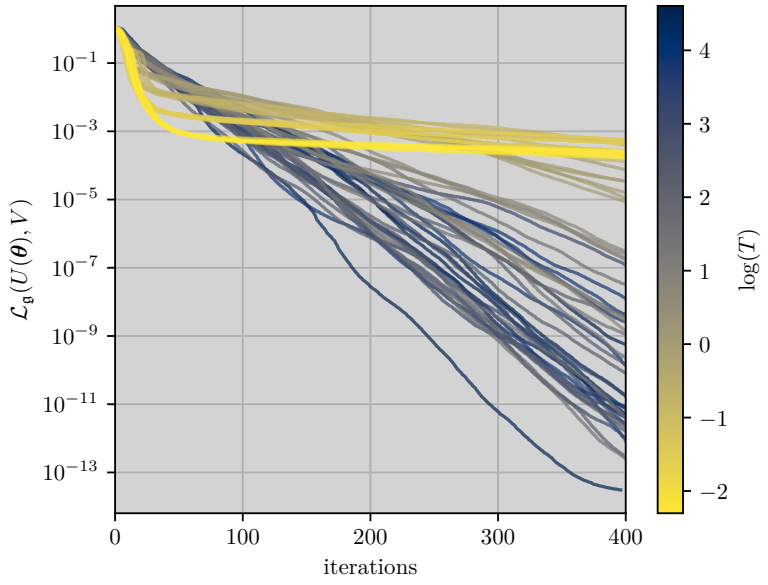



**Figure 5.10: Compiling short-duration dynamics for global Hamiltonians.** For a system of $n = 20$ qubits, I train an overparametrized (27-layer) ansatz to minimize $\mathcal{L}_{\mathfrak{g}}$, with a target unitary generated by a normalized global Hamiltonian. At larger values of $T$, the expected exponential convergence is observed. At $T \approx 1$, a transition is observed to a regime with poorer trainability.

when compiling global dynamics. After a brief initial period where $\mathcal{L}_{\mathfrak{g}}$ vanishes exponentially in the number of iterations, a transition to very slow convergence is observed. This is demonstrated in Figure 5.10. We have not yet been able to determine an explanation for this phenomenon — it is highly counter-intuitive, since one would expect *a priori* that longer-duration global dynamics would be the most difficult to capture with a local ansatz, yet that appears to be easy in comparison.

Overall, this demonstrates that the methods I have introduced can efficiently minimize the loss $\mathcal{L}_{\mathfrak{g}}$ on an overparametrized ansatz for a diverse range of random unitaries in $\mathcal{G}_0$ with varying locality, system size $n$, and dynamics duration $T$, providing a strong foundation for my compilation scheme. In the following section, I turn my attention to the faithfulness of this compilation scheme — under what conditions is minimizing $\mathcal{L}_{\mathfrak{g}}$ sufficient to faithfully compile a target unitary $V$?

#### 5.4.3.4 Faithfulness of compilation

Although I observed consistent convergence of the training with respect to the adjoint cost function $\mathcal{L}_{\mathfrak{g}}$ (Figure 5.9), one should question whether minimizing $\mathcal{L}_{\mathfrak{g}}$

is sufficient for unitary compilation. It can be seen from Equation 5.55 that $\mathcal{L}_{\mathfrak{g}}$ is a faithful loss function for unitary training iff $\Phi_{\mathcal{G}}^{\mathrm{Ad}}$ is a faithful representation. This is not guaranteed, and to understand why, first note the definition of the center of a group $Z(\mathcal{G})$ of $\mathcal{G}$:

**Definition 9** (Center of a group)**.** *The center $Z(\mathcal{G})$ of a group $\mathcal{G}$ is the subset of $\mathcal{G}$ that simultaneously commutes with all elements of $\mathcal{G}$. That is,*

$$Z(\mathcal{G}) \equiv \{W \in \mathcal{G} \mid \forall U \in \mathcal{G}, WU = UW\}. \tag{5.59}$$

Using this definition, one can prove that the group adjoint representation $\Phi_{\mathcal{G}}^{\mathrm{Ad}}$ need not be faithful even if the algebra representation $\Phi_{\mathfrak{g}}^{\mathrm{ad}}$ is.

**Statement 5** (Faithfulness of $\Phi_{\mathcal{G}}^{\mathrm{Ad}}$.)**.** *Suppose that $\Phi_{\mathcal{G}}^{\mathrm{Ad}}$ is an adjoint representation of a dynamical Lie group $\mathcal{G}$, induced by a faithful adjoint representation $\Phi_{\mathfrak{g}}^{\mathrm{ad}}$ of the corresponding dynamical Lie algebra $\mathfrak{g}$. Then, $\Phi_{\mathcal{G}}^{\mathrm{Ad}}$ is faithful if and only if the center of $\mathcal{G}$ is trivial, i.e., $Z(\mathcal{G}) = \{\mathbb{1}\}$.*

*Proof.* The forward direction can be proven by contradiction. Suppose that the center of $\mathcal{G}$ is trivial, and $\Phi_{\mathcal{G}}^{\mathrm{Ad}}$ is not faithful. Then, there exists $U, V \in \mathcal{G}$ with $U \neq e^{i\theta}V$ for all $\theta$ such that $\Phi_{\mathcal{G}}^{\mathrm{Ad}}(U) = \Phi_{\mathcal{G}}^{\mathrm{Ad}}(V)$. However, since the adjoint representation is a homomorphism, it follows that $\Phi_{\mathcal{G}}^{\mathrm{Ad}}(U^{-1}V) = \Phi_{\mathcal{G}}^{\mathrm{Ad}}(\mathbb{1})$. By assumption, $UV^{-1} \neq e^{i\theta}\mathbb{1}$ for all $\theta$, and since its adjoint representation is the identity it follows that the generator of $UV^{-1}$ is a nontrivial element of $\mathcal{G}$ which commutes with all elements of $\mathcal{G}$. Consequently, $UV^{-1} \in Z(\mathcal{G})$, which is a contradiction and proves the forward direction. Similarly, for the backward direction, let $W \in Z(\mathcal{G})$ be a nontrivial element of the center of $\mathcal{G}$. Then, for any $U \in \mathcal{G}$, it follows that

$$\Phi_{\mathcal{G}}^{\mathrm{Ad}}(WU) = \Phi_{\mathcal{G}}^{\mathrm{Ad}}(UW) = \Phi_{\mathcal{G}}^{\mathrm{Ad}}(U), \tag{5.60}$$

where the last step used the fact that $\Phi_{\mathcal{G}}^{\mathrm{Ad}}$ is a homomorphism and that $W \in Z(\mathcal{G})$. Therefore, it follows that $\Phi_{\mathcal{G}}^{\mathrm{Ad}}$ is not faithful, completing the proof. □

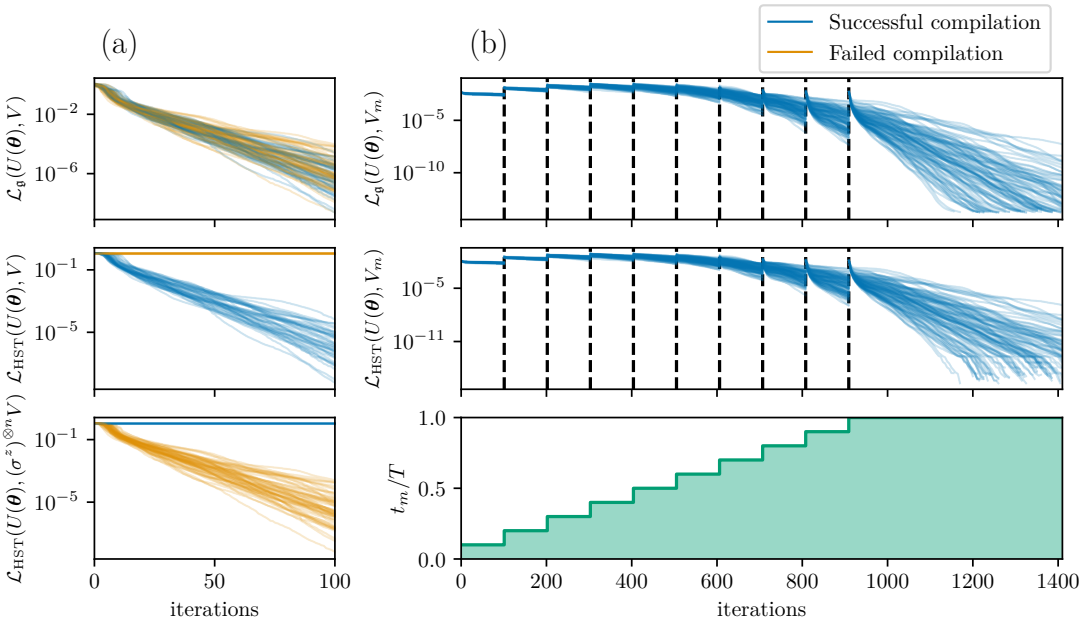



**Figure 5.11: Successfully compiling circuits in an algebra with non-trivial group center elements. (a)** Although the classical loss function $\mathcal{L}_{\mathfrak{g}}(U(\boldsymbol{\theta}), V)$ can be efficiently minimized (Figure 5.9), this is a necessary but insufficient condition to minimize the Hilbert-Schmidt loss function $\mathcal{L}_{\mathrm{HST}}$, and thus to faithfully compile a target unitary $V$. We train a linear-depth ansatz $U(\boldsymbol{\theta})$ (see Fig. 5.5(a)) initialized with uniform random parameters to minimize $\mathcal{L}_{\mathfrak{g}}(U(\boldsymbol{\theta}), V)$ at $T = 1$ and $n = 10$. Although $\mathcal{L}_{\mathfrak{g}}$ is successfully minimized in all instances (top panel), $\mathcal{L}_{\mathrm{HST}}$ is only minimized in approximately half the instances, and rather *maximized* otherwise (middle panel). In all cases where the compilation fails, the optimized ansatz instead approximates $(\sigma^z)^{\otimes n} V$ (bottom panel). **(b)** The scheme outlined in Sec. 5.4.3.4 successfully rectifies this issue. The ansatz is initialized at $\boldsymbol{\theta} = \mathbf{0}$, and compiled to a sequence of intermediate targets $V_m$ corresponding to increased $t_m$ from 0 to T (bottom panel), with the parameters obtained from compiling a target used to initialize the next . For sufficiently small time steps $\Delta t_m$, successfully minimizing $\mathcal{L}_{\mathfrak{g}}$ (top panel) ensures minimization of $\mathcal{L}_{\mathrm{HST}}$ (middle panel) and thus faithful compilation of the final target $V$.

Thus, faithful representation $\Phi_{\mathfrak{g}}^{\mathrm{ad}}$ of the Lie algebra does not guarantee faithful representation $\Phi_{\mathcal{G}}^{\mathrm{Ad}}$ of the Lie group: one must determine the center of the group $Z(\mathcal{G})$, and account for its presence in the compilation scheme. For $W \in Z(\mathcal{G})$, in the adjoint representation one cannot distinguish $V$ from $WV$, since $\Phi_{\mathcal{G}}^{\mathrm{Ad}}(W) = \Phi_{\mathcal{G}}^{\mathrm{Ad}}(\mathbb{1})$. Throughout this chapter I mostly use $\mathfrak{g}_0$, which is indeed afflicted by this: the algebra $\mathfrak{g}_0$ itself is centerless, but the corresponding group center is $Z(e^{i\mathfrak{g}_0}) = \{(\sigma^z)^{\otimes n}, \mathbb{1}\}$ (up to global phase).

This issue indeed manifests seen in my numerics. In the first row of Figure 5.11(a),

I report systematic success in minimizing $\mathcal{L}_{\mathfrak{g}}$. However, as can be distinguished by the full Hilbert-Schmidt loss function $\mathcal{L}_{\mathrm{HST}}$, only half of the optimizations yield the correct target $V$ while the other half rather yield the unitary $(\sigma^z)^{\otimes n}V$ (second and third row), which is indistinguishable in the adjoint representation. To guarantee successful compilation of $V$, one must either ensure convergence to the manifold of correct solutions, or be able to flag when an error has occurred such that it could be corrected (i.e. by applying an additional unitary $(\sigma^z)^{\otimes n}$). I now detail a strategy achieving the former.

Any unitary target $V \in \mathcal{G}$ may be written in the form $V = e^{-iTH}$ with $iH \in \mathfrak{g}$ as in Equation 5.58. Rather than directly aiming for the compilation of $V$, I consider a family of intermediary targets $V_m = e^{-iHt_m}$ for increasing steps $t_m \in [0, T]$, with $t_0 = 0$. I then solve the corresponding compilation problems $\boldsymbol{\theta}_m = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{\mathfrak{g}}(U(\boldsymbol{\theta}), V_m)$ sequentially, with $\boldsymbol{\theta}$ initialized to $\boldsymbol{\theta}_{m-1}$ at each step. For sufficiently small $\Delta t_m = t_m - t_{m-1}$, it is expected that no jump between solution manifolds will occur: in the limit of small $\Delta t$, this is asymptotically guaranteed. Crucially, given that $V_0 = \mathbb{1}$, one can ensure initialization in the correct manifold by setting $\boldsymbol{\theta}_0 = \mathbf{0}$ such that $U(\boldsymbol{\theta}_0) = \mathbb{1}$ (rather than $(\sigma^z)^{\otimes n}$). I confirm viability of the proposed scheme numerically, via the numerical experiment depicted in Figure 5.11(b). For all the random unitaries $V$ assessed, the optimization concludes with parameters accurately replicating the desired target, as evidenced by the low values of $\mathcal{L}_{\mathrm{HST}}(U(\boldsymbol{\theta}), V) < 10^{-6}$. This demonstrates that faithful compilation is possible even when the cost function $\mathcal{L}_{\mathfrak{g}}$ is not faithful, ensuring that compilation with $\mathfrak{g}$-sim can lead to robust and reliable unitary synthesis.

#### 5.4.3.5 Application to dynamical simulation

As demonstrated in Section 5.4.3.3, given a Hamiltonian supported by a Lie algebra $\mathfrak{g}$ with $\dim(\mathfrak{g}) \in \mathcal{O}(\mathrm{poly}(n))$, my scheme enables compilation in polynomial-depth circuits of the corresponding time-evolution operators. This allows the study of dynamics of observables *not* supported by $\mathfrak{g}$-sim and arbitrary states, which are in general not classically tractable. Here I demonstrate the utility of this

scheme by synthesizing circuits for the Hamiltonian time-evolution of the TFXY spin chain (5.42) with open boundary conditions and random local magnetic fields $b_j$ drawn from $N(0, \xi^2)$. This Hamiltonian supports the phenomenon of Anderson localization [390], and has previously been utilized to demonstrate related compilation techniques based on Cartan decomposition [382].

I seek to train an ansatz $U(\boldsymbol{\theta}_{t_m})$ to approximate $V_{t_m} \equiv e^{-it_m H_{\mathrm{TFXY}}}$ at a range of discrete times $t_m \in [0, 200]$. For each $t_m$, the ansatz has a structure

$$U(\boldsymbol{\theta}) = \prod_{l=1}^{L} \Bigg[ \prod_{j=1}^{n-1} e^{-i\theta_{l,(n+2j+1)}\sigma_j^x \sigma_{j+1}^x} e^{-i\theta_{l,(n+2j)}\sigma_j^y \sigma_{j+1}^y} \prod_{j=1}^{n} e^{-i\theta_{l,j}\sum_{j=1}^{n}\sigma_j^z} \Bigg]. \tag{5.61}$$

One can verify that the dynamical Lie algebra associated with this circuit is again $\mathfrak{g}_0$ as defined in Equation 5.38, meaning that I can again utilize the structure constants I have already pre-computed. At this point I note that while $U(\boldsymbol{\theta})$ has the exact same structure as that of a first-order Trotterization of any of the $V_{t_m}$ unitaries, each term of the Hamiltonian is associated with an independent trainable parameter in $U(\boldsymbol{\theta})$. This fact is important, since $U(\boldsymbol{\theta})$ is *not* trying to learn a Trotterized version of $V_{t_m}$, but the *exact* unitary $V_{t_m}$ at time $t_m$. In fact, because I use $\mathfrak{g}$-sim, I do not need to ever perform a Trotterization of the target unitary, as $V_{t_m}$ can be compiled exactly for all evolution times $t_m$. This is due to the fact that one can efficiently compute the adjoint representation $\Phi_{\mathcal{G}}^{\mathrm{Ad}}(V_{t_m}) = e^{-it_m \Phi_{\mathfrak{g}}^{\mathrm{ad}}(H_{\mathrm{TFXY}})}$ on a classical computer, using matrix exponentiation of the $\dim(\mathfrak{g}) \times \dim(\mathfrak{g})$ matrix $\Phi_{\mathfrak{g}}^{\mathrm{ad}}(H_{\mathrm{TFXY}})$. This is advantageous compared to variational compilation schemes for time evolution that necessitate the target to be implementable on a quantum device in the first place: as I stated earlier, if one can already implement the target unitary, then the need for variational compilation is much more tenuous. In contrast, other unitary variational compilation schemes often approximate the target by means of a Trotter approximation [188–193], which therefore introduces approximation errors to the target before the recompilation procedure has even begun.

For the sake of concreteness, I focus on the dynamics of a single-spin-flip initial state $|{\downarrow}{\uparrow}{\uparrow}{\uparrow}{\uparrow}{\uparrow}{\uparrow}{\uparrow}{\uparrow}{\uparrow}{\uparrow}{\uparrow}\rangle$ for $n = 12$ qubits. In the absence of a magnetic field

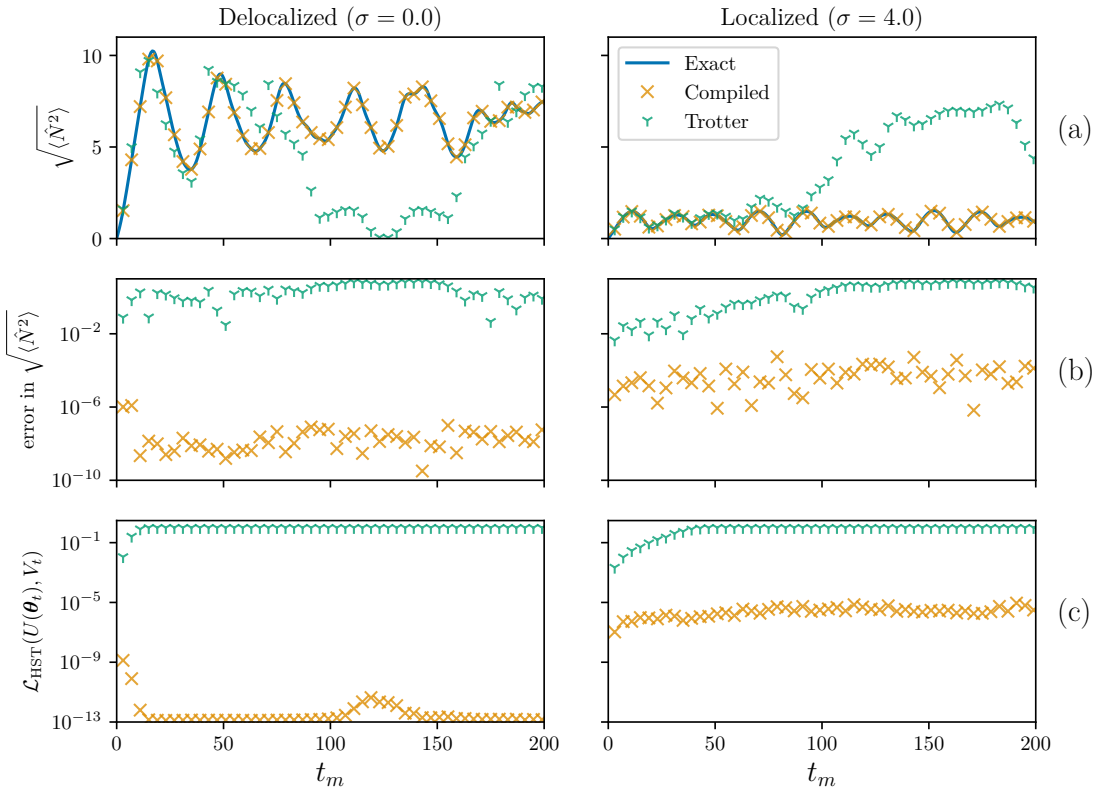



**Figure 5.12: Compiling Hamiltonian dynamics with 𝔤-sim.** Dynamics of a single-spin-flip excitation in a TFXY spin chain (Equation 5.42 with $n = 12$) in the absence of magnetic fields ($\xi = 0$, left panels) and with random fields inducing Anderson localization ($\xi = 4$, right panels). The exact dynamics (blue) is compared to the dynamics yielded by our compilation scheme (orange X), and first-order Trotterization (green Y). The compilation scheme accurately showcases Anderson localization, reproducing the RMS position $\sqrt{\hat{N}^2}$ of the excitation (a) with low errors (b), while Trotterization quickly diverges beyond small simulation times. Furthermore, comparison of the compiled unitaries to the exact time-evolution operators by means of the Hilbert-Schmidt test $\mathcal{L}_{\mathrm{HST}}$, show small errors at all times (c), guaranteeing faithful dynamics of all observables.

($b_j = 0$ for all $j$) this excitation diffuses throughout the system, but a disordered field ($\xi > 0$) restricts this diffusion (this is known Anderson localization [390] in condensed matter literature). Following the example of Ref. [382], I study the position operator for the excitation

$$\hat{N} = \sum_{j=1}^{n} (j-1) \frac{\mathbb{1} - \sigma_j^z}{2}, \tag{5.62}$$

whose moments provide a key signature of Anderson localization. In particular, for this system the $p$-th moment $\langle |\hat{N}|^p \rangle$ admits a time-independent upper bound [382, 391]. I note that $\hat{N}^2$ is expressible as a linear combination of elements in the

algebra ($\sigma_j^z$) and product of elements in the algebra ($\sigma_j^z \sigma_{j'}^z$), and thus is precisely of the form in Equation 5.16. Moreover, given that the algebra is composed of Pauli operators, and since the initial state is separable, one can readily compute the entries of the vector $\boldsymbol{e}^{(\text{in})}$ and the matrix $\boldsymbol{E}^{(\text{in})}$. Hence, the $p = 2$ moment can be efficiently computed with $\mathfrak{g}$-sim. Higher-order moments can be computed (Remark 1), albeit with complexity growing exponentially in $p$. Furthermore, I stress that my choice of a classically-tractable system size ($n = 12$ qubits) is purely to enable me to compute the loss $\mathcal{L}_{\text{HST}}(U(\boldsymbol{\theta}_{t_m}), V_{t_m})$ for exact verification of correct unitary compilation. In general, my scheme can achieve compilation at larger system sizes (e.g., $n = 50$ qubits in Fig. 5.9), although verifying faithfulness would require quantum resources (one can typically detect manifold jumping on a quantum device using only implementations of the ansatz $U(\boldsymbol{\theta})$, rather than the full target $V_{t_m}$, so this is viable).

I compile the time-evolution operators for the case of no magnetic field ($\xi = 0$) and of a disordered magnetic field ($\xi = 4$) with a Hamiltonian renormalized as $H_{\text{TFXY}} \to \frac{H_{\text{TFXY}}}{\|H_{\text{TFXY}}\|_{\text{HS}_1}}$ to eliminate any norm dependence of the dynamics. The ansatz is determined by Equation 5.61 and is composed of $L = 17$ layers, which is overparametrized to ensure trainability. In line with the strategy previously discussed, the circuit is first initialized with $\boldsymbol{\theta}_{t_0} = \mathbf{0}$, and thereafter initialized with $\boldsymbol{\theta}_{t_m}$ according to the parameters found in the $t_{m-1}$ step of optimization. In Figure 5.12, I report a comparison between the compiled dynamics and a first-order Trotterization at the same depth (i.e. identical circuit structure with $\theta_{lk} = t/L$). I find that, despite using identical circuit structures, my scheme outperforms Trotterization by several orders of magnitude. Looking at the dynamics of the position operator $\sqrt{\hat{N}^2}$ (Figure 5.12(a)), one can see that Trotterization fails to reproduce results beyond $t \approx 60$, while my scheme continues to track them accurately at all times $t_m$ considered, with errors smaller than $10^{-6}$ (Figure 5.12(b)). More generally, I find that the compiled unitaries accurately reflect the true time-evolution with $\mathcal{L}_{\text{HST}}(U(\boldsymbol{\theta}_t), V_t) < 10^{-5}$ (Figure 5.12(c)), thus guaranteeing that the compiled circuits can faithfully reproduce the dynamics of *all* observables, not just those

supported by $\mathfrak{g}$ and products thereof. Furthermore, the errors entailed by the compiled circuit do not appear to vary substantially in the duration of the dynamics (fixed-depth simulation is known to be possible for $\dim(\mathfrak{g}) \in \text{poly}(n)$ models [382]). Overall, this shows that for time-evolution operators whose associated Lie algebra $\mathfrak{g}$ has $\dim(\mathfrak{g}) \in \mathcal{O}(\text{poly}(n))$, my $\mathfrak{g}$-sim compilation scheme can determine a much more efficient circuit implementation than standard Trotterization.

### 5.4.4 Supervised quantum machine learning

As a final demonstration, I employ $\mathfrak{g}$-sim for the training of a quantum-phase classifier, showcasing its applicability in the context of QML. Despite being trained fully classically on tractable states, such a classifier could then be employed to classify unknown quantum states on a quantum device. Provided that useful classifiers can be constructed in $\mathcal{G}$ with $\dim(\mathfrak{g}) \in \text{poly}(n)$ and classical representations are sampled for nontrivial states, such schemes could find utility in real experiments. Additionally, in the spirit of Section 5.4.2, the $\mathfrak{g}$-sim framework could form the basis of approximate quantum models that are then refined on a quantum computer.

#### 5.4.4.1 General framework

In general problems of supervised QML, one assumes repeated access to a training dataset $\mathcal{S} = \{(\rho_s, y_s)\}_{s=1}^N$ consisting of $N$ pairs of states $\rho_s$ together with labels $y_s = F(\rho_s)$ that have been assigned by an unknown underlying function $F$. The task is to learn parameters $\boldsymbol{\theta}$ of a function $h_{\boldsymbol{\theta}}$, with the aim of approximating $F$ as accurately as possible. Upon successful training, it should then be possible to accurately infer the labels of previously unseen states.

As is typical in QML tasks, in this section I consider a model $h_{\boldsymbol{\theta}}$ that relies on the expectation value of an observable $O$ after application of a circuit $U(\boldsymbol{\theta})$. That is, the cost function will be of the form $\ell_{\boldsymbol{\theta}}(\rho_s) = \text{Tr}\Big[OU(\boldsymbol{\theta})\rho_s U^\dagger(\boldsymbol{\theta})\Big]$. Training then relies upon minimization of a mean-squared error loss function for classification, defined here as

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{N}\sum_{s=1}^{N}(y_s - \ell_{\boldsymbol{\theta}}(\rho_s))^2. \tag{5.63}$$

#### 5.4.4.2 Training a binary quantum-phase classifier

For this numerical study, I apply the $\mathfrak{g}$-sim framework to the classification of ground states across a phase transition of the TFIM (5.49), at system size $n = 50$ qubits (well beyond the reach of exact state-vector methods). I consider parameters $h_z, h_{xx} \in [0, 1]$ in order to focus on a single phase transition (binary classification). For $h_z/h_{xx} > 1$, the ground state is in the disordered phase, while for $h_z/h_{xx} < 1$ it is in the antiferromagnetic phase. Furthermore, to ensure that the problem is non-trivial, I 'disguise' the TFIM according to a random $V \in \mathcal{G}$ (Equation 5.58 with $T = 10$) such that

$$H_{\text{disguised}} = V H_{\text{TFIM}} V^\dagger. \tag{5.64}$$

To generate the training and test datasets, I start by randomly sampling values of $h_z$ and $h_{xx}$ and assign labels $y_s = -1$ $(+1)$ to Hamiltonian parameters corresponding to the disordered (antiferromagnetic) phase. Each set of parameters corresponds to a distinct instance of $H_{\text{TFIM}}$ and, for each of them, I take the corresponding state $\rho_s$ to be the (approximate) ground state of this Hamiltonian instance. A $\mathfrak{g}$-sim classical representation of $\rho_s$ consists of the vector of expectation values $(\boldsymbol{e}^{(s)})_\alpha = \text{Tr}[\rho_s G_\alpha]$. Since exact diagonalization is intractable for $n = 50$ qubits, to compute these, I use VQE in $\mathfrak{g}$-sim with an overparametrized ansatz to obtain a circuit $U(\boldsymbol{\theta})$ such that $\rho_s \approx U(\boldsymbol{\theta}) |0\rangle \langle 0|^{\otimes n} U^\dagger(\boldsymbol{\theta})$. Due to overparametrization, this is guaranteed to produce a good approximation of the ground state, with convergence as $L \to \infty$: to ensure good ground states, I train states on a fixed Hamiltonian with varying $L$, selecting an $L$ large enough to ensure convergence. I then compute the classical representation $\boldsymbol{e}^{(s)}$ by applying Equation 5.12 to a classical representation of the computational zero state (Equation 5.46). Finally, I transform this classical representation of the ground state of $H_{\text{TFIM}}$ to the corresponding 'disguised' ground state of $H_{\text{disguised}}$ by applying $V$ to it. This procedure is repeated to generate a training dataset of 200 labeled states, equally divided in the two phases. To evaluate the performances of the classifier, I apply the same procedure to create a separate test dataset with the

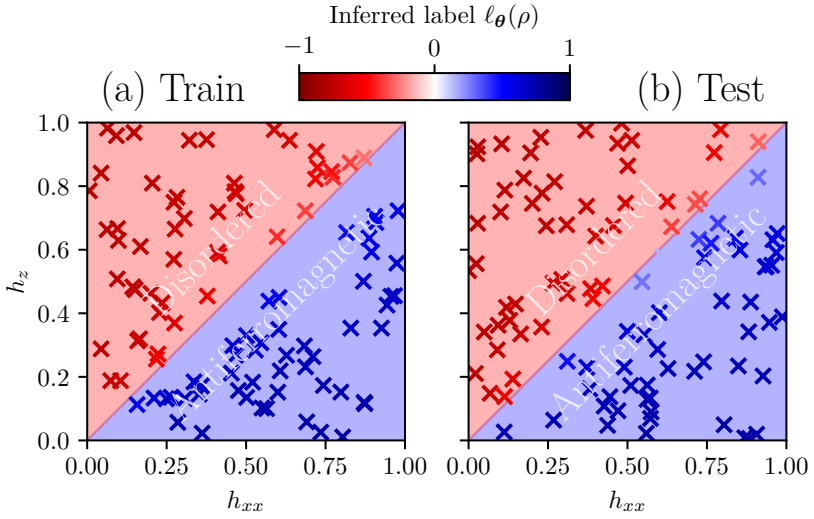



**Figure 5.13: Supervised QML with a classically-efficient circuit. (a)** Label assignments from a 50-qubit classifier on a training dataset of disguised TFIM ground states for $T = 10$ (details in the main text), across the disordered-antiferromagnetic phase transition. The classifier is trained to minimize the loss function in Equation 5.63. **(b)** Inference on a new set of data drawn from the same distribution. The model achieves 100% classification accuracy on the test dataset.

same number of states; these have not been seen by the optimizer during training and therefore represent a genuine test of its in-distribution generalization.

The classifier is realized by $L = 21$ layers of the 2-local ansatz in Equation 5.61, and a measurement observable $O = \sigma_1^z \in i\mathfrak{g}_0$. I train the classifier classically using $\mathfrak{g}$-sim to minimize the loss function in Equation 5.63 over the training dataset. Once the parameters are trained, I assess accuracy of the classifier on the test dataset. Given trained parameters $\boldsymbol{\theta}^*$, to perform inference I assign inferred labels as

$$h_{\boldsymbol{\theta}^*}(\rho_s) = \mathrm{sgn}(\ell_{\boldsymbol{\theta}^*}(\rho_s)) \in \mathcal{Y}. \tag{5.65}$$

I depict the results of this numerical experiment in Figure 5.13(a), showing that for this problem the classifier achieves 100% classification accuracy on the new data. Consistent classification accuracy is also verified across multiple random instances of the unitary disguise $V$, thus demonstrating successful application of $\mathfrak{g}$-sim to a supervised QML problem.

Once the classifier is trained, obtaining a set of optimal parameters $\boldsymbol{\theta}^*$, one should question how it can most efficiently be used for inference. For a standard

quantum-circuit-based classifier, the only efficient option would be to apply the unitary $U_* \equiv U(\boldsymbol{\theta}^*)$ to the unknown state on a quantum computer, measure the operator $O = \sigma_1^z$, and assign a label according to Equation 5.65. For a classifier based on $\mathfrak{g}$-sim, when $\dim(\mathfrak{g}) \in \text{poly}(n)$, the structure of the algebra and limited coupling between observables allows for a different approach. Although Result 2 was framed as an evolution of expectation values, it can equivalently be seen as an (Heisenberg) evolution of the observable operator itself. Given the adjoint representation $\bar{\boldsymbol{U}}_* = \Phi_{\mathcal{G}}^{\text{Ad}}(U_*)$ of the trained circuit (computed in $\mathfrak{g}$-sim), and a description of the observable $O \in i\mathfrak{g}$ as the vector $\boldsymbol{w}$ such that $O = \sum(\boldsymbol{w})_\alpha G_\alpha$, one can write

$$O' := U_*^\dagger O U_* = \sum_{\alpha=1}^{\dim(\mathfrak{g})} (\boldsymbol{w}')_\alpha G_\alpha, \text{ with } \boldsymbol{w}' = \boldsymbol{w}^{\mathrm{T}} \bar{\boldsymbol{U}}_*. \tag{5.66}$$

Hence, one recovers the classifier outputs as a weighted sum of the expectation values of the observables $G_\alpha$, which is only $\text{poly}(n)$ in size. Therefore, one can efficiently determine the classifier output by measuring the expectation values of a polynomial number of observables, without ever needing to apply the full unitary $U_*$. This represents a substantial saving in circuit depth, albeit at increased (still polynomial) measurement cost. Depending on the characteristics of the quantum device at hand (i.e. measurement versus gate fidelity and cycle times), one approach may be preferred to another. In the conclusions to this chapter (Section 5.5), I discuss the implications of the circuit-based approach; here, I expand upon the measurement-based approach.

For the measurement-based approach, one may reduce the amount of measurements required further by pruning the model (i.e. approximating Equation (5.66) with a much smaller subset of the terms involved). For instance, this could be achieved by discarding terms corresponding to small weights $|(\boldsymbol{w}')_\alpha| < \delta$ for some threshold $\delta$, or by discarding greater-than-$k$-local terms. In particular, the latter could be combined with classical shadows (Section 2.1) by means of random measurements (rather than direct measurements of expectation values of all the $G_\alpha$), ensuring small approximation error by bounded locality.

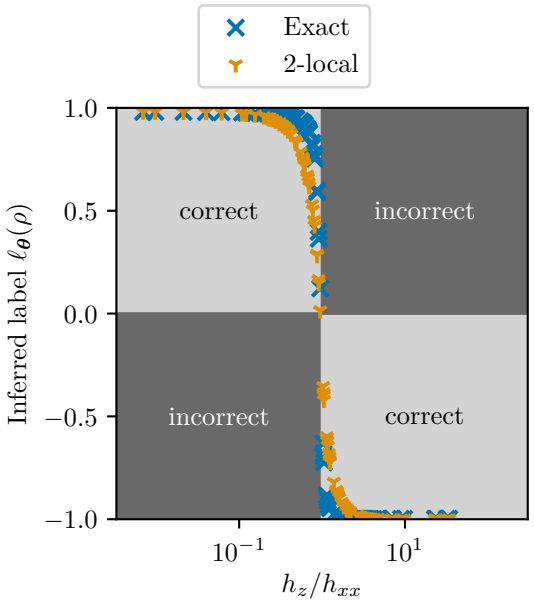



**Figure 5.14: Comparison of exact label inferences to their 2-local approximation.** I compare the exact inferred labels of the TFIM phase classifier on an unseen training dataset to a 2-local approximation. The 2-local approximation still achieves 100% classification accuracy, although $\ell_{\boldsymbol{\theta}}(\rho)$ changes less rapidly near the phase transition, indicating increased perplexity of the classifier at the phase boundary.

To further study the effect of such locality pruning, I compare the accuracy of a trained classifier to its 2-local approximation, for an undisguised ($T = 0$) TFIM of $n = 50$ qubits. As can be seen in Figure 5.14, despite slight differences in the output of the models, the approximated classifier retains full accuracy (with 100% of the testing data correctly classified) that can be explained by the high locality of the TFIM order parameters. It is likely that this approach can be refined further by carefully adapting the choice of classical shadow procedure to the data or the classifier architecture. For example, due to the free-fermionic structure of this classifier, further improved results may be obtained using matchgate- or free-fermion-based variants of classical shadows [329–331].

## 5.5 Conclusions

Efficient classical simulations of quantum circuits are a critical tool in scaling towards quantum advantage. In this chapter, I have introduced $\mathfrak{g}$-sim, a classical simulation

and optimization framework that relies on the Lie-algebraic structure of the circuits. Basing my approach on older, relatively under-utilized tools for Lie-algebraic simulation of quantum dynamics [112, 113], I adapted the presentation of these techniques to the needs of the modern quantum computing community. I further generalized these techniques to relax the conditions for efficient simulation, improved upon the asymptotic complexity of circuit-based simulations, provided guidelines for efficient computer implementation, and modified the framework to accommodate general input states. In doing the latter, I performed a simulation task which is not possible with any other known technique. By introducing circuit optimization to this framework, I enabled the scalable classical study of paradigmatic examples relevant for variational quantum computing, demonstrating utility in studying scaling behaviors of VQA problems, improving trainability and mitigating barren plateaus via classical pre-training, and implementing supervised QML problems. Furthermore, I expanded the framework of $\mathfrak{g}$-sim to include compilation and circuit synthesis, which is a novel direction for simulation schemes of this type. By constructing and optimizing a circuit fidelity cost function that can be computed entirely in $\mathfrak{g}$-sim, I demonstrated that one can synthesize linear-depth circuits for unitary transformations in $e^{\mathfrak{g}_0}$. Compiling such rotations has already proven to be of great utility to the community, with e.g. the use of Givens rotation decomposition to prepare Hartree-Fock states with linear circuit depth [348, 349]. However, $\mathfrak{g}$-sim does not necessarily require an underlying free-fermion structure, and therefore expands the class of compilable transformations to include *any* system corresponding to a poly($n$)-dimensional algebra, such as systems with permutation symmetries [373].

Prior to this work, variational circuits with $\dim(\mathfrak{g}) \in \text{poly}(n)$ seemed highly promising: being provably barren-plateau-free [213] and efficiently overparametrizable [225], they are trainable from the perspective of both acquiring gradients and navigating the optimization landscape. The ability to classically simulate these circuits is, admittedly, disappointing for variational quantum computing. Nonetheless, this chapter represents an important pillar of the ongoing effort to understand the connections between trainability and simulability in variational

quantum computing [134]. By understanding the structures that *prevent* a quantum advantage, the community's efforts in variational quantum computing can be better focused.

### 5.5.1 Scope for future work

To date, Lie-algebraic considerations have been pivotal in many topics of quantum science including quantum error correction [392, 393], controllability of quantum systems [369, 370, 394–396], efficient measurements [397], dynamical simulations [377, 398], and studying trainability properties of parametrized quantum circuits [213, 225, 227, 228]. This chapter has demonstrated further utility in existing applications (dynamical simulations and studying trainability properties), and expanded this list to include classical pre-training of VQAs, efficient circuit compilation, and supervised QML. I anticipate that the $\mathfrak{g}$-sim framework will be a useful tool with which others in the community can further expand this list, a process which has already begun [226].

One promising prospective application is quantum error mitigation (see Section 2.3 for a brief overview). In particular, learning-based QEM methods [119–122] require access to pairs of noisy circuit outputs (obtained from quantum hardware) and corresponding noiseless outputs (obtained from classically-efficient simulation techniques) in order to learn a function mapping noisy outputs to their correct noiseless values. Since $\mathfrak{g}$-sim can lead to classically-efficient simulation of circuits that are distinct from these existing methods, it can expand upon the range of noiseless training data available, and likely further facilitate these methods.

In a similar vein, $\mathfrak{g}$-sim has potential applications in the context of randomized benchmarking [117]. The greatest limitation in quantum computing is the effect of hardware errors in computations, both coherent and incoherent. There is thus a pressing need to comprehensively characterize the type and magnitude of errors present on quantum hardware. One approach to this is randomized benchmarking, which compares outputs of random gate sequences of increasing length to known expected results without resorting to standard process tomography. Recent work [118] has investigated the use of matchgate circuits for randomized

benchmarking. Matchgates are closely related to $\mathfrak{g}_0$, and the underlying simulation schemes have much in common with $\mathfrak{g}$-sim. Much like the case of error mitigation, $\mathfrak{g}$-sim could be used to generate benchmarking data for non-matchgate circuits, expanding the scope of these techniques.

I also note the opportunity to further improve the compilation scheme in Section 5.4.3. Although its performance appears competitive with the Cartan decomposition approach of Ref. [382], one disadvantage is that my approach requires re-optimization at each time step, whereas the former requires only one optimization. I believe that this limitation could be overcome by using a variational fast-forwarding ansatz [188], which should work since the diagonalization unitary must necessarily be in $\mathcal{G}$. Furthermore, my scheme could trivially be extended to the compilation of evolution unitaries for time-dependent Hamiltonians. In the case of Hamiltonians with periodic Floquet driving, it should even be possible to construct a fast-forwardable solution, which to the best of my knowledge would not be possible with Cartan decomposition methods.

Lastly — but perhaps most promisingly — one should carefully consider the role of $\mathfrak{g}$-sim in the future of supervised QML. Even if the trainability-simulability connections proves so fundamental as to make variational QML pointless on quantum computers, there is still scope for usefulness. I have already noted that classical machine learning algorithms equipped with data from quantum computers can outperform their purely classical counterparts [129–133]. It seems reasonable to suspect that such classical machine learning algorithms would benefit from inductive biases that take into account quantum-mechanical structure, and building dequantized QML algorithms from $\mathfrak{g}$-sim would be a natural way to incorporate this. Furthermore, since $\mathfrak{g}$-sim directly simulates the action of a quantum circuit, one can decide based on needs and context whether to execute the circuit classically or quantumly. Given the substantial complexities of actually implementing a hybrid quantum-classical loop (and the effect of noise inhibiting trainability [218, 399]), training on a classical device is highly desirable. However, once trained classically on a quantum device, the model can be deployed on a quantum computer for inference

(essentially the opposite of the scheme considered at the end of Section 5.4.4). Although performing inference in $\mathfrak{g}$-sim requires only $\mathcal{O}(\text{poly}(n))$ shots (since one needs to compute a classical representation), performing it on a quantum device would likely require only $\mathcal{O}(1)$ shots. Therefore, quantum training with classical deployment could prove to drastically reduce the measurement overheads required in these tasks, and could provide practical advantages in *inference* even in cases where there is provably no quantum advantage in *learning*.

A particularly important context where measurement overheads must be carefully accounted for is quantum sensing. In the following chapter, I study the application of quantum error mitigation and learning-based methods to quantum sensing, carefully accounting for the effects of measurement overheads when considering their practicality on a finite shot budget.

*I know what you're thinking. 'Did he fire six shots or only five?' Well, to tell you the truth, in all this excitement, I've kinda lost track myself.*

— Harry Callahan, *Dirty Harry* [400]

# 6

# More buck-per-shot: Why learning trumps mitigation in noisy quantum sensing

*The research in this chapter was conducted jointly with Aroosa Ijaz, Cinthia Huerta Alderete, Frédéric Sauvage, Lukasz Cincio, and Marco Cerezo, and led to the published paper of Ref [401], which this chapter closely follows. Following a core idea conceived by Marco, I led the project as senior author, significantly expanding the framework and methodology. Aroosa, Marco, Cinthia and I worked together on the anaytic derivations of error bounds, and the numerical simulations and bounds comparisons were solely my work. In places where text from the published manuscript is used verbatim, I am the original author of this text.*

## Contents


**6.1 Framework . . . . . 182**
6.1.1 Noisy-state quantum sensing . . . . . 182
6.1.2 An illustrative toy model . . . . . 186
6.1.3 Zero-noise extrapolation . . . . . 187
6.1.4 Hyperparameter optimization . . . . . 189
6.1.5 Learning-based inference methods . . . . . 191
6.1.6 Parameter measurement via phase measurement . . . . 195
**6.2 Analytic estimates of error bounds . . . . . 197**
6.2.1 Noise-aware noisy quantum sensing . . . . . 200
6.2.2 Naive noisy quantum sensing . . . . . 201
6.2.3 Noisy sensing mitigated by zero-noise extrapolation . . 202
6.2.4 Inference-based noisy sensing . . . . . 203

6.2.5 Inference-based noisy sensing mitigated by zero-noise extrapolation . . . . . 208
6.2.6 Comparison of error bounds . . . . . 209
**6.3 Numerical experiments . . . . . 213**
6.3.1 Noise models . . . . . 214
6.3.2 Numerical comparison of errors across varying regimes . 215
6.3.3 Requirements for pre-characterized inference-based sensing 218
**6.4 Conclusions . . . . . 220**

---

Thus far in this thesis, I have explored a range of topics relating to the prospects of using entangled quantum states to perform computations more efficiently than any classical device. In Section 1.3, I also outlined how entangled quantum states can in principle be used to perform high-precision measurements better than any classical device. In the absence of entanglement, parameter measurement is limited by the SQL (Equation 1.3), but allowing for entanglement leads to the better bound of the HL (Equation 1.4). Much like quantum computing, quantum sensing is plagued by the deleterious effects of noise, which hinders a probe state's capability as a sensor [99–102]. Indeed, the similarities to quantum computing might be the source of solutions as well, since much of the wealth of techniques developed for quantum computing can be repurposed for quantum sensing. In Section 2.2.2, I mentioned how VQA techniques have been proposed to prepare optimized probe states and measurement protocols [170–181], and in Section 2.3 I discussed how error mitigation techniques have been considered for quantum sensing [281–285]. This chapter is concerned with the latter: while there is some promise for active error mitigation techniques in quantum sensing [281–284], there has been almost no work on passive error mitigation for sensing, and existing work on the matter [285] considers mostly single-qubit Ramsey interferometry (i.e. unentangled and thus not 'quantum sensing' by the definition of Section 1.3) and does not consider asymptotic scalings relative to the SQL and HL.

This leaves a gap in the literature that can be addressed by a careful comparative study of different sensing protocol errors, relative to the all-important SQL and HL. In this chapter, I study whether investing shots in error mitigation, learning-based inference techniques [286], or combinations thereof, can genuinely improve the

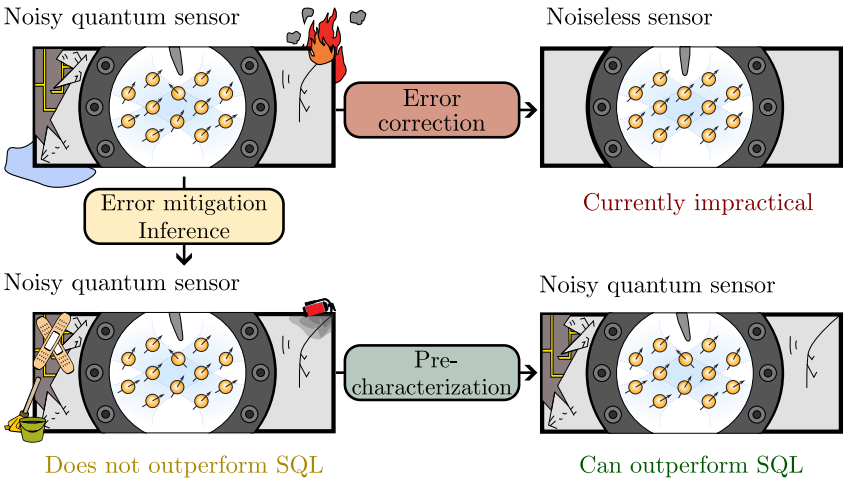



**Figure 6.1: Summary of this chapter's results.** This chapter considers a noisy quantum sensing task, under the assumption that error correction is not available. I study whether the use of error-mitigation techniques (specifically ZNE) and inference tools can be used to improve the sensor's performance. This chapter's results show that in order to beat the SQL, pre-characterization of the system's response is a necessary condition.

sensitivity of a noisy quantum sensor on a constrained shot budget. The latter is a crucial point: in quantum sensing, as in quantum computing, shots are a precious resource, as one can see from the presence of the shot budget $N$ in the denominator of the quantum Cramér-Rao bound (1.1), and infinite shots can lead to infinite precision. Since error mitigation and learning-based inference techniques both incur a shot overhead, one must take great care to ensure that any shot not being used for sensing is actually leading to some metrological improvement. This chapter critically assesses this via a detailed bias-variance error analysis — in the case of ZNE, I show that the inherent bias-variance tradeoff (see Section 2.3) leads to worse performance, regardless of existing knowledge about the system. I ultimately conclude that pre-characterizing a quantum sensor via inference techniques leads to the best performance — and in fact, within the regimes of analytic and numerical study presented in this chapter, is the only strategy capable of gaining a quantum advantage. Since pre-characterization requires a stable response function between the pre-characterization and sensing phases, this chapter unveils that stability is a crucial (but often overlooked) criterion for a useful quantum sensor. This conclusion is summarized in Figure 6.1.

The chapter is structured as follows. In Section 6.1, I outline our framework and assumptions for an archetypal noisy quantum sensing task. In Section 6.2, I present our analytic derivations of error bounds for various sensing protocols, and directly compare them in an indicative example. Moving beyond idealized noise models, in Section 6.3 I simulate these protocols explicitly under realistic noise models learned from a real quantum device, supporting our analytic study with numerical evidence. I then draw conclusions from our study in Section 6.4, yielding clear insights for the comparative value of these protocols.

## 6.1 Framework

In this section, I outline the definitions and tools that underpin this chapter. I begin by introducing a rather general framework for quantum sensing with noise-afflicted operations, and then demonstrate explicitly with a simple toy model how noise can compromise a quantum sensor. Following this, I review basics of ZNE and learning-based inference techniques for sensing [286], the tools which will be comparatively studied throughout this chapter.

### 6.1.1 Noisy-state quantum sensing

Throughout this chapter, I consider a setting wherein one attempts to use a quantum sensor afflicted by noise to learn a single unknown parameter $\alpha$ encoded into a probe state by some external field. To initialize the system, one prepares a probe state $\rho$ by applying a noisy state preparation channel $\mathcal{E}_\lambda$ (where $\lambda$ parametrizes the overall fault rate such that $\lambda = 0$ indicates a noiseless operation) to a fiducial $n$-qubit input state $\rho_{\text{in}}$ (taken throughout this chapter to be the computational zero state). In general, the action of $\mathcal{E}_\lambda$ will be extremely difficult to determine, since it can induce complicated operations on an exponentially-sized Hilbert space. Realistic noise channels are extremely difficult to model, and even in the noiseless case $\lambda > 0$, the action of non-trivial unitary probe state preparation channels may not be analytically tractable (e.g. heuristically/variationally trained probe states [170–181]). Since the mapping between measurements and the encoded parameter (i.e. the

response function, see Equation 6.3) will typically not be known analytically, one must choose between a variety of strategies to compensate. These include simplifying assumptions (our so-called 'naive' protocol), error mitigation techniques, or learning-based inference techniques, all of which I will evaluate and compare in this chapter.

Once prepared via the channel $\mathcal{E}_\lambda$, the probe state is then allowed to interact with an external field that encodes an unknown parameter $\alpha$ in the probe state through the action of a Hamiltonian

$$H = \alpha H_{\text{field}}. \tag{6.1}$$

Consequently, the parameter encoding channel is

$$\mathcal{S}_\theta(\rho) = e^{-i\theta H}\rho e^{i\theta H}, \tag{6.2}$$

where $\theta$ is a phase imprinted on the state by the interaction. Here, I note that most theoretical treatments of quantum metrology conflate $\theta$ and $\alpha$, but this is not sufficient for this chapter. For now, it suffices to say that by measuring the phase $\theta$, one can usually obtain information about the parameter $\alpha$. For instance, in the simplest case, one has $\theta = \alpha T$ where $T$ is the interrogation time. In Section 6.1.6 I discuss how additional effects can lead to more intricate relations between the parameter $\alpha$ and its imprinted phase $\theta$, and how this affects our results.

Finally, with the parameter encoded, one sends the state through a noisy pre-measurement channel $\mathcal{D}_\lambda$, after which one computes the expectation value of some suitable observable $O$ from which the parameter $\alpha$ might be inferred. I note that this framework is more general than it may appear: since $\mathcal{S}_\theta$ is invertible as written, any noise that acts during the interrogation step can be incorporated into $\mathcal{D}_\lambda$ (as can measurement-related errors, or any other noise which acts during or after interrogation). Also, much like the state preparation noise, analytically obtaining $\mathcal{D}_\lambda$ for any $\lambda \neq 0$ can be difficult (typically requiring expensive full-process tomography). Determining an optimal measurement $O$ is therefore usually done heuristically.

Combining all of these ingredients, one obtains the system's (noise-dependent) response function $R_\lambda(\theta)$, given by

$$R_\lambda(\theta) = \mathrm{Tr}[\mathcal{D}_\lambda \circ \mathcal{S}_\theta \circ \mathcal{E}_\lambda(\rho_{in})O]. \tag{6.3}$$

Throughout this chapter, I denote the unknown phase of interest that one aims to estimate (or 'target phase') as $\theta^*$. Since in all practical settings the response to the external field is estimated via $N$ shots (i.e. $N$ repetitions per the quantum Crámer-Rao bound (1.1)), I will use $\overline{R}$ to indicate an $N$-shot-limited response. Once one obtains a measurement of this $N$-shot-limited response $\overline{R}_\lambda(\theta^*)$, one must invert it via the response function to obtain an estimate $\hat{\theta}^*$, given by

$$\hat{\theta}^* \equiv R_\lambda^{-1}(\overline{R}_\lambda(\theta^*)). \tag{6.4}$$

Before proceeding, it is important to remark on four underlying assumptions required to achieve useful sensing with this general framework:

1. One must have a precise knowledge of the system-environment interaction, i.e. the parameter encoding mechanism $\mathcal{S}$.

2. One must be able to identify (and prepare, via $\mathcal{E}_\lambda$) a suitable probe state, with a level of metrological usefulness defined by its QFI $F_Q(\rho, H)$.

3. Given this probe state, one must determine a suitable measurement procedure.

4. Having selected a probe state and measurement procedure, one must characterize the response function $R_\lambda(\theta)$, such that the inversion in Equation 6.4 can be performed.

For many metrological tasks, such as magnetometry, the first assumption is quite reasonable. There is some subtlety to the second and third assumptions. Although in the absence of hardware noise, one can find analytic schemes where the HL is saturated [402–404], determining optimal probe states is far more challenging in the presence of noise, and highly dependent on the structure of the noise itself. Furthermore, while a large QFI $F_Q(\rho, H)$ tells us that there *exists* a measurement

that is metrologically useful, it tells us nothing about what that measurement actually is. In practice, satisying the second and third assumptions might be done by using constructions known to work well in the noiseless case, applying a system-dependent heuristic, or using variational techniques to optimize the probe state and measurement [170–181]. In this chapter, I rely upon the former method; determining metrologically optimal measurement and probe state protocols is beyond the scope of this investigation.

The fourth and final assumption is perhaps the most overlooked in standard treatments of quantum metrology, and this chapter mostly deals with the subtleties of satisfying it. To determine the inverse resonse $R_\lambda^{-1}$, one must be able to evaluate Equation 6.3 for a range of $\theta$ values. Although one may be able to determine the functional form of $R_0^{-1}$ analytically for some probe states and measurements, when one considers general noise models, state preparation channels and measurements, this is a computationally hard task that cannot in general be expected to be classically tractable. One approach considered in this chapter is to use model-agnostic, inference-based learning techniques [286] to learn the response function efficiently from a set of noisy measurements. Although outlined in Ref. [286] for fixed error bounds, their all-important asymptotic scaling versus the SQL and HL had not been studied prior to this work. Another approach is to use error mitigation techniques, such as ZNE, to attempt to drive the measurement $\overline{R}_\lambda(\theta^*)$ closer to the noiseless response $R_0(\theta^*)$ (assuming that one actually knows the noiseless response). Both approaches are summarized in the latter part of this section. An additional concern is the fact that the response function $R_\lambda(\theta)$ will not in general be an invertible function, and must be constrained to a fixed domain — this point is addressed in Section 6.1.6.

With a sensing protocol and response function defined, one should then be concerned with the sensitivity of parameter estimation — that is, the preicision to which the unknown phase $\theta^*$ can be estimated. A local linearization leads to the error propagation formula [405]

$$\Delta^2\hat{\theta}^* = \frac{\Delta^2\overline{R}_\lambda(\theta^*)}{(\partial_\theta R_\lambda(\theta)|_{\theta=\theta^*})^2} = \frac{\Delta^2 R_\lambda(\theta^*)}{N(\partial_\theta R_\lambda(\theta)|_{\theta=\theta^*})^2}, \tag{6.5}$$

where $\Delta^2\hat{\theta}^*$ denotes the variance of the estimated phase, the variance of the response function depends on the number of shots $N$ as $\Delta^2\overline{R}_\lambda(\theta^*) = \Delta^2 R_\lambda(\theta^*)/N$, and the inherent quantum variance of the response is $\Delta^2 R_\lambda(\theta^*) = \mathrm{Tr}[\mathcal{D}_\lambda \circ \mathcal{S}_{\theta^*} \circ \mathcal{E}_\lambda(\rho_{in})O^2] - \mathrm{Tr}[\mathcal{D}_\lambda \circ \mathcal{S}_{\theta^*} \circ \mathcal{E}_\lambda(\rho_{in})O]^2$. Throughout this chapter, for notational compactness I denote partial derivatives with the shorthand $\partial_\theta(\cdot) \equiv \partial(\cdot)/\partial\theta$.

### 6.1.2 An illustrative toy model

This rather general framework becomes clearer upon considering a simple toy model, which explicitly illustrates the effects of noise on a quantum sensor. Consider a magnetometry task, wherein one aims to use an $n$-qubit sensor to estimate the value $\alpha$ of an unknown magnetic field, encoded via a channel $\mathcal{S}_\theta(\rho)$ (6.2), per the Hamiltonian

$$H = \frac{\alpha}{2}\sum_{j=1}^{n}\sigma_j^z. \tag{6.6}$$

Assuming that the dominant source of noise is in entangled probe state preparation, one may take $\mathcal{D}_\lambda$ to be the identity map. Targeting a Greenberger-Horne-Zeilinger (GHZ) probe state such that in the noiseless case $\mathcal{E}_0(\rho_{\mathrm{in}}) = |\mathrm{GHZ}\rangle\langle\mathrm{GHZ}|$ with $|\mathrm{GHZ}\rangle = \frac{1}{\sqrt{2}}(|0\rangle^{\otimes n} + |1\rangle^{\otimes n})$, then in the absence of noise a metrologically optimal measurement is the parity operator $O = X^{\otimes n}$. To see this, one can substitute these components into Equation 6.3, from which one obtains the response function $R(\theta) = \cos(n\theta)$; substituting this into Equation 6.5 yields $\Delta^2\theta = \frac{1}{Nn^2}$, which is the HL.

Including noise in this model compromises the sensitivity of the protocol. For the simplest possible noise model, let $\mathcal{E}_\lambda$ consist of a unitary GHZ state preparation channel composed with a global depolarizing channel acting on all qubits with probability $p_\lambda$ (parametrized as $p_\lambda = 1 - e^{-\lambda}$ such that $\lambda$ is a fault rate). Clearly, the parameter-encoded state will then by given by

$$\mathcal{S}_\theta \circ \mathcal{E}_\lambda(\rho_{in}) = (1 - p_\lambda)\mathcal{S}_\theta(\rho_{\mathrm{in}}) + \frac{p_\lambda}{2^n}\mathbb{1}. \tag{6.7}$$

Since parity measurement vanishes in the maximally mixed state, it follows that $\Delta^2 R_\lambda(\theta) = 1 - (1 - p_\lambda)^2\cos^2(n\theta)$ and $|\partial_\theta R_\lambda(\theta)| = (1 - p_\lambda)n|\sin(n\theta)|$. Combining

these, overall the sensitivity is given by

$$\Delta^2\theta = \frac{1}{Nn^2} + \frac{2p_\lambda - p_\lambda^2}{Nn^2(1-p_\lambda)^2\sin^2(n\theta^*)}, \tag{6.8}$$

where the first term corresponds to the noiseless limit of the HL, but the second, noise-induced term is strictly positive for any $p_\lambda > 0$, indicating that this toy-model sensor will always be worse than the HL in the presence of noise. Indeed, this noise-induced term can even be large enough such that the overall precision is worse than the SQL, negating all advantage from quantum entanglement (and even causing the system to fail to meet my definition of a 'quantum sensor' in Section 1.3). Here, the two-fold detrimental effect of noise is clear: not only does it spread the state more across the eigenvalues of $O$, increasing the inherent quantum variance $\Delta^2 R_\lambda(\theta^*)$ on the numerator, but it also flattens the response function, reducing the derivative on the denominator. Both factors contribute to the overall loss of precision, and will prove important in the analysis of Section 6.2. Although these results assumed a global depolarizing noise model, they generalize straightforwardly to a range of noise models which have the maximally mixed state as their fixed point [406, 407]. With the detrimental effects of noise in quantum sensing now illustrated, I elaborate upon the methods that may be considered to counteract them.

### 6.1.3 Zero-noise extrapolation

#### 6.1.3.1 The method

As outlined in Section 2.3, zero-noise extrapolation (ZNE) is among the earliest quantum error mitigation techniques, and relies upon the extrapolation of an observable at various boosted noise levels, such that a fitting function can be used to extrapolate to the zero-noise limit. For reasons explained there, I focus on Richardson extrapolation, where polynomial interpolation is used to perform the extrapolation. I now review the basics of Richardson extrapolation in the context and notation of this chapter. Given a base noise level $\lambda$ and a desired observable — in this case, the noiseless response function $R(\theta^*)$ evaluated at the target phase $\theta^*$ — the experimenter evaluates the response at $m+1$ distinct

noise levels (ignoring for now the limited shot cost, which will be discussed in the following section). These noise levels, or 'nodes', are parametrized as $\lambda_j = x_j\lambda$, where $x_j \in [1, \infty)$ are the noise amplification factors, with $x_0 = 1$ yielding the base noise level $\lambda_0 = \lambda$. Given this set of points $\{\lambda_j, R_{\lambda_j}(\theta^*)\}_{j=0}^m$, there exists a unique polynomial interpolant of degree $m$ that passes through all of these points [408]. Writing this polynomial in the Lagrange basis

$$\ell_j(\lambda) = \prod_{\substack{0\leq k\leq m \\ k\neq j}} \frac{\lambda - \lambda_k}{\lambda_j - \lambda_k}, \tag{6.9}$$

one can write the interpolant in noise level $\lambda$ as

$$R_\lambda(\theta^*) \approx \sum_{j=0}^{m} R_{\lambda_j}(\theta^*)\ell_j(\lambda). \tag{6.10}$$

Here, one is mostly interested in the zero-noise limit $\lambda \to 0$. Defining the zero-noise extrapolation coefficients $\gamma_j \equiv \ell_j(\lambda = 0)$, the error-mitigated response $R_M(\theta^*)$ is given by

$$R_M(\theta^*) = \sum_{j=0}^{m} \gamma_j R_{\lambda_j}(\theta^*). \tag{6.11}$$

Now, it is important to note that the inexactness of Equation 6.10 is due to the use of a finite $m$-th degree polynomial — assuming a well-behaved function in $\lambda$, one could write the response exactly in terms of its Taylor series about $\lambda = 0$. Thus, the bias induced by this truncation is of order $\mathcal{O}(\lambda^{m+1})$, and

$$R_M(\theta^*) = R(\theta^*) + \mathcal{O}(\lambda^{m+1}). \tag{6.12}$$

For sufficiently small $\lambda$ (which holds well even for noise sources considered relatively high by the standards of modern devices), this can be used to estimate an error bound, as we do in Section 6.2.

#### 6.1.3.2 The shot cost

In the preceding section, I ignored the shot cost of evaluating the response at each noise level. Of course, the point of this chapter is that any estimated observables on a quantum device will be limited by a finite shot budget $N$, the magnitude

of which may vary greatly across hardware types. The performance of the error mitigation scheme can be greatly influenced by the allocation of this shot budget across the $m+1$ noise nodes. However, it is shown in Ref. [409] that the optimal shot distribution is given by

$$N_j = N\frac{|\gamma_j|}{\sum_{j=0}^{m}|\gamma_j|}, \tag{6.13}$$

where $N_j$ is the shot allocation at each noise level $\lambda_j$, and the total budget $N$ has been divided such that $\sum_{j=0}^{m} N_j = N$. Denoting the corresponding shot-limited expectation values at these noise levels as $\overline{R}_{\lambda_j}(\theta^*, N_j)$, the shot-limited zero-noise-extrapolated estimate of the response at $\theta^*$ comes from Lagrange interpolation of the set $\{\lambda_j, \overline{R}_{\lambda_j}(\theta^*, N_j)\}_{j=0}^{m}$, yielding

$$\overline{R}_M(\theta^*, N) = \sum_{j=0}^{m}\gamma_j\overline{R}_{\lambda_j}(\theta^*, N_j). \tag{6.14}$$

Since ZNE attempts to denoise the parameter estimate by estimating the *noiseless* response to the external field, one should invert it accordingly as $\hat{\theta}^* = R^{-1}(\overline{R}_M(\theta^*, N))$. One could attempt to devise a more faithful inversion of the error-mitigated response, but as we show in Section 6.2, the shot investment required to do so renders this pointless. Note that to estimate the phase, one must assume that the noiseless response function $R(\theta)$ is known, which may not be practical for very complicated probe states. In Figure 6.2, I illustrate the process of how ZNE may be applied to a noisy quantum sensor. In later sections, I evaluate whether ZNE can be expected to improve the sensitivity of parameter estimation when the overall shot budget is fixed, and compare it to alternative protocols.

### 6.1.4 Hyperparameter optimization

There are many choices to be made when implementing ZNE including extrapolation method, node spacing, and number of nodes. Having already selected Richardson extrapolation, one has many degrees of freedom in the choices of nodes, and some choices will not be performant. These may be enforced by system constraints (e.g. the use of unitary folding requires integer node spacing) or otherwise chosen

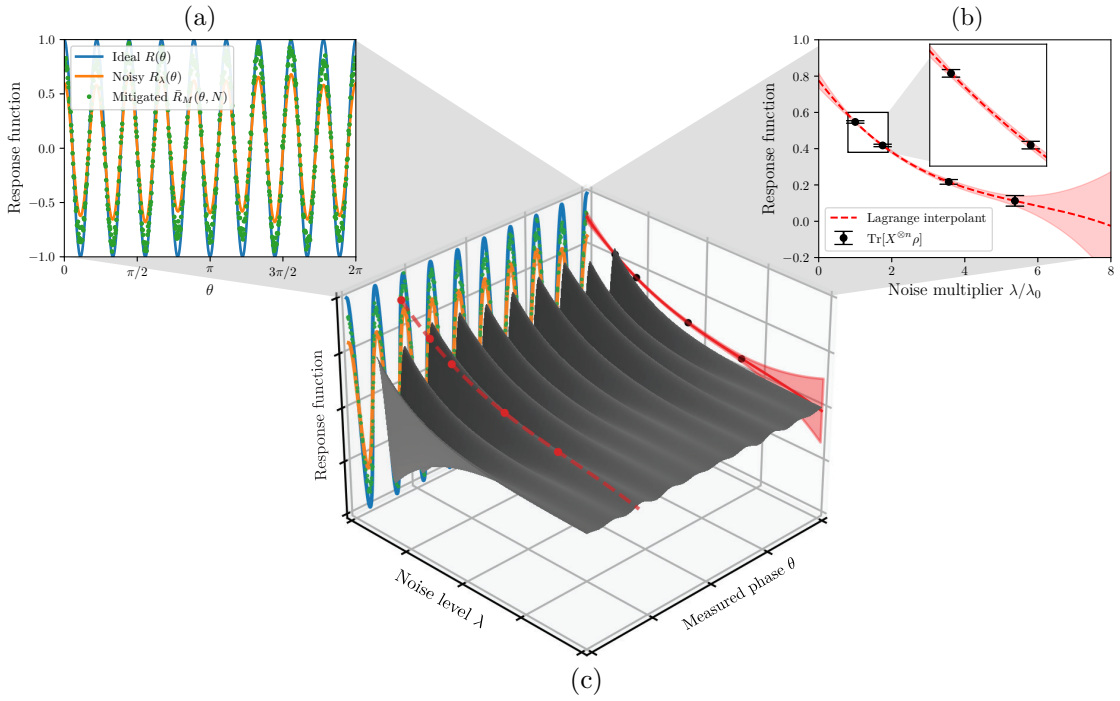



**Figure 6.2: Using zero-noise extrapolation for noisy quantum sensing.** Here I illustrate the use of ZNE to mitigate errors in a GHZ-state magnetometry task. **(a)** The ideal response function is $R(\theta) = \cos(n\theta)$ (blue). When realistic noise is included in state preparation, a noisy response function $R_\lambda(\theta)$ (orange) of smaller amplitude and differing periodicity is obtained. Applying ZNE to the noisy response function yields an imperfect approximation $\overline{R}_M(\theta, N)$ (green) of the ideal response function. **(b)** Fixing the phase at $\theta = \theta^*$, a zero-noise estimate $\overline{R}_M(\theta^*, N)$ is computed by evaluating the noisy response function at multiple boosted noise levels (black nodes) and computing a polynomial interpolant (mean result red dashed line, shaded bars represent 95% confidence intervals). Noise nodes and shots are allocated as described in Section 6.1.3.2. **Inset:** The optimal measurement distribution of Equation 6.13 allocates significantly more shots to lower-noise nodes (since they influence the zero-noise estimate more), leading to comparatively lower variance. **(c)** The zero-noise-extrapolated response function $\overline{R}_M(\theta, N)$ can be determined for any value of $\theta$ with the shot budget $N$ by applying the procedure shown in (b), resulting in the curve depicted in (a).

to optimize performance. In this chapter, I consider Richardson extrapolation with tilted Chebyshev nodes

$$x_j = 1 + \frac{\sin^2\left(\frac{j}{n+1}\frac{\pi}{2}\right)}{\sin^2\left(\frac{1}{n+1}\frac{\pi}{2}\right)}(x_1 - 1), \tag{6.15}$$

which have been proven to minimize variance for a given number of nodes; a thorough investigation of the optimization of Richardson extrapolation for ZNE is given in Ref. [270]. Fixing the node spacing in this manner leaves two hyperparameters: the number of nodes (extrapolation order) $m$, and the position $x_1$ of the first boosted

node. To ensure ZNE is given a fair comparison in this chapter, it is crucial to choose good values of these hyperparameters. However, the optimal choice of hyperparameters depends heavily on noise model and overall shot budget. As it is expensive and impractical to re-optimize the hyperparameters for every new shot budget and noise profile, I select values that are performant across a range of typical conditions in the regimes studied. Since it may be difficult *a priori* to operate the sensor in its region of maximal sensitivity (see Section 6.1.6), it is crucial to optimize for good performance of ZNE across the entire response function. Therefore, I quantify the performance of ZNE by the objective function

$$\epsilon_{x_1,m}(R,\overline{R}_M) \equiv \mathbb{E}\left[\int_0^{2\pi} d\theta \left|R(\theta)-\overline{R}_M(\theta)\right|^2\right], \tag{6.16}$$

generating $\overline{R}_M(\theta)$ from shot-limited estimates of response functions generated with the IBM device noise model used for numerics in Section 6.3 (and also in Section 4.5.2). In Figure 6.3, I plot the objective function $\epsilon_{x_1,m}(R,R_M)$ across a range of overall shot budgets, base noise levels, and hyperparameter choices. Although the optimum hyperparameters depend on noise profile and shot budget, I find that setting $x_1 = 1.75$ and $m = 4$ yields good performance for a variety of scenarios typical for this chapter.

### 6.1.5 Learning-based inference methods

#### 6.1.5.1 The method

As detailed in Section 6.1.1, one of the primary challenges in paramter estimation with noisy quantum sensors is the fact that for general noise models, even for very simple probe states the functional form of the noisy response $R_\lambda(\theta)$ will not be known. To address this issue, the authors of Ref. [286] proposed a learning-based inference method to approximately learn the noisy response function using relatively few measurements (i.e. without performing process tomography or learning the structure of the noise itself). Within the context of the framework outlined in Section 6.1 (which is built upon that of Ref. [286]), this method is applicable when the external field Hamiltonian takes the form $H = \sum_j h_j$ with $h_j^2 = \mathbb{1}$ and $[h_j, h_k] = 0$ for all $j$

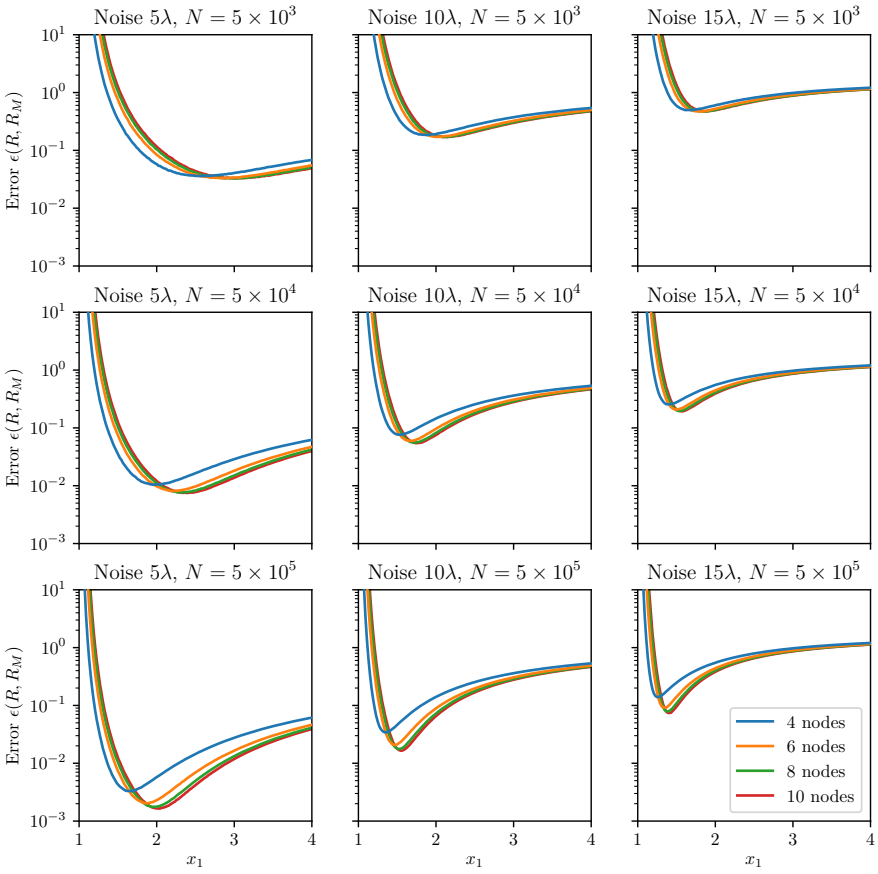



**Figure 6.3: Choosing hyperparameters for ZNE.** I plot the error in the overall mitigated response function $\epsilon_{x_1,m}(R, R_M)$ versus the first boosted node position $x_1$ at varying node counts $m$, for a range of different base noise levels $\lambda_0$ and overall shot budgets $N_M$. Confidence intervals are too small to visualize. Qualitatively, one can observe that higher shot budgets and higher noise levels lead to optima at lower $x_1$.

and $k$. For many practical physical quantities of interest, this is not a particularly strong assumption. As shown in Ref. [286], under these conditions the noisy response function $R_\lambda(\theta)$ takes the form of an $n$-th degree trigonometric polynomial:

$$R_\lambda(\theta) = \sum_{s=1}^{n} [a_s \cos(s\theta) + b_s \sin(s\theta)] + c. \tag{6.17}$$

By estimating the noisy response at $2n + 1$ distinct phases $\{\theta_k\}$, one can use trigonometric interpolation methods to learn the coefficients $a_s$, $b_s$, $c$ — much like how for ZNE, polynomial interpolation methods were used to approximate the response function at differing noise levels. The chief differences here are that (a) we are learning a function in the phase $\theta$, not the noise level $\lambda$ and (b) fortunately, the response function is mathematically guaranteed to take the same form as the

interpolant itself (whereas ZNE truncates the series at order $\mathcal{O}(\lambda^{m+1})$). Analogously to the polynomial interpolation in Section 6.1.3, given $2n+1$ nodes $\{\theta_k, R_\lambda(\theta_k)\}_{k=1}^{2n+1}$, there exists a unique trigonometric polynomial interpolant of degree $n$ that passes through all of these points [408]. Solving the linear system

$$\begin{bmatrix} 1 & \dots & \cos(n\theta_1) & \sin(\theta_1) & \dots & \sin(n\theta_1) \\ 1 & \dots & \cos(n\theta_2) & \sin(\theta_2) & \dots & \sin(n\theta_2) \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 1 & \dots & \cos(n\theta_{2n+1}) & \sin(\theta_{2n+1}) & \dots & \sin(n\theta_{2n+1}) \end{bmatrix} \begin{bmatrix} \tilde{c} \\ \tilde{a}_1 \\ \vdots \\ \tilde{a}_n \\ \tilde{b}_1 \\ \vdots \\ \tilde{b}_n \end{bmatrix} = \begin{bmatrix} \overline{R}_\lambda(\theta_1) \\ \vdots \\ \overline{R}_\lambda(\theta_{2n+1}) \end{bmatrix}, \quad (6.18)$$

one learns the inferred coefficients $\tilde{a}_s, \tilde{b}_s, \tilde{c}$, forming the inferred noisy response function $\tilde{R}_\lambda(\theta)$ given by

$$\tilde{R}_\lambda(\theta) = \sum_{s=1}^{n} [\tilde{a}_s \cos(s\theta) + \tilde{b}_s \sin(s\theta)] + \tilde{c}. \quad (6.19)$$

Since the interpolant and noisy response function are mathematically guaranteed to take the same form, we would have exactly $\tilde{R}_\lambda(\theta) = R_\lambda(\theta)$ if not for shot noise, the effect of which will be discussed in the following section. Crucially, this inference method does not require exhaustive noise characterization (e.g. process tomography), but only knowledge of its effect on the response function — a substantially less demanding requirement. In realistic scenarios where the system's dynamics are not easily accessible (or cannot be characterized at all), this is a far less demanding method to satisfy the fourth assumption of Section 6.1.1. Furthermore, this requires very few assumptions about the probe state, measurement scheme, or noise process, leading to far broader applicability than error-mitigated approaches.

#### 6.1.5.2 The shot cost

Of course, the data used to learn the response function is shot-limited, with a total shot budget $N_I$ divided amongst the $2n+1$ nodes. Throughout this chapter, where the inference budget itself is relevant, I include it in the notation as $\tilde{R}_\lambda(\theta, N_I)$, indicating that a finite-sample inference training set $\{\theta_k, \overline{R}_\lambda(\theta_k, N_k)\}_{k=1}^{2n+1}$ was used. Here, $N_k$ is the individual shot allocation to the $k$-th node, where $\sum_{k=1}^{2n+1} N_k = N_I$;

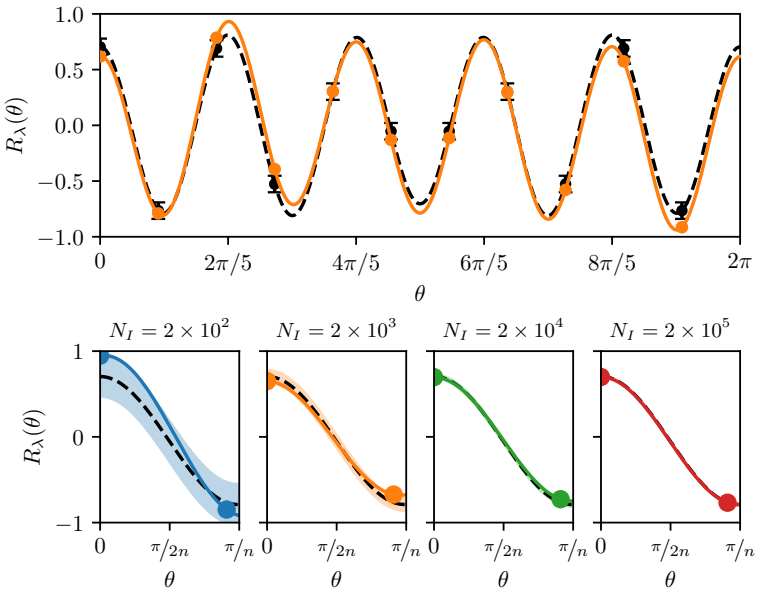



**Figure 6.4: The effect of shot budget on response function inference.** Here I illustrate the inference of a response function for a 5-qubit system following the noise model of Section 4.5.2, at varying inference shot budgets $N_I$. **(Top)** Evaluating exactly $R_\lambda(\theta_k)$ at $2n+1$ nodes $\theta_k$ (black dots) and performing trigonometric interpolation exactly recovers the noisy response function $R_\lambda(\theta)$ (black dashed line). In practice, one has a limited inference shot budget $N_I = (2n+1)N_k$ and obtains $N_k$-shot estimated $\overline{R}_\lambda(\theta_k, N_k)$ (black error bars indicate uncertainty at $N_I = 2 \times 10^3$, orange dots indicate a single realization of sampling at this inference budget). Performing trigonometric interpolation yields an approximate inferred response function $\tilde{R}_\lambda(\theta, N_I)$ (orange line). **(Bottom)** I compare the inferred response function on a single fringe at four inference budgets, where colored dots again represent single realizations of inference nodes, the black dashed line is the exact response $R_\lambda(\theta)$, and shaded regions represent the uncertainty of the inferred response function. The error in inference clearly decreases with increasing $N_I$.

for the purposes of this chapter, I take a uniform allocation $N_k = N_I/(2n+1)$ which was heuristically found to be the most performant. This effect of shot budget on the accuracy of the response function inference is illustrated in Figure 6.4. One can also choose different node spacings (assuming they lead to an invertible Vandermonde-like matrix in Equation 6.18) — well-conditioned linear systems lead to better approximation, and the approximation error is minimized for a uniform spacing (i.e. for the $2\pi$-periodic response functions encountered in this chapter, $\theta_{j+1} - \theta_j = 2\pi/(2n+1)$).

Once a response function has been inferred, one can use it for parameter estimation. Throughout this chapter, I denote the estimation budget as $N_E$: one

measures $\overline{R}_\lambda(\theta^*, N_E)$, and obtains a parameter estimate by inverting it through the inferred response function as $\hat{\theta}^* = \tilde{R}_\lambda^{-1}(\overline{R}_\lambda(\theta^*, N_E))$ (assuming invertibility, which I address in Section 6.1.6). In contrast to other methods, there are now two shot budgets: the inference budget $N_I$, used to learn the response function (independent of the external field or the target phase $\theta^*$) and the estimation budget $N_E$, used to make a measurement to be inverted with this learned response function. The total shot budget is then $N = N_I + N_E$, although there are important subtleties to this division: the estimation step requires the sensor to be exposed to the target field, but the inference step does not. Having performed inference *once*, one could in principle perform parameter estimation on many different fields, reusing this response function. Therefore, assuming the noise model does not drift significantly on the relevant timescale (i.e. the sensor is 'stable'), one can treat inference as a pre-characterization process for the sensor, in which case one can consider the entire shot budget to be allocated to estimation as $N_E = N$ when comparing to other methods. In order to make a fair comparison, and to understand the importance of pre-characterization, I consider both of these protocols throughout this chapter.

### 6.1.6 Parameter measurement via phase measurement

In Section 6.1.1 I mentioned that most treatments conflate the field strength parameter $\alpha$ and the phase it imprints $\theta^*$. While these are closely related, one obtains an estimate of $\theta^*$ from an experiment, and must then use this to inform an estimate of $\alpha$, the actual quantity of interest. For a practical, experimentally-relevant treatment, one must carefully account for this distinction, since failing to do so can have drastic impacts on the apparent sensitivity arising from theoretical analysis. Given that the simple case of GHZ-state mangetometry with parity measurements already showcases some of the key issues, I begin by evaluating this case. In the simplest case, $\alpha$ is related to the imprinted phase as $\theta^* = \alpha T$. Next, note that since the response function is periodic ($R(\theta) = \cos(n\theta) = R(\theta + 2\pi/n)$), it is not invertible over its whole domain $[0, 2\pi]$. Without further restrictions, one cannot even uniquely map expectation values to phases, let alone phases to field

strengths. To resolve this, one needs to restrict to some domain $\theta \in [\theta_{\min}, \theta_{\max}]$ over which $R_\lambda(\theta)$ is bijective. Therefore, estimating the phase as $\hat{\theta}^* = R_\lambda^{-1}(\overline{R}_\lambda(\theta^*))$ only allows us to obtain a value of $\theta^*$ which can be used, at best, to determine the remainder of $\alpha T$ modulo $2\pi/n$. This illustrates the fact that a quantum sensor, on its own, cannot unequivocally determine $\alpha$.

Since most quantum sensors measure some sort of phase information, which is inherently periodic, the best use of such a sensor is to *refine* a prior, but comparatively cruder knowledge of the parameter. That is, quantum sensors can reduce the error in the determination of a given parameter for which one has a prior estimate $\alpha_{\text{prior}}$, usually obtained by classical means. Indeed, having access to $\alpha_{\text{prior}}$ enables one to determine suitable $\theta_{\min}$ and $\theta_{\max}$, and thus successfully perform the response function inversion. Generally, when noise is accounted for, the period of the function might change (e.g. in the case of GHZ magnetometry, the symmetry of individual response function fringes will be broken) and the sensitivity $\Delta^2\theta$ of the sensor can vary with $\theta$ [286]. In this case, one needs to determine the domain over which the response will be inverted, as well as guarantee that the sensor is tuned as close to the point of optimal sensitivity as possible. If knowledge of the functional form of the response function is available (e.g. because one has used inference techniques for pre-characterization), then one can identify its most sensitive region (6.5), and intentionally imprint an additional phase $\theta_{\text{bias}}$ to shift the sensor towards that highly sensitive region, if required. The target phase would then become $\theta^* = \alpha T + \theta_{\text{bias}}$. Physically, this could correspond to the application of continuous rotation gates, or the application of a bias field or frequency chirp as seen in quantum sensing experiments [410, 411].

This additional phase $\theta_{\text{bias}}$ used to tune the sensor to its optimal sensitivity will typically depend on the prior estimate $\alpha_{\text{prior}}$, and therefore the modelling of this chapter must make assumptions regarding how well the field is already known. Throughout this work, I assume that whenever an $N$-shot quantum measurement protocol is executed, it is initialized with an optimal $N$-shot classical measurement $\alpha_{\text{prior}}$ that saturates the SQL (i.e. the best possible result without entanglement has

already been achieved, such that I directly assess whether entanglement is providing an advantage). Due to the (classical) measurement uncertainty in $\alpha_{\text{prior}}$, the target phase for the quantum sensing protocol will have an innate randomness, and I acknowledge this fact by introducing the phase random variable $\Theta^*$. Additionally, in practice one will not be able to apply $\theta_{\text{bias}}$ to arbitrary precision and will at best be limited by e.g. the resolution of the digital-to-analog converter used in the control system, leading to some rounding error $\epsilon_B$. Combining these effects, I assume that for a single run of any sensing protocol, the target phase is drawn from a normal distribution as

$$\Theta^* \sim \mathcal{N}\left(\alpha_{\text{prior}} T + \theta_{\text{bias}} + \epsilon_B, \frac{1}{Nn}\right), \tag{6.20}$$

where the rounding error is $\epsilon_B = 2\pi/2^B$ for $B$ bits of precision, and we assume a precision $B = 10$ throughout (which is a generous assumption for rotation gates on even late-NISQ hardware [412]). I emphasize that the randomness of this distribution originates entirely from uncertainty in the classical prior measurement, and not instability in the field itself, which I do not model. Crucially, each run of a quantum sensing protocol will be conditioned on a particular value of $\Theta^*$ drawn from this distribution, requiring me to consider expected values over this distribution in Section 6.2 and Monte Carlo sampling thereof in Section 6.3.

## 6.2 Analytic estimates of error bounds

In this section, I present different noisy quantum sensing protocols based on error mitigation, inference, and combinations thereof. I depict the different protocols, and the conditions under which they are relevant, in Figure 6.5. The goal is to derive and compare error bounds for each protocol, yielding qualitative insight into whether or not they are performant, and why. These bounds also enable a back-of-the-envelope, shot-for-shot comparison of the protocols, providing basic insight to their relative performance — this is then refined by numerical simulations in Section 6.3.

I begin with definitions used to quantify error. There are multiple independent sources of randomness in this framework — the combined shot budget of naïve

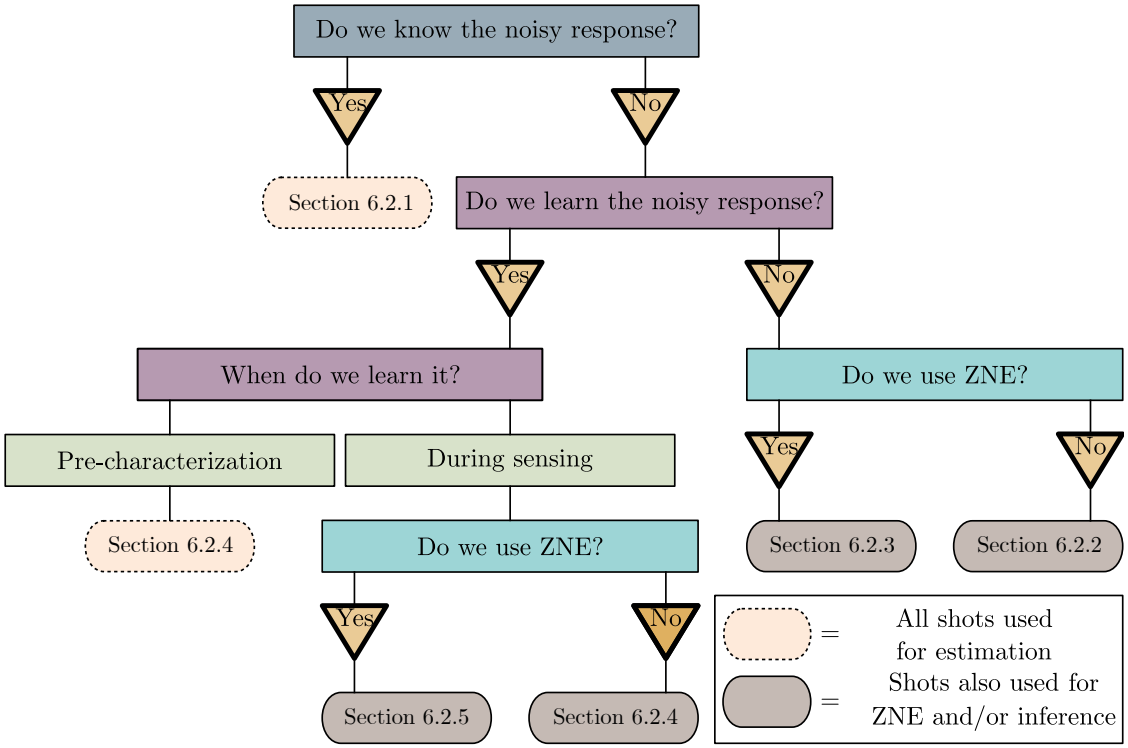



**Figure 6.5: Flowchart of the protocols studied in this chapter.** This flowchart explains the connection between the different sensing protocols presented in Section 6.2, outlining the choices and conditions leading to each protocol, as well as whether I find evidence that the protocol can provide an advantage by outperforming the SQL.

and error-mitigated protocols, the potentially separate inference and estimation budgets of inference-based protocols (Section 6.1.5), and the classical randomness of the prior measurement (Section 6.1.6). Therefore, a standard definition of MSE (Equation 2.12) will not suffice, since a single stochastic average $\mathbb{E}$ cannot capture the subtlety of how these sources of randomness interact. I build upon the approach of Refs. [178–180], defining the conditional mean squared error (CMSE) for a phase estimate $\hat{\theta}^*$ as

$$\mathrm{CMSE}[\hat{\theta}^*|\theta^*] \equiv \mathbb{E}_N \left[ (\hat{\theta}^* - \theta^*)^2 \middle| \, \Theta^* = \theta^* \right], \tag{6.21}$$

where as the name implies, this error is conditioned on a paticular value of the phase random variable $\Theta^*$ such that $\Theta^*$ assumes a value of $\theta^*$ from its sample space. The CMSE can be thought of as the mean error for a *single* run of a quantum sensing protocol, conditioned on a particular value of the classical prior estimate used to bootstrap it. The average $\mathbb{E}_N$ indicates averaging over multiple runs of

phase estimation (with each run having an $N$-shot budget). To determine the average performance over all classical priors, and therefore fully incorporate the effect of the prior classical measurement on the accuracy of the quantum sensing protocol, I define the Bayesian mean squared error (BMSE) as

$$\begin{aligned} \mathrm{BMSE}[\hat{\theta}^*] &\equiv \mathbb{E}_{\Theta^*}\left[\mathrm{CMSE}[\hat{\theta}^*|\theta^*]\right] \\ &= \int d\theta^* \mathcal{P}_{\Theta^*}(\theta^*)\,\mathrm{CMSE}[\hat{\theta}^*|\theta^*], \end{aligned} \tag{6.22}$$

where $\mathbb{E}_{\Theta^*}$ indicates an average over possible random values of the target phase and $\mathcal{P}_{\Theta^*}$ is its probability density (for the purposes of this chapter, $\Theta^*$ is normally distributed as in Equation (6.20)). I emphasize that in this framework, the CMSE is itself a random variable and a function of $\Theta^*$, whilst the BMSE is a posterior mean-squared error accounting for all sources of randomness in the protocol (both that of the prior classical initialization and the shot noise realizations in the quantum sensing protocol). Therefore, the BMSE quantifies the overall performance of a sensing protocol via a single number, enabling useful comparisons. This is also a metrologically informative quantity, since minimizing the BMSE guarantees that one can asymptotically approach the Cramér-Rao bound [413]. In the remainder of Section 6.2, our derivations will fix the value of $\Theta^*$ at some deterministic but arbitrary $\theta^*$ and derive the CMSE for each considered protocol, which can then be averaged for a particular shot budget to numerically obtain the BMSE for comparison.

I note also that the CMSE, much like the standard definition of MSE (2.12), decomposes into variance and bias terms as

$$\mathrm{CMSE}[\hat{\theta}^*|\theta^*] = \underbrace{\mathbb{E}_N\left[(\hat{\theta}^* - \mathbb{E}_N[\hat{\theta}^*|\theta^*])^2\right]}_{\mathrm{Var}_N[\hat{\theta}^*|\theta^*]} + \left(\underbrace{\mathbb{E}_N[\hat{\theta}^*|\theta^*] - \theta^*}_{\mathrm{Bias}_N[\hat{\theta}^*|\theta^*]}\right)^2. \tag{6.23}$$

This is proven by my collaborator in Appendix D1 of our full manuscript [401]. Therefore, throughout the derivations below, I focus on keeping track of these individual contributions, such that the impact of the (conditional) bias-variance tradeoff can be qualitatively understood. Furthermore, these derivations assume that the additional conditions for inference (commuting Hamiltonian terms with

$\|O\|_\infty \leq 1$) are satisfied, and that the point of optimal sensitivity is relatively close to the point of maximum response function gradient (as is the case in many practical sensing scenarios [410]).

### 6.2.1 Noise-aware noisy quantum sensing

The first protocol considered is an idealized one, where the functional form of the noisy response $R_\lambda(\theta)$ is perfectly known, and $N$ shots are used to estimate the response at the target phase $\theta^*$, which is then inverted via this known response. This could arise in the unlikely scenario where the noise model and probe state are sufficiently simple to derive the response analytically. More importantly for this chapter, this protocol is also the idealized limit of pre-characterized inference (i.e. the limit $N_I \to \infty$, $N_E = N$ for the protocol in Section 6.1.5), whereby one assumes that the response function has been perfectly inferred and all errors are incurred due to shot limits in the estimation step. Conditioned on a target phase $\Theta^* = \theta^*$, the phase estimate is therefore given by $\hat{\theta}^* = R_\lambda^{-1}(\overline{R}_\lambda(\theta^*, N))$. The conditional bias is exactly zero, and the conditional variance is given by the error propagation formula (Equation 6.5), leading to a CMSE of

$$\begin{aligned} \mathrm{CMSE}[\hat{\theta}^*|\theta^*] &= \frac{\Delta^2 R_\lambda(\theta^*)}{N(\partial_\theta R_\lambda(\theta)|_{\theta=\theta^*})^2} \\ &\geq \underbrace{\frac{\Delta^2 R_\lambda(\theta^*)}{N L_\lambda^2}}_{\mathrm{Var}_N[\hat{\theta}^*|\theta^*]}, \end{aligned} \tag{6.24}$$

where the bound in the second line comes from noting the Lipschitz constant of $R_\lambda(\theta)$ on the relevant invertible domain

$$L_\lambda \equiv \sup_{\theta \in [\theta_{\min}, \theta_{\max}]} |\partial_\theta R_\lambda(\theta)|. \tag{6.25}$$

Since by assumption $\|O\|_\infty \leq 1$, it follows that $L_\lambda \in \Theta(n)$ — this is proven by my collaborator in Appendix D2 of our full manuscript [401].

### 6.2.2 Naive noisy quantum sensing

Of course, as outlined in Section 6.1.1, it is usually impractical to assume that the functional form of the noisy response $R_\lambda(\theta)$ is known. Furthermore, if one were to wrongly assume the response's functional form, even slightly, this would induce a bias. In this setting, we assume that noise acts during the sensing process, but the noisy response $R_\lambda(\theta)$ is not known, and instead one naively inverts an $N$-shot estimated $\overline{R}_\lambda(\theta^*, N)$ via the noiseless response function $R(\theta)$ (which might be known analytically for simple probe states, e.g. the GHZ state). This setting can also model the case where one wrongly assumes a functional form for the noisy response by substituting $R(\theta)$ with the incorrectly assumed form. The phase estimated by this protocol is $\hat{\theta}^* = R^{-1}(\overline{R}_\lambda(\theta^*, N))$, leading to not only a (conditional) variance, but also a (conditional) bias

$$\mathrm{Bias}_N[\hat{\theta}^*|\theta^*] = \left| \mathbb{E}_N[R^{-1}(\overline{R}_\lambda(\theta^*, N))] \; - \; R_\lambda^{-1}(\overline{R}_\lambda(\theta^*, N)) \right|.$$

A lower bound on this bias can be computed using a local linearization, yielding

$$\begin{aligned} \mathrm{Bias}_N[\hat{\theta}^*|\theta^*] &= \left| \mathbb{E}_N \left[ \frac{\overline{R}_\lambda(\theta^*, N) - R(\theta^*)}{\partial_\theta R(\theta)|_{\theta=\theta^*}} \right] \right| \\ &\geq \frac{|R_\lambda(\theta^*) - R(\theta^*)|}{L}, \end{aligned} \tag{6.26}$$

where

$$L \equiv \sup_{\theta \in [\theta_{\min}, \theta_{\max}]} |\partial_\theta R(\theta)| \tag{6.27}$$

is the Lipschitz constant of the noiseless $R(\theta)$, in analogy to Equation 6.25. Much like the previous protocol, the variance is given by the error propagation formula (6.5), leading to a CMSE of

$$\mathrm{CMSE}[\hat{\theta}^*|\theta^*] \geq \underbrace{\frac{\Delta^2 R_\lambda(\theta^*)}{NL^2}}_{\mathrm{Var}_N[\hat{\theta}^*|\theta^*]} + \left( \underbrace{\frac{|R_\lambda(\theta^*) - R(\theta^*)|}{L}}_{\mathrm{Bias}_N[\hat{\theta}^*|\theta^*]} \right)^2. \tag{6.28}$$

A few remarks on the form of this bound are warranted. The numerator of the variance term depends on the noisy $R_\lambda(\theta)$, since this originates from the inherent quantum variance, where the noisy response is measured. However, the

denominator depends on the noiseless $R(\theta)$, since it measures the spread caused by the inversion, which is performed with the noiseless response. More importantly, however, the bias term does not depend on the overall shot budget, imposing a hard limit on the sensitivity of this protocol; I show later in Section 6.2.6 that this renders it metrologically unuseful.

### 6.2.3 Noisy sensing mitigated by zero-noise extrapolation

In the previous section, an ultimate limit on the precision was imposed by the mismatch between the noisy response function $R_\lambda(\theta)$ and the noiseless response function $R(\theta)$. Such a bias cannot be entirely eliminated without knowledge of $R_\lambda(\theta)$, which will incur a resource cost. In the following section we will consider this cost; here, we consider the use of error mitigation techniques to reduce the bias when one does not know $R_\lambda(\theta)$. The most obvious protocol would be to use ZNE to obtain an $N$-shot mitigated estimate of the *noiseless* response at the target phase of interest $\overline{R}_M(\theta^*, N)$, and then use the noiseless response function for inversion. Thus, for this protocol, the phase estimate is $\hat{\theta}^* = R^{-1}(\overline{R}_M(\theta^*, N))$. By noting the truncation in Equation 6.12 and assuming that $(m+1)$-th order contribution is nonzero (as it usually will be), then we have

$$R_M(\theta^*) = R(\theta^*) + \Theta(\lambda^{m+1}), \tag{6.29}$$

by which we bound the bias of this protocol as

$$\begin{aligned} \mathrm{Bias}_N[\hat{\theta}^*|\theta^*] &= \left| \mathbb{E}_N \left[ \frac{\overline{R}_M(\theta^*, N) - R(\theta^*)}{\partial_\theta R(\theta)|_{\theta=\theta^*}} \right] \right| \\ &\geq \frac{|R_M(\theta^*) - R(\theta^*)|}{L}. \end{aligned} \tag{6.30}$$

Clearly, $\mathrm{Bias}_N[\hat{\theta}^*|\theta^*] \in \Omega\left(\lambda^{m+1}/L\right)$, indicating that this protocol has reduced the bias from $\Omega(\lambda)$ in the naive unmitigated case (Section 6.2.2) to $\Omega(\lambda^{m+1})$. However, this does not come for free, as we now show via a corresponding increase in the variance term — the inherent bias-variance tradeoff of error mitigation methods discussed in Section 2.3 rears its head. We compute this variance term as

$$\mathrm{Var}_N[\hat{\theta}^*|\theta^*] = \frac{\Delta^2 \overline{R}_M(\theta^*, N)}{(\partial_\theta R(\theta)|_{\theta=\theta^*})^2}, \tag{6.31}$$

where the variance in the $N$-shot mitigated response is given by

$$\Delta^2\overline{R}_M(\theta^*, N) = \sum_{j=0}^{m} \gamma_j^2 \frac{\Delta^2 R_{\lambda_j}(\theta^*, N_j)}{N_j}, \tag{6.32}$$

which follows straightforwardly by taking the variance of Equation 6.11. Clearly, this shows that the variance $\Delta^2\overline{R}_M(\theta^*, N)$ directly depends on both the choice of noise levels $\lambda_j$ and the relative allocation of shot budget to each noise level — intuitively, this makes sense as choosing excessively noisy nodes or allocating too few shots to the most important nodes will compromise performance. If $\lambda$ is overall relatively small, as is typical in most settings (and furthermore assumed by the polynomial truncation in the Richardson extrapolation formalism), one can approximate the variance at each node as being equal (i.e. $\Delta^2 R_{\lambda_j}(\theta^*) \approx \Delta^2 R_\lambda(\theta^*)$); this is proven by my collaborator in Appendix D3 of our full manuscript [401]. This enables a further simplification of Equation 6.32, such that

$$\Delta^2\overline{R}_M(\theta^*, N) \approx \sum_{j=0}^{m} \gamma_j^2 \frac{\Delta^2 R_\lambda(\theta^*)}{N_j} = \frac{\Lambda^2 \Delta^2 R_\lambda(\theta^*)}{N}, \tag{6.33}$$

where in the last step we have used the optimal shot allocation $N_j = N|\gamma_j|/\sum_{j=0}^{m}|\gamma_j|$ of [270] (see Section 6.1.3.2) and defined the 'sampling overhead'

$$\Lambda \equiv \sum_{j=0}^{m} |\gamma_j|. \tag{6.34}$$

Comparing this to the noise-aware protocol (Equation 6.24), we see that the variance of the error-mitigated protocol is larger by a factor of $\Lambda^2$ (since, by construction, $\Lambda \geq 1$). Combining bias and variance terms, the conditional error for this protocol is lower bounded by

$$\mathrm{CMSE}[\hat{\theta}^*|\theta^*] \geq \underbrace{\frac{\Lambda^2 \Delta^2 R_\lambda(\theta^*)}{NL^2}}_{\mathrm{Var}_N[\hat{\theta}^*|\theta^*]} + \underbrace{\left(\Omega\left(\frac{\lambda^{m+1}}{L}\right)\right)^2}_{\mathrm{Bias}_N[\hat{\theta}^*|\theta^*]}. \tag{6.35}$$

### 6.2.4 Inference-based noisy sensing

In the previous two sections, I outlined protocols where one needs to use the noiseless response, as knowledge about the noisy one is unavailable. Not only is it

prudent to deal with the inherent bias of this mismatch between response functions by attempting to learn the noisy response, but this may be necessary anyway if one uses a complicated probe state for which even the noiseless response function cannot be tractably determined, such as a variational probe state [170–181]. I now present our derivations of the error for a scheme where one approximately learns the noisy response via inference techniques (Section 6.1.5) at the cost of some quantum resources. The phase estimate in this protocol will be $\hat{\theta}^* = \tilde{R}_\lambda^{-1}(\overline{R}_\lambda(\theta^*, N))$, where $\tilde{R}_\lambda(\theta)$ is the inferred response function. This estimate will therefore not only depend on variance originating from uncertainty in the response estimation $\overline{R}_\lambda(\theta^*, N_E)$, but also a bias term due to the accuracy of inference — that is, it will depend on $|R_\lambda(\theta) - \tilde{R}_\lambda(\theta, N_I)|$.

Our derivation of error bounds begins with the observation that the inferred response function $\tilde{R}_\lambda(\theta, N_I)$ is a random variable, since it is constructed from a set of random variables — that is, the observations at each inference node $\overline{R}_\lambda(\theta_k, N_k)$ have an uncertainty of $\Delta^2 R_\lambda(\theta_k)/N_k$ which propagates to the inferred function. A resulting error bound (in absolute terms, for individual protocol runs) for the inferred response is given by Ref. [286], which I review below.

**Theorem 6.** *Let $R_\lambda(\theta)$ be the exact noisy response function, and $\overline{R}_\lambda(\theta_k, N_k)$ be the shot-limited response observations at each inference node. Let $\tilde{R}_\lambda(\theta, N_I)$ be the trigonometric polynomial approximation to $R_\lambda(\theta)$ obtained from a single realization of $\{\theta_k, \overline{R}_\lambda(\theta_k, N_k)\}_{k=1}^{2n+1}$ via the inference scheme of Section 6.1.5, where $N_k$ shots are used per inference node such that $\sum_{k=1}^{2n+1} N_k = N_I$. Defining the maximum sampling error as*

$$\epsilon \equiv \max_{\theta_k \in \{\theta_k\}} |R_\lambda(\theta_k) - \overline{R}_\lambda(\theta_k, N_k)|, \tag{6.36}$$

*then we have that for all $\theta$, the inference error is upper bounded as*

$$|R_\lambda(\theta) - \tilde{R}_\lambda(\theta, N_I)| \leq 5\epsilon \log(n). \tag{6.37}$$

*Proof sketch:* Note that $R_\lambda(\theta)$ and $\tilde{R}_\lambda(\theta, N_I)$ are both $n$-degree trigonometric polynomials, and therefore so is their difference $R_\lambda(\theta) - \tilde{R}_\lambda(\theta, N_I)$. Write the

difference in this form, and upper-bound using the general bound on trigonometric interpolation errors [414]. The full proof is given in Appendix IV of Ref. [286]. □

This result may be used to proceed. Per Theorem 6, we know that the error in the inferred response is upper-bounded by $5\epsilon \log(n)$, where $\epsilon$ is the maximum sampling error in any inference node. This is the form of the error bounds given in Ref. [286], but this maximum sampling error $\epsilon$ will vary across different protocol runs, which is an issue given that this chapter is concerned with obtaining comparable bounds averaged over protocol runs. Consequently, using $\epsilon_{\max}$ to denote the largest possible sampling error in the noisy responses in *any* run, then with high probability (in the sense of Hoeffding's theorem) we have

$$\epsilon_{\max} \equiv \max[\epsilon] \approx \max_{\theta_k \in \{\theta_k\}} \frac{\Delta R_\lambda(\theta_k, N_k)}{\sqrt{N_k}}. \tag{6.38}$$

By construction, this bounds (with high probability) the maximum uncertainty in the inferred response in *any* run as

$$|R_\lambda(\theta) - \tilde{R}_\lambda(\theta, N_I)| \leq 5\epsilon_{\max} \log(n). \tag{6.39}$$

Noting that Equation 6.39 depends on the per-node inference budgets $N_k$ via Equation 6.38, it is clear that the inference error can be reduced by increasing $N_I$; one can further straightforwardly derive from this that the noise-aware protocol is obtained in the limit $N_I \to \infty$, as I have previously claimed. Now, I note that these error bounds are in terms of some error $\epsilon_{\max}$ on an absolute scale, which depends on the shot budget $N_I$ — one will typically wish to choose $N_I$ such that the inference error is below some threshold, which hopefully scales asymptotically at the same rate as the HL. Such a threshold is provided in Ref. [286], which I summarize below; in the analysis that follows, I will apply this result to study asymptotic behaviour, taking it beyond its original scope.

**Corollary 3.** *To ensure with a (constant) high probability that the inference error does not exceed $\delta$, for all $\theta$, the number of shots $N_k$ used at any $\theta_k$ should satisfy* $N_k \in \Omega(\frac{\log^3(n)}{\delta^2})$.

*Proof sketch:* Consider the sampling error $|R_\lambda(\theta_k) - \overline{R}_\lambda(\theta_k, N_k)|$ at a single inference point $\theta_k$. First bound the probability that this error exceeds a threshold $\epsilon$ using Hoeffding's inequality. Then, use Boole's inequality to bound the probability that *any* of the $2n+1$ sampling errors (one per $\theta_k$) exceed $\epsilon$. Finally, use Theorem 6 to relate the sampling error to inference error, which results in

$$N_k \geq \frac{50 \log^2(n) \log[(4n+2)/a]}{\delta^2} \tag{6.40}$$

for some small constant failure probability $a$. The full proof is given in Appendix V of Ref. [286]. □

With this, we are able to derive the error bounds for the phase estimate itself. Let $F$ be a function-valued random variable representing the inferred function (whose randomness comes from the shot noise at each inference node), and condition on $\Theta^* = \theta^*$ (i.e. a single realization of the prior classical protocol). Conditioning on a particular $F = f$ from the distribution (i.e. taking a single realization of these shot-limited samples at the $2n+1$ nodes used to construct $\tilde{R}_\lambda(\theta)$), the BMSE follows from the generalized law of total expectation as [415]

$$\mathrm{BMSE}[\hat{\theta}^*] = \mathbb{E}_{\Theta^*}\left[\mathbb{E}_{N_I}\left[\mathbb{E}_{N_E}\left[(\hat{\theta}^* - \theta^*)^2 \middle| \theta^*, f\right] \middle| \theta^*\right]\right], \tag{6.41}$$

where $\mathbb{E}_{N_I}$ represents an average over multiple realizations of the response function inference step, and $\mathbb{E}_{N_E}$ represents an average over multiple realizations of the parameter estimation step. This total error can again be decomposed into conditional bias and variance terms, which we now bound in terms of $\epsilon_{\max}$ — while the somewhat arbitrary $\epsilon_{\max}$ will appear in our final bounds, I will use the scaling from Corollary 3 to evaluate the bounds numerically when comparing all protocols in Section 6.2.6. I note that this approach considers the worst-case performance (whereas most other bounds in this section are average-case), but this is a necessity required to achieve simple, comparable bounds with clear qualitative meaning. Ultimately, this remains a fair comparison, since even with this relative handicap the inference-based protocol proves more performant than its competitors, and our analytic work is supported by numerical study of actual average error bounds in Section 6.3.

Beginning with the (worst-case) conditional bias, we have

$$\begin{aligned}
\text{Bias}[\hat{\theta}^*|\theta^*] &= \mathbb{E}_{N_I}\left[\mathbb{E}_{N_E}\left[\hat{\theta}^*\middle|\theta^*, f\right]\,\middle|\,\theta^*\right] \;-\; \theta^* \\
&= \mathbb{E}_{N_I}\left[\left|\mathbb{E}_{N_E}\left[\tilde{R}_\lambda^{-1}(\overline{R}_\lambda(\theta^*, N_E))\right] \;-\; R_\lambda^{-1}(R_\lambda(\theta^*))\right|\right] \\
&= \mathbb{E}_{N_I}\left[\left|\mathbb{E}_{N_E}\left[\frac{\overline{R}_\lambda(\theta^*, N_E) - \tilde{R}_\lambda(\theta^*, N_I)}{\partial_\theta \tilde{R}_\lambda(\theta)|_{\theta=\theta^*}}\right]\right|\right] \\
&= \mathbb{E}_{N_I}\left[\left|\frac{R_\lambda(\theta^*, N_E) - \tilde{R}_\lambda(\theta^*, N_I)}{\partial_\theta \tilde{R}_\lambda(\theta)|_{\theta=\theta^*}}\right|\right] \\
&\gtrsim \frac{5\epsilon_{\max}\log(n)}{L_\lambda},
\end{aligned} \tag{6.42}$$

where in the last line, we assumed the maximum possible inference error in the numerator and that $\partial_\theta \tilde{R}_\lambda(\theta) \sim L_\lambda$ holds near the point of maximal slope since the inference error will be comparatively small. As one would expect, the bias originates from how well the inferred response function approximates the noisy response function, and can be decreased by increasing $N_I$ (which will decrease $\epsilon_{\max}$). The conditional variance follows accordingly as

$$\text{Var}[\hat{\theta}^*|\theta^*] = \mathbb{E}_{N_I}\left[\text{Var}_{N_E}\left[\hat{\theta}^*\middle|\theta^*, f\right]\,\middle|\,\theta^*\right] + \text{Var}_{N_I}\left[\mathbb{E}_{N_E}\left[\hat{\theta}^*\middle|\theta^*, f\right]\,\middle|\,\theta^*\right], \tag{6.43}$$

the first term of which can be lower bounded as

$$\begin{aligned}
\mathbb{E}_{N_I}\left[\text{Var}_{N_E}\left[\hat{\theta}^*\middle|\theta^*, f\right]\,\middle|\,\theta^*\right] &= \mathbb{E}_{N_I}\left[\frac{\Delta^2 R_\lambda(\theta^*)}{N_E(\partial_\theta \tilde{R}_\lambda(\theta)|_{\theta=\theta^*})^2}\right] \\
&= \frac{\Delta^2 R_\lambda(\theta^*)}{N_E(\mathbb{E}_{N_I}[\partial_\theta \tilde{R}_\lambda(\theta)|_{\theta=\theta^*}])^2} \\
&\gtrsim \frac{\Delta^2 R_\lambda(\theta^*)}{N_E L_\lambda^2},
\end{aligned} \tag{6.44}$$

while the second term is lower bounded as

$$\begin{aligned}
\text{Var}_{N_I}\left[\mathbb{E}_{N_E}\left[\hat{\theta}^*\middle|\theta^*, f\right]\,\middle|\,\theta^*\right] &= \frac{\Delta^2 \tilde{R}_\lambda}{(\partial_\theta \tilde{R}_\lambda(\theta)|_{\theta=\theta^*})^2} &(6.45)\\
&\approx \left(\frac{5\epsilon_{\max}\log(n)}{\partial_\theta \tilde{R}_\lambda(\theta)|_{\theta=\theta^*}}\right)^2 \\
&\gtrsim \left(\frac{5\epsilon_{\max}\log(n)}{L_\lambda^2}\right)^2. &(6.46)
\end{aligned}$$

Considering these three error sources qualitatively, the bias (6.42) represents the worst-case error due to imperfect inference of the response function, the first

term of the variance (6.44) represents the error induced by both shot noise and inherent quantum variance in probing the target field, and the second term of the variance (6.46) represents the worst-case error in sensitivity due to *fluctuations* in the inferred function. Combining these sources, our worst-case lower bound on the CMSE is

$$\mathrm{CMSE}[\hat{\theta}^*|\theta^*] \gtrsim \underbrace{\frac{\Delta^2 R_\lambda(\theta^*)}{N_E L_\lambda^2} + \left(\frac{5\epsilon_{\max}\log(n)}{L_\lambda}\right)^2}_{\mathrm{Var}[\hat{\theta}^*|\theta^*]} + \underbrace{\left(\frac{\mathbb{E}_{N_I}\left[5\epsilon\log(n)\right]}{L_\lambda}\right)^2}_{\mathrm{Bias}[\hat{\theta}^*|\theta^*]}. \tag{6.47}$$

### 6.2.5 Inference-based noisy sensing mitigated by zero-noise extrapolation

In previous sections, we showed that applying ZNE to quantum sensing can reduce — but not eliminate — bias, at the cost of increased variance. Since this bias places an ultimate limit on the efficacy of error-mitigation-based protocols due to the inherent mismatch between the inverting $R(\theta)$ and the actual $R_M(\theta)$, it is reasonable to question whether the protocol might be refined by combining error mitigation with inference-based techniques — that is, by using the techniques of Section 6.1.5 to infer the 'true' error-mitigated response function, using ZNE to evaluate each inference node. In doing so, one would replace the bias with one determined only by the mismatch of $R_M(\theta)$ and some inferred $\tilde{R}_M(\theta)$, i.e. limited by the shot noise in the inference step in the same manner as Section 6.2.4. For this protocol, the phase is estimated as $\hat{\theta}^* = \tilde{R}_M^{-1}(\overline{R}_M(\theta^*, N_E))$. Our derivation of error bounds rely upon the assumption that $R_M(\theta)$ is also a trigonometric polynomial of degree $n$; unlike $R_\lambda(\theta)$, there is no mathematical guarantee of this. Since I find in Section 6.2.6 that this protocol is not performant, this strong assumption is not of concern — since failing to meet it would only make it worse.

My collaborator derives an error bound for this case in our full manuscript [401], in much the same manner as we did for the previous four protocols above. Generalizing Theorem 6 to the error-mitigated response using Bernstein's inequality [416],

deriving a corollary analogous to Corollary 3 for this case, and following our approach from Section 6.2.4, one obtains the CMSE bound

$$\mathrm{CMSE}[\hat{\theta}^*|\theta^*] \gtrsim \underbrace{\frac{\Lambda^2\Delta^2 R_\lambda(\theta^*)}{N_E L^2} + \left(\frac{5\chi_{\max}\log(n)}{L^2}\right)^2}_{\mathrm{Var}[\hat{\theta}^*|\theta^*]} + \underbrace{\left(\frac{\mathbb{E}_{N_I}\left[5\chi\log(n)\right]}{L}\right)^2}_{\mathrm{Bias}[\hat{\theta}^*|\theta^*]}, \quad (6.48)$$

where

$$\chi \equiv \max_{\theta_k\in\{\theta_k\}} |R_M(\theta_k) - \overline{R}_M(\theta_k, N_k)| \quad (6.49)$$

is the largest sampling error across nodes for a set of mitigated response in a single inference protocol run (analogous to $\epsilon$ in Equation 6.36), and

$$\chi_{\max} \equiv \max[\chi] \approx \max_{\theta_k\in\{\theta_k\}} \frac{\Lambda\Delta R_\lambda(\theta_k)}{\sqrt{N_k}} \quad (6.50)$$

is, with high probability, the largest possible such sampling error in any run (analogous to $\epsilon_{\max}$ in Equation 6.38). I note the appearance of the error-mitigation sampling overhead $\Lambda$ in all three terms, originating from the use of ZNE to mitigate the noisy estimate (for the first varaince term) and the use of ZNE to compute the inference nodes (in the bias term and second variance term). Directly comparing the error bound of the error-mitigated, inference-based protocol (6.48) to that of the unmitigated inference-based protocol (6.47), it is obvious that the functional form is essentially the same, but boosted by a factor of $\Lambda^2$ on each term. Therefore, the shots invested in this combination of error mitigation and inference are ultimately wasted, since the attempt to eliminate the remaining bias in error mitigation results in an overhead a factor of $\Lambda^2$ worse than simply accepting the noise and using inference.

### 6.2.6 Comparison of error bounds

Having derived error bounds for the different protocols considered, I am now in a position to make an approximate comparison between them. Importantly, I note that the goal here is to gain intuition for which methods lead to the best performance (in terms of error versus system size $n$) as well as about their relative sensitivities. Throughout this section, I assume that the global-depolarizing noise presented in Section 6.1.2 acts throughout the sensing scheme. For convenience, recall the five

protocols studied above (see also Figure 6.5 to determine when each should be used): noise-aware noisy sensing (Section 6.2.1), naive noisy sensing (Section 6.2.2), error-mitigated sensing (Section 6.2.3), inference-based sensing (Section 6.2.4), and error-mitigated, inference-based sensing (Section 6.2.5). Then, in the case of inference-based sensing, I further consider two cases: a case where the sensor is only operated at the time of parameter estimation (thus total shot budget $N = N_E + N_I$), and the 'pre-characterized' case described in Section 6.1.5 with $N_E = N$ (i.e. all budgeted shots are expended for parameter estimation) and some prior shot budget $N_I$ has already been expended to characterize the response function. Here, I set $N_I = C_{\text{pre}} \times n \times N$, where the constant $C_{\text{pre}}$ is the 'pre-characterization overhead'. In this section I use $C_{\text{pre}} = 100$, but I note that this is a significant overestimate (due to the worst-case nature of the error bound derived in Section 6.2.4, which is very unfavourable compared to the average-case behaviour), and a substantially lower overhead will be found to be sufficient in the numerical study of Section 6.3.

In Figure 6.6, I compare the analytical error bounds of all protocols versus the system size $n$. I provide a summary of the exact expressions used to evaluate these in Appendix B under the assumption of depolarizing noise. I use a shot budget of $N \propto n^2 \log(n)^3$, in line with the requirements to achieve Heisenberg-like scaling with an inferred response function (see Corollary 3). Additionally, I apply a noise scaling of $\lambda = 0.1\sqrt{n}$. Crucially, I note that all protocols that lack knowledge of the system's noisy response function fail to outperform the SQL, highlighting the importance of pre-characterization in achieving tangible advantage from entangled probe states. As expected, the noise-aware sensing protocol (blue) is the best-performing protocol, as it reaches sub-SQL errors within the considered parameter regime. In contrast, naive noisy sensing (red) has the overall worst performance due to its large bias, highlighting the importance of either learning a good approximation of the noisy response function (e.g. via inference) or ensuring the measurement approximately obeys a known noiseless response function (e.g. error mitigation or full error correction). Crucially, the error-mitigated protocol (orange) and noisy inference-based sensing (solid green) protocols also fail to outperform the SQL. I

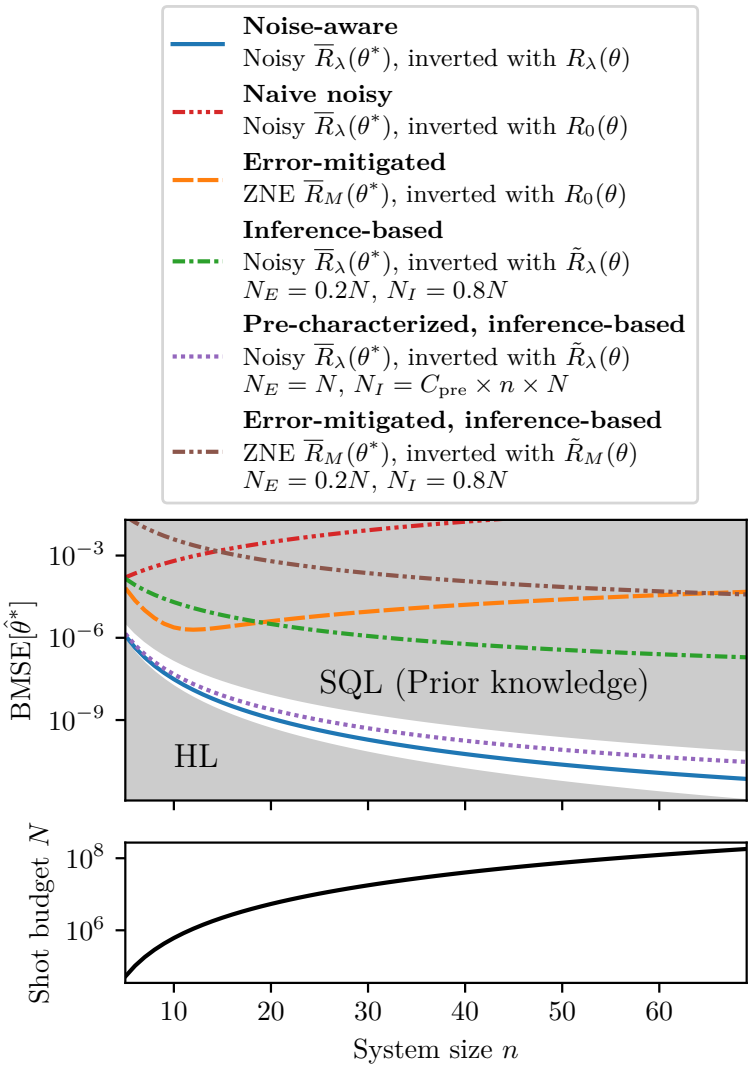



**Figure 6.6: Comparison of analytically estimated error bounds.** I analytically estimate the errors (BMSE) of all protocols considered in Section 6.2 (top panel), comparing errors to the SQL and the HL. I assume that all protocols are initialized with prior knowledge at the SQL, as outlined in Section 6.1.6. I further assume a shot-budget scaling of $N \propto n^2 \log(n)^3$ (bottom panel), and a noise scaling of $\lambda = 0.1\sqrt{n}$.

also note that error-mitigated inference (brown) is not performant, as predicted in Section 6.2.5. Although it eventually outperforms sensing with error mitigation alone (orange) at larger system sizes where the bias of ZNE is a bottleneck, its performance falls well short of the SQL and is always worse than noisy inference-based sensing (green). Overall, in the realistic scenario where one does not know $R_\lambda(\theta)$, then the only protocol to outperform the SQL here is pre-characterized inference (purple).

Dividing the error into bias and variance terms enables additional insight into

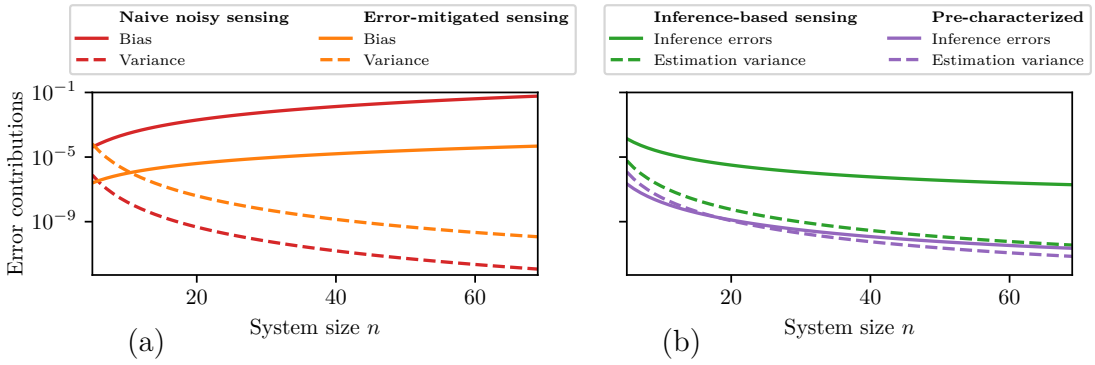



**Figure 6.7: Individual error contributions in parameter estimation.** I show the different terms that affect the error estimations of Figure 6.6. **(a)** I compare naive noisy sensing (red) to error-mitigated sensing (orange). Solid lines indicate bias terms, and dashed lines indicate variance terms. **(b)** I compare inference-based sensing (green) to pre-characterized, inference-based sensing (purple). Solid lines indicate error originating from the inference protocol (this includes both bias and inference variance), while dashed lines indicate estimation variance.

the limiting sources of error in each protocol, and the trade-offs involved. As shown in Figure 6.7(a), I observe the bias-variance trade-off inherent to error mitigation methods (see Section 2.3), which ultimately limits the effectiveness of ZNE for sensing applications. In Figure 6.7(b), I plot inference error (Equation 6.42) and estimation variance error (Equation 6.44). Here, one can see that pre-characterization greatly reduces the error in the dominant terms of inference-based sensing. Furthermore, I note that in contrast to error mitigation, there is no trade-off associated with this improvement: all error terms are diminished by the application of pre-characterization, including bias. This provides an intuitive explanation for the comparatively good performance of pre-characterized inference methods.

I emphasize that the error comparisons of this subsection are intended only as a back-of-the-envelope, approximate, intuition for the factors affecting the performance of these protocols. In particular, as stated previously (and further outlined in Appendix B), the error of inference-based sensing is particularly overestimated. However, as I will show in the next section, in numerical simulations including realistic noise, the errors achieved via inference are much smaller.

## 6.3 Numerical experiments

In the previous section, I presented our analytically derived error bounds for the considered quantum sensing protocols, enabling a back-of-the-envelope comparison of the performance for every scheme. While such analysis provides useful insight, one can only draw limited conclusions from such worst-case studies (e.g. the errors in inference-based protocols are substantially overestimated). Furthermore, the results plotted in Section 6.2.6 are based on a simple global-depolarizing noise model. Such an oversimplified noise model preserves the cosine shape of the response function (merely reducing its amplitude), whereas more general and realistic noise models can significantly change the functional form of the noisy response function $R_\lambda(\theta)$, introducing shifts, breaking the symmetry of invertible regions and creating vast differences in sensitivity between them [286]. Inference-based sensing protocols are particularly well-equipped to deal with these more complex situations. It is therefore prudent to compare the considered protocols on more realistic noise models that capture essential features of response functions encountered on real hardware, which cannot be evaluated analytically. Therefore, in this section I support our analytic results with numerical experiments, explicitly simulating parameter estimation protocols for a noisy GHZ magnetometry task. In all cases I explicitly follow the procedures defined in Section 6.1: e.g. for ZNE I explicitly evaluate shot-limited estimates at each noise level and perform Richardson extrapolation (using the hyperparameters selected in Section 6.1.4), and for inference I fully construct the inferred response function $\tilde{R}_\lambda(\theta)$ by solving Equation 6.18 at each random realization of the inference step. For every protocol studied in this section, I compute the BMSE by direct Monte Carlo sampling of the definition in Equations 6.21 and 6.22. In this section, I restrict my attention to noise-aware sensing (Section 6.2.1), error-mitigated sensing (Section 6.2.3), and inference-based sensing (Section 6.2.4), excluding other protocols that were demonstrated to perform poorly in Section 6.2.6.

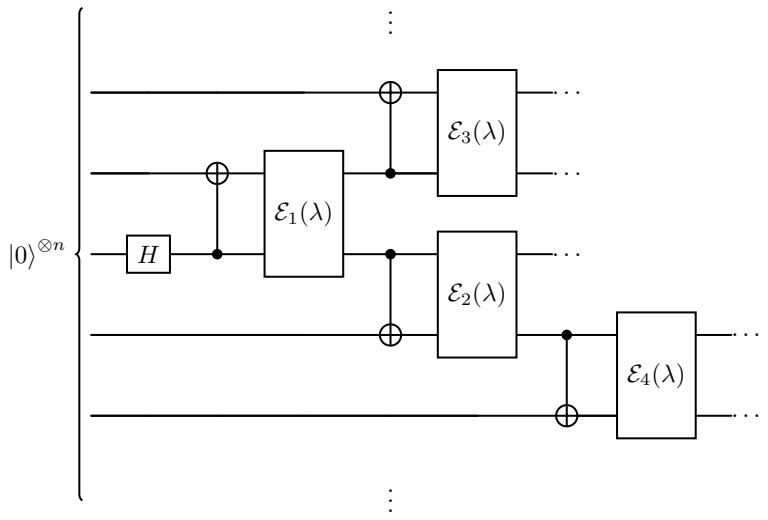



**Figure 6.8: GHZ state preparation channel for numerical experiments.** To prepare a GHZ state using linear-chain CNOT connectivity, I apply a Hadamard at the center of the chain, followed by CNOTs outwards in a ladder fashion. Each CNOT is accompanied by a noise channel $\mathcal{E}_m(\lambda)$, which can be site-dependent.

### 6.3.1 Noise models

Noise is explicitly included in every simulation in this section. In all numerical experiments, I consider noisy preparation of an $n$-qubit GHZ probe state. I assume linear-chain CNOT connectivity and consider the standard preparation circuit involving one Hadamard gate (applied at the center of the chain) and $n-1$ CNOT gates. Each two-qubit gate is accompanied by a CPTP channel $\mathcal{E}_m(\lambda)$ implementing decoherence, and I allow the channels associated with each site to differ in general to model device calibration effects. This state-preparation channel is depicted in Figure 6.8, where the noise channel $\mathcal{E}_m(\lambda)$ is context-dependent. Whenever studying a system that is sufficiently small for full-density-matrix simulations, I utilize a realistic NISQ noise model with coefficients learned from IBM's Eagle processor — this noise model was outlined in Section 4.5.2, and was also used for the NISQ simulations in Chapter 4.

When showing the scaling behaviour in system size $n$, a simpler model is necessary, since full-density-matrix simulations are infeasible for even modest $n$ (see Figure 2.1). In this simplified model, two-qubit gates in state preparation

are accompanied by a local two-qubit depolarizing noise of fault probability $p$. Writing the $m$-qubit GHZ state as $\rho_m^{(\mathrm{GHZ})}$, after applying a Hadamard, the state is $\rho_1^{(\mathrm{GHZ})} \otimes (|0\rangle\langle 0|)^{\otimes n-1}$. Applying a CNOT followed by a two-qubit depolarizing channel, the state becomes

$$\left((1-p)\,\rho_2^{(\mathrm{GHZ})} + p\frac{\mathbb{1}}{2^2}\right) \otimes (|0\rangle\langle 0|)^{\otimes n-2}\,. \tag{6.51}$$

Applying the second CNOT and depolarizing channel yields

$$\left((1-p)^2\,\rho_3^{(\mathrm{GHZ})} + (1-p)\,\rho_1^{(\mathrm{GHZ})} \otimes \frac{\mathbb{1}}{2^2} + p\frac{\mathbb{1}}{2} \otimes (|00\rangle\langle 00| + |11\rangle\langle 11|)\right) \otimes (|0\rangle\langle 0|)^{\otimes n-3}\,, \tag{6.52}$$

and so forth. Iterating this yields a state

$$(1-p)^{n-1}\,\rho_n^{(\mathrm{GHZ})} + \ldots\,, \tag{6.53}$$

where the remaining terms are maximally mixed on the subspace of at least one qubit, and thus have zero contribution to the parity measurement $X^{\otimes n}$. The noisy response function is thus

$$R(\theta) = \mathrm{Tr}\Big[(1-p)^{n-1}\,(e^{-i\theta Z/2})^{\otimes n}\rho_n^{(\mathrm{GHZ})}(e^{i\theta Z/2})^{\otimes n}X^{\otimes n}\Big] = (1-p)^{n-1}\cos(n\theta)\,. \tag{6.54}$$

Unlike the more realistic model of Section 4.5.2, this yields a simple exponential reduction of fringe contrast, and all fringes are identical (that is, due to symmetry in the error model the response function has a period of $2\pi/n$, not $2\pi$).

### 6.3.2 Numerical comparison of errors across varying regimes

The relative performance of the quantum sensing protocols under consideration could in principle depend on properties of the system such as system size, noise level, and shot budget. Here, I use numerical experiments to compare protocol errors across varied regimes, ultimately strengthening the previous conclusions that ZNE typically will not outperform the SQL. As such, in the absence of *a priori* knowledge of the noise model or the response function, only pre-characterized inference can reliably outperform the SQL (and thus obtain actual benefits from quantum entanglement). These numerical experiments are depicted in Figure 6.9.

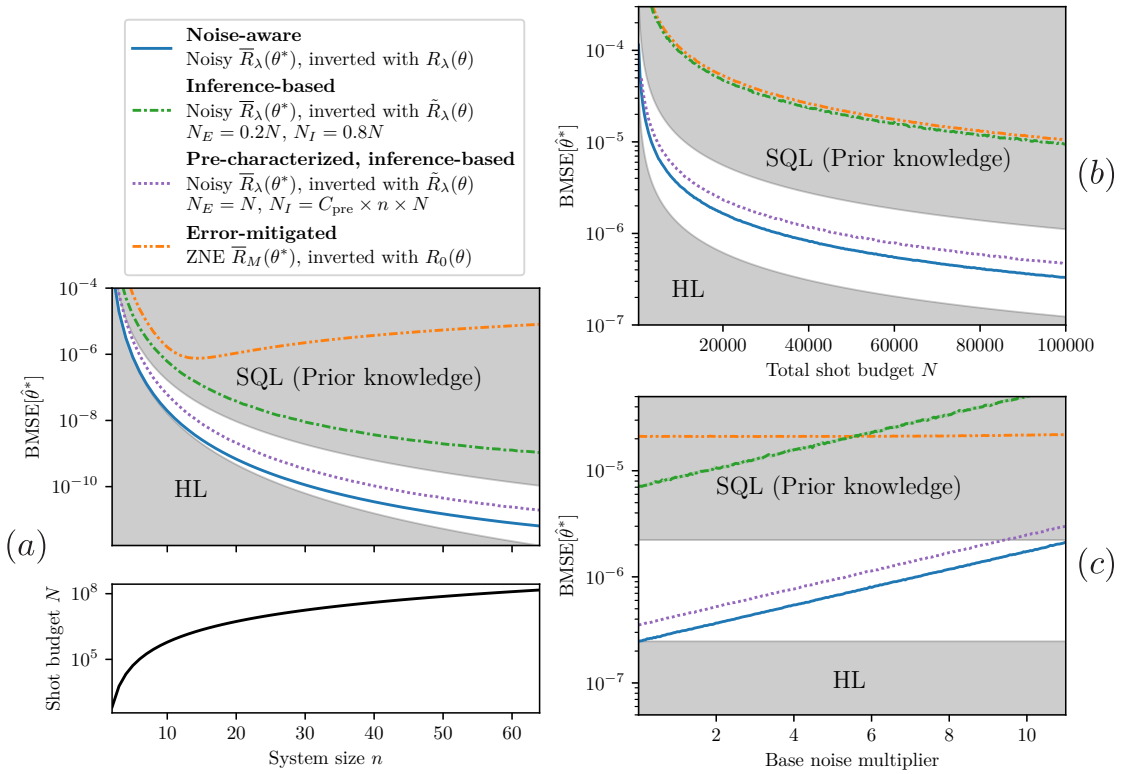



**Figure 6.9: Numerical experiments for magnetometry protocols. (a)** I consider the scaling of protocol errors in system size (top panel), with shot budget scaling $N \propto n^2 \log(n)^3$ (bottom panel). This experiment is conducted with the local-depolarizing error model outlined in Section 6.3.1 at per-gate fault probability $p = 9 \times 10^{-3}$. **(b)** Scaling of protocol errors with the shot budget, at fixed system size $n = 9$ and base noise rate. This experiment is conducted with the IBM Eagle error model outlined in Section 4.5.2. **(c)** Scaling of protocol errors with base noise rate, at fixed system size $n = 9$ and shot budget $N = 5 \times 10^4$. This experiment is conducted with the IBM Eagle noise model outlined in Section 4.5.2. Confidence intervals in all figures are too small to visualize.

In Figure 6.9(a), I compare the numerically computed protocol errors at varying system sizes. Since the comparison reaches system sizes well beyond what could be simulated with full-density-matrix methods, I utilize the local-depolarizing error model of Section 6.3.1, wherein every CNOT in state preparation induces local 2-qubit depolarizing noise at per-gate fault probability $p = 9 \times 10^{-3}$. Once again, to ensure Heisenberg-like scaling in the inference-based protocol, I use a shot budget scaling of $N \propto n^2 \log(n)^3$. Qualitatively, these numerical results can be compared to those obtained from the analytical error bounds in Section 6.2.6 (Figure 6.6). A similar ordering of curves is obtained: ZNE and inference-based sensing fail to outperform the SQL at any system size, but pre-characterized inference does

outperform the SQL. However, as previously suspected, I find that inference-based sensing is overall more performant than the analytic estimates would suggest since they pessimistically assume upper bounds are saturated in inference error. In contrast to Figure 6.6, inference (without pre-characterization) strictly outperforms ZNE here, although it still does not outperform the SQL. Encouragingly, pre-characterized inference outperforms the SQL with a much lower overhead — here, similar performance is obtained with only $C_{\text{pre}} = 1$ (versus $C_{\text{pre}} = 100$ in the analytic estimates of Section 6.2.6). Overall, I find that the qualitative intuition of our analytic error estimates was sound, but inference performs better than one may have believed from the worst-case analytical bounds alone.

While the scaling in $n$ is most important here (i.e. SQL vs HL), it is also crucial to verify that these results are not an artifact of a particular regime — e.g. one may naively expect that ZNE would perform relatively better at a larger shot budget $N$ (where the effect of increased variance is lower) or at a lower base noise level. Furthermore, real experimental noise models will typically differ significantly from the idealized depolarizing models already considered [286]. I therefore perform further numerical experiments, using a realistic noise model based on real current-generation hardware. Noise is simulated using full Kraus maps applied in a density-matrix simulator, with a noise model based on data learned from IBM's Eagle processor (see Section 4.5.2) with the base noise rate boosted by a factor of 5 (in order to operate in a regime where fringe contrast is sufficiently reduced to warrant the use of error mitigation).

In Figure 6.9(b), I compare numerically simulated protocol errors at a fixed system size $n$ and varying shot budget $N$. Here, I find again that inference-based sensing and ZNE fail to outperform the SQL, inference (slightly) outperforms ZNE, and overall, only pre-characterized inference outperforms the SQL, indicating that my previous conclusions were not a product of considering a shot budget regime unfavorable to ZNE.

Finally, in Figure 6.9(c), I compare these errors at a fixed system size $n$ and shot budget $N$, and varying base noise level. I once again observe that inference-

based and error-mitigated sensing do not outperform the SQL, although the error in inference-based sensing does worsen beyond that of error-mitigated sensing at very high noise levels (well exceeding those of current generation IBM hardware, so of lesser relevance given that quantum hardware continues to advance). Pre-characterized inference performs within a constant factor of noise-aware sensing and is again the only protocol that obtains a clear quantum advantage without assuming *a priori* knowledge of the system.

Interestingly, in the limit of a noiseless system, inference-based sensing (without pre-characterization) does *not* outperform the SQL. Hence, even in the absence of noise, there is a cost associated with learning the system's response function. Although in this case (GHZ probe state) the noiseless response function $R_0(\theta)$ is known, there is much ongoing investigation into the use of non-trivial optimized probe states [170–181] for which even the noiseless response function may not be known *a priori*. Therefore, this result expands the relevance of this chapter's conclusions regarding the utility of pre-characterization and learning — it may prove useful even in the absence of noise, if a complicated probe state is used.

### 6.3.3 Requirements for pre-characterized inference-based sensing

Throughout the error comparisons of Sections 6.2.6 and 6.3.2, I consistently observed that in the absence of *a priori* knowledge of the noisy response function $R_\lambda(\theta)$ (which is generally not a practical assumption since device noise is challenging to characterize, and full-process tomography is costly), the only protocol that could reliably outperform the SQL was pre-characterized inference. This can prove crucial in achieving asymptotic advantage from entanglement in quantum sensing. However, it is natural to question whether the costs of pre-characterized sensing are indeed practical, or whether the pre-characterization cost might prove excessive. In previous sections, I considered a simple overhead $N_I = C_{\text{pre}} \times n \times N$, finding that overall $N_I \in \mathcal{O}(nN)$ shots were more than sufficient to pre-characterize a sensor to the point of outperforming the SQL, across a range of regimes spanning

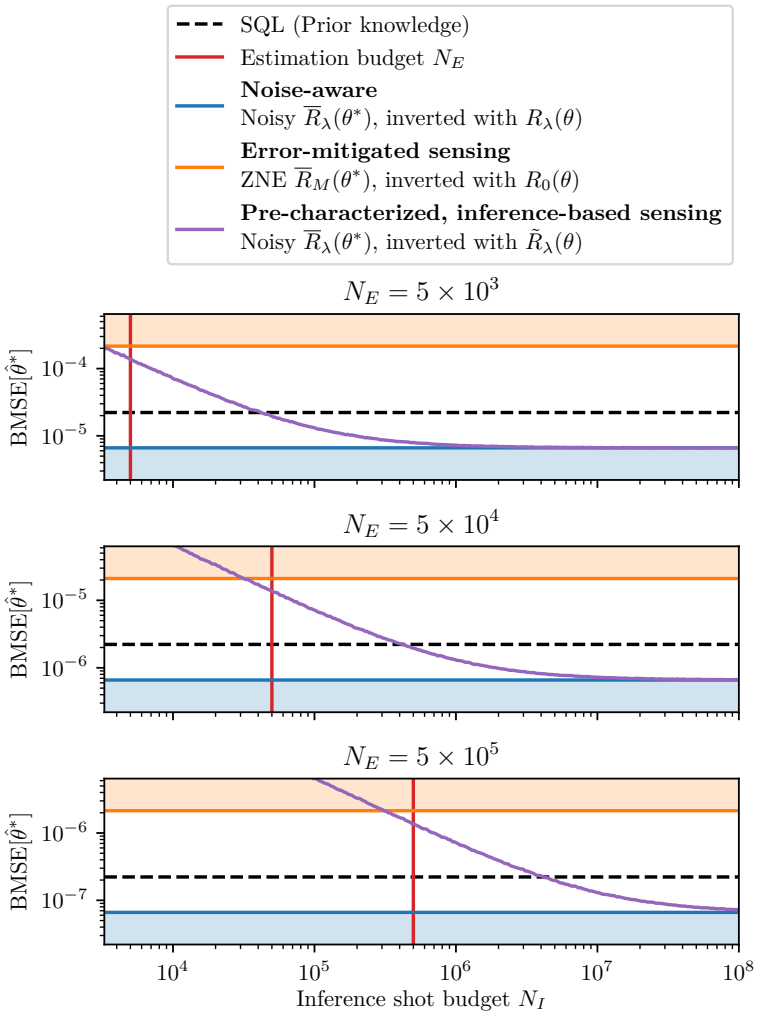



**Figure 6.10: Error convergence of pre-characterized, inference-based sensing.** I numerically study a 9-qubit sensor afflicted by realistic noise in state preparation. At three fixed estimation budgets $N_E \in 5 \times \{10^3, 10^4, 10^5\}$ (red vertical lines), I numerically compute the error for pre-characterize inference-based sensing at a range of inference budgets $N_I$. I further compare this error to that of error-mitigated sensing (orange), noise-aware sensing (blue), and the SQL (black dashed).

different system sizes, shot budgets, noise rates, and error models. Hence, in this section I study the effect of varying pre-characterization shot budgets to better understand the associated overhead.

In Figure 6.10, I consider the use of pre-characterized, inference-based sensing on a 9-qubit device, with state preparation afflicted by noise model learned from an IBM Eagle device (Section 4.5.2) at a base noise rate boosted by a factor of 5. I consider, at three fixed estimation shot budgets $N_E \in 5 \times \{10^3, 10^4, 10^5\}$, the

dependence of inference-based sensing error on the pre-characterization budget $N_I$. I further compare these to the error achieved with error-mitigated sensing at $N = N_E$ and noise-aware sensing at $N = N_E$. As $N_I \to \infty$, I observe that the inference-based protocol converges to the noise-aware sensing protocol, as one would expect from the remarks of previous sections. In this regime, I find that the threshold $N_I$ for the protocol to outperform the SQL is less than an order of magnitude greater than $N_E$, and $N_I$ two orders of magnitude larger than $N_E$ is already sufficient to be extremely close to the noise-aware limit. Furthermore, I find for all three estimation budgets $N_E$ that when $N_I$ is pessimistically chosen to be slightly less than $N_E$, one can still outperform error-mitigated sensing. Overall, I observe that with only very modest pre-characterization overheads, a single pre-characterization can be reused for as many parameter estimations as desired (within the window of noise model stability). This supports the practicality of pre-characterization to achieve quantum advantage with inference-based sensing.

## 6.4 Conclusions

The notion of an entanglement advantage for sensing problems only makes sense in the context of a finite shot budget $N$: per the quantum Cramér-Rao bound (Equation 1.1), even in the classical (unentangled) case, infinite measurements $N \to \infty$ can yield infinite precision. For any technique that claims to enhance quantum sensing — be it error mitigation, learning-based inference, or something else — one must always consider the shot investment required to conduct this enhancement. Depending on the experiment, such methods may or may not be necessary prerequisites to conduct sensing at all (e.g. learning unknown response functions for complicated probe states), but if not strictly necessary, one must always consider whether or not the enhancement was worth the cost, or whether those shots should have just been directly expended in a simpler protocol. Additional overheads, unrealistic assumptions of prior knowledge about the system, and the associated tradeoffs can conceal the true performance of a protocol.

In this chapter, I have attempted to unmask the subtleties of these tradeoffs, specifically considering whether ZNE and/or learning-based inference techniques can enhance quantum sensing on a fixed shot budget. Using analytic and numerical means, I compared the shot-for-shot performance of various protocols, each corresponding to a different level of assumed prior knowledge. Unsurprisingly, perfect prior knowledge of the noisy response function leads to the best performance, but this is generally not a realistic assumption. In the absence of such knowledge, biases can completely compromise the sensor's performance to no better than that of a classical device — naïve assumptions led to poor performance, and while error mitigation reduces these biases, it proved insufficient to achieve a quantum advantage. Learning-based inference techniques fared relatively better, and analytically seemed more promising, but when the shot budget had to be shared between learning and sensing, even this could not outperform the SQL. However, inferring a response function does not require exposure to the field to be measured, and thus could be done before the sensing experiment as a device pre-characterization stage, being reused for many field measurements. In both analytic and numerical studies, I found that when assuming a reasonable shot budget $N_I$ had been expended prior to pre-characterize the sensor and the whole shot budget $N$ was used for parameter estimation ($N_E = N$), inference-based sensing was indeed capable of achieving a quantum advantage by outperforming the SQL.

The latter conclusion relies upon an assumption that is not guaranteed: my analysis assumed that the response function had not changed between the pre-characterization and the posterior sensing experiment. In essence, this is an assumption of *stability*: the structure of the device's noise must be such that, on the timescale of the duration between pre-characterization and sensing, the noise model has not drifted much. This is extremely device-dependent, and a full account of stability across experimental platforms would be a significant undertaking, but many modern quantum hardware platforms have noise that changes quickly enough to require frequent re-calibration. Nonetheless, this is the most significant qualitative insight of this chapter — *stability is a key criterion for a useful quantum sensor.*

A sensor's stability will determine how often it must be re-characterized, and therefore the effective characterization budget $N_I$ that can be used (relative to the time to perform a quantum operation on the device). It is not at all unexpected that one might achieve superior performance on a slower device (capable of fewer measurement repetitions in a fixed time) that is more stable, due to a reduction of the second and third terms in Equation 6.47 outweighing an increase in the first term.

Finally, this chapter has demonstrated that learning-based techniques are particularly valuable in the pursuit of quantum advantage for sensing problems. This may guide future work — it seems probable that future advances in quantum sensing will be made by incorporating tools from QML and classical machine learning, a prospect I will discuss in the conclusions of this thesis (Chapter 7).

*I am glad you are here with me. Here at the end of all things, Sam.*

— Frodo Baggins, *The Return of the King* [417]

# 7 Conclusions

It seems increasingly likely that quantum computers will soon become genuinely useful. Nonetheless, the definitions of 'soon', 'useful', and even 'quantum computer' in this regard are a matter of intense debate. The road towards the dream of fully-fault-tolerant quantum computing is an arduous one, plagued by the challenges of noise, decoherence, and the immense complexity of engineering highly-entangled quantum states. Furthermore, even as we achieve remarkable milestones towards this dream, we still face the question of what algorithms we will even run on these devices. In this thesis, I have contributed to both of these challenges, by studying protocols to deal with the inherent imperfections of quantum devices, by introducing new algorithms that respect and are robust to these limitations, and by clarifying the boundary between quantum and classical.

Beginning with an overview of the landscape of quantum technology in Chapter 1, I introduced the theoretical promises of quantum computing and quantum sensing, as well as the practical challenges and state of affairs at the overlap between the NISQ and early-fault-tolerant eras. Following this with a deeper dive, in Chapter 2 I contextualized my work in terms of the literature on classical simulation of quantum devices, variational quantum algorithms and their trainability, quantum error mitigation, the calculation of spectral properties of quantum systems, and classical shadow tomography protocols.

The individual conclusions of each chapter provide a far more detailed overview, but I briefly recall my contributions here. With these tools in hand, I began presenting my own work in Chapter 3, where I introduced 'shadow spectroscopy', a protocol to estimate eigenvalue gaps of Hamiltonians using only simple time evolution and logarithmic-measurement-cost classical shadow protocols. Using tensor-network and simulated variational approaches, I demonstrated good scalability of the protocol and its successful application on simulated and real near-term quantum computers. Probing further into the use of time evolution to compute spectral properties, in Chapter 4 I introduced a protocol to directly estimate the density of states, a key thermodynamic, on a quantum computer. With a particular focus on early-fault-tolerant devices (and some additional consideration of NISQ deployment), I demonstrated the protocol's simplicity in sampling and evolution requirements, its applicability to high-value fermionic problems, and its robustness to both algorithmic and hardware noise. Then, considering the variational methods that underpin proposals for NISQ advantage and QML, in Chapter 5 I introduced '$\mathfrak{g}$-sim', a framework for Lie-algebraic classical simulation of PQCs. Developing the theoretical framework, I introduced a range of powerful generalizations of previous related results, and I demonstrated its utility for characterizing and pre-training VQAs, compiling unitaries to compact circuits, and supervised QM. Finally, in Chapter 6 I turned my attention towards quantum sensing, conducting a comparative study of error-mitigated and learning-based inference protocols for quantum sensing, and ultimately concluding that sensor pre-characterization via inference is the most promising avenue for noisy quantum sensing.

These studies represent contributions over three very distinct areas of quantum technology: development of NISQ and early-fault-tolerant algorithms for the study of quantum systems, classical simulation of quantum devices, and quantum sensing. With research topics so seemingly disparate, what insights, if any, might be gleaned collectively from my contributions as a whole? First, I note that this thesis raises serious concerns for the prospects of quantum advantage with VQAs on NISQ devices, a goal which was the subject of intense optimism at the start of my doctorate that

has since declined over these past three years. The $\mathfrak{g}$-sim framework I developed in Chapter 5 forms one of the pillars of a broader observation that trainability and classical simulability of VQAs are deeply interlinked [134]. It seems that in the absence of extremely good initialization strategies, the prospects for efficiently-trainable, non-classically-simulable VQAs of actual utility are slim. Therefore, variational dynamics methods (including the one I developed in Section 4.5.1) will likely be dependent on the landscape features that occur as a result of initializing a new timestep with the previous one, which is still a matter of open investigation [263]. The outlook there remains pessimistic, and since the NISQ demonstrations in Chapters 3 and 4 hinge upon this, it seems very possible that these methods will not yield an advantage on NISQ devices.

However, the methods I have introduced in this thesis will likely prove advantageous in the early-fault-tolerant era, the prospects of which seem far more promising. As we show in our full manuscript [334], there is plenty of evidence that shadow spectroscopy will be a genuinely useful technique in early- and fully-fault-tolerant eras, with studies to this end conducted by my collaborators Richard and Hans, respectively. Since my numerical experiments in Chapter 4 showed high levels of resilience to both the algorithmic and hardware errors that will be most prominent in the early-fault-tolerant era, my contributions in Chapter 4 seem likely to prove advantageous.

Furthermore, even if a system is classically simulable, there may be a quantum advantage in extracting information from it. This could be true in the sense discussed in the conclusions of Chapter 5, where a classically-trained, quantumly-deployed classifier could outperform a classical classifier, even where no quantum advantage for learning exists. Another such avenue is in quantum sensing, where Chapter 6 is concerned with metrological advantage from entanglement in parameter estimation tasks. I speculate that an intersection of the two could prove fruitful: since the $\mathfrak{g}$-sim framework can (for Pauli-basis algebras) simulate arbitrary Pauli noise, which a very general class of noise models can be twirled to [340, 341], $\mathfrak{g}$-sim could prove useful for e.g. training optimal measurements or probe states entirely classically.

Even though this would be limited to systems with no quantum *computational* advantage, this does not preclude a quantum *metrological* advantage.

In conclusion, the work I have presented in this thesis represents an incremental, but genuine step in understanding how to harness and exploit the power of imperfect quantum devices. As with all scientific and human endeavours, the long road ahead in quantum technology will be paved with occasional monumental breakthroughs, built upon a foundation of countless smaller steps. It is a humbling privilege to have taken four such steps in this thesis, which I thank you, the reader, for taking the time to understand.

# Appendices

# A

# Lie-algebraic quantum dynamics

## Contents


**A.1 Proof of Lemma 1 . . . . . 228**
**A.2 Proof of Lemma 2 . . . . . 229**
**A.3 Proof of Theorem 3 . . . . . 230**


The $\mathfrak{g}$-sim framework of Chapter 5 builds upon existing work in Lie-algebraic classical simulation [112, 113]. The core primitives for evolving observables in $i\mathfrak{g}$ and their products follow the existing approach (aside from the inclusion of arbitrary initial states, which do not change the method of proof). The relevant proofs are included in this appendix for completeness, such that all proofs presented in the main text of Chapter 5 are genuinely novel.

## A.1 Proof of Lemma 1

Since $U \in \mathcal{G}$, there exists some $iH \in \mathfrak{g}$ such that $U = e^{-iH}$. The exponential mapping is defined by

$$e^{-iH} = 1 - iH + \frac{(iH)^2}{2!} - \frac{(iH)^3}{3!} + \ldots, \tag{A.1}$$

Noting that $G'_\alpha \equiv U^\dagger G_\alpha U$ and applying commutator identities to cancel $U$ and $U^\dagger$ factors, one obtains

$$G'_\alpha = G_\alpha + i\,[H, G_\alpha] - \frac{1}{2}\,[H,[H, G_\alpha]] + \dots\,. \tag{A.2}$$

Since all terms of Equation A.2 are generated by nested commutators of terms of $\mathfrak{g}$ (up to appropriate factors of $i$ for each term), by Definition 2 it follows that $iG'_\alpha \in \mathfrak{g}$.

## A.2 Proof of Lemma 2

Since $U \in \mathcal{G}$, there exists some $iH \in \mathfrak{g}$ such that $U = e^{-iH}$. Furthermore, $U$ may be decomposed into infinitesimal steps

$$U = \lim_{M\to\infty} \prod_{m=1}^{M} U^{(\Delta)}, \tag{A.3}$$

where the infinitesimal unitary is given by

$$U^{(\Delta)} = e^{-i\Delta H} = 1 - i\Delta H + \mathcal{O}(\Delta^2) \tag{A.4}$$

for $\Delta = 1/M$. By Lemma 1 I note that $U^{(\Delta)}$ induces a group operation

$$U^{(\Delta)^\dagger} G_\alpha U^{(\Delta)} = G_\alpha - \sum_\beta v^{(\Delta)}_{\alpha\beta} G_\beta, \tag{A.5}$$

where I have separated out the identity for later convenience without loss of generality. Substituting Equations A.4 and A.5 into Equation A.2, one obtains

$$G_\alpha - \sum_\beta v^{(\Delta)}_{\alpha\beta} G_\beta = G_\alpha + i[\Delta H, G_\alpha] + \mathcal{O}(\Delta^2). \tag{A.6}$$

One may discard vanishing terms in $\mathcal{O}(\Delta^2)$ since this is in the infinitesimal limit $M \to \infty$. Doing so, and taking $\mathrm{Tr}[G_\gamma \dots]$ of both sides of Equation A.6 yields

$$-\sum_\beta v^{(\delta)}_{\alpha\beta} \,\mathrm{Tr}[G_\gamma G_\beta] = i\,\mathrm{Tr}[G_\gamma [\Delta H, G_\alpha]] \tag{A.7}$$

$$\implies \Delta v^{(\delta)}_{\alpha\gamma} = i \sum_p \Delta h_p \,\mathrm{Tr}[G_\gamma, [G_p, G_\alpha]] \tag{A.8}$$

$$\implies v^{(\delta)}_{\alpha\gamma} = i\Delta \sum_k h_k (\bar{\boldsymbol{G}}_k)_{\alpha\gamma} = i\Delta \left(\Phi^{\mathrm{ad}}_{\mathfrak{g}}(H)\right)_{\alpha\gamma}, \tag{A.9}$$

where I have assumed a basis choice such that the $G_\alpha$ are Schmidt-orthonormal (see Definition 4), expanded the adjoint representation, and expanded $H = \sum_p h_p G_p$ for $h_p \in \mathbb{R}$. Substituting the result of Equation A.9 into Equation A.5 reveals that the infinitesimal group operation is determined entirely by the adjoint representation $\Phi^{\mathrm{ad}}_{\mathfrak{g}}(H)$ as

$$U^{(\Delta)\dagger} G_\alpha U^{(\Delta)} = \sum_\beta \left(1 - i\Delta\Phi^{\mathrm{ad}}_{\mathfrak{g}}(H)\right)_{\alpha\beta} G_\beta = \sum_\beta \left(e^{-i\Delta\Phi^{\mathrm{ad}}_{\mathfrak{g}}(H)}\right)_{\alpha\beta} G_\beta + \mathcal{O}(\Delta^2). \tag{A.10}$$

One may again discard terms in $\mathcal{O}(\Delta^2)$ since this is in the infinitesimal limit $M \to \infty$. Noting that the full unitary is obtained by composing infinitesimal steps (Equation A.3), one obtains from Equation A.10 that

$$U^\dagger G_\alpha U = \sum_\beta \left(e^{-i\Phi^{\mathrm{ad}}_{\mathfrak{g}}(H)}\right)_{\alpha\beta} G_\beta = \sum_\beta \left(\Phi^{\mathrm{Ad}}_{\mathcal{G}}(U)\right)_{\alpha\beta} G_\beta, \tag{A.11}$$

which is exactly the desired result.

## A.3 Proof of Theorem 3

First, I seek to compute expectation values of all $\langle G_\alpha G_\beta \rangle$, composing the classical representation $\rho^{(\mathrm{out})}$. These are given by

$$\left(\bar{\boldsymbol{E}}^{(\mathrm{out})}\right)_{\alpha\beta} \equiv \mathrm{Tr}\left[G_\alpha G_\beta \rho^{(\mathrm{out})}\right] = \mathrm{Tr}\left[U^\dagger G_\alpha U U^\dagger G_\beta U \rho^{(\mathrm{in})}\right], \tag{A.12}$$

where I have used trace cyclicity and inserted the identity $UU^\dagger = \mathbb{1}$ in the last step. Noting that $G_\alpha$ and $G_\beta$ are conjugated by elements of $\mathcal{G}$ and applying Lemma 2, one obtains

$$\bar{\boldsymbol{E}}^{(\mathrm{out})} = \sum_{\alpha',\beta'} \left(\Phi^{\mathrm{Ad}}_{\mathcal{G}}(U)\right)_{\alpha,\alpha'} \left(\Phi^{\mathrm{Ad}}_{\mathcal{G}}(U)\right)_{\beta,\beta'} \mathrm{Tr}\left[G_{\alpha'} G_{\beta'} \rho^{(\mathrm{in})}\right], \tag{A.13}$$

which is Equation 5.15 in index form. Equation 5.14 follows trivially from this by taking the left-hand side of Equation 5.14, expanding $O^{(a/b)} = \sum_j (\boldsymbol{w}^{(a/b)})_\alpha G_\alpha$, and identifying matrix operations in index form, thus concluding the proof.

# B

# Evaluating error bounds for quantum sensing protocols

## Contents


**B.1 Noise-aware noisy quantum sensing . . . . . 231**
**B.2 Naive noisy sensing . . . . . 232**
**B.3 Noisy sensing mitigated by zero-noise extrapolation . 232**
**B.4 Inference-based noisy sensing . . . . . 233**
**B.5 Inference-based noisy sensing mitigated by zero-noise extrapolation . . . . . 234**


In Section 6.2, the error bounds for a variety of quantum sensing protocols were given to varying levels of approximation. I compared these error bounds in Section 6.2.6. In this appendix, I detail exactly which expressions are numerically evaluated in the comparison, and how their terms are computed. I use a global depolarizing noise model with fault rate $\lambda$, such that $R_\lambda(\theta) = (1-p_\lambda)\cos(n\theta)$, $\Delta^2 R_\lambda(\theta) = 1-(1-p_\lambda)^2\cos^2(n\theta)$, and $|\partial_\theta R_\lambda(\theta)| = (1-p_\lambda)n|\sin(n\theta)|$ (per Section 6.1.2).

## B.1 Noise-aware noisy quantum sensing

The first line of Equation 6.24 gives

$$\mathrm{CMSE}[\hat{\theta}^*|\theta^*] = \frac{\Delta^2 R_\lambda(\theta^*)}{N(\partial_\theta R_\lambda(\theta)|_{\theta=\theta^*})^2}\,, \tag{B.1}$$

where all terms are exactly known. Since for the purposes of this computation one knows $\partial_\theta R_\lambda(\theta)$, I elect not to approximate the gradient on the denominator by its Lipschitz constant, since this can underestimate the error in the case that the sensor is operating far from the point of maximal gradient.

## B.2 Naive noisy sensing

I elect again not to approximate the gradient by its Lipschitz constant (c.f. Eq. (6.28)), yielding

$$\mathrm{CMSE}[\hat{\theta}^*|\theta^*] = \frac{\Delta^2 R_\lambda(\theta^*)}{N(\partial_\theta R(\theta)|_{\theta=\theta^*})^2} + \left(\frac{|R_\lambda(\theta^*) - R(\theta^*)|}{\partial_\theta R(\theta)|_{\theta=\theta^*}}\right)^2, \tag{B.2}$$

where all terms are exactly known.

## B.3 Noisy sensing mitigated by zero-noise extrapolation

Here I compute

$$\mathrm{CMSE}[\hat{\theta}^*|\theta^*] = \mathrm{Var}_N[\hat{\theta}^*|\theta^*] + (\mathrm{Bias}_N[\hat{\theta}^*|\theta^*])^2 . \tag{B.3}$$

The variance term can be computed exactly by substituting Equation (6.32) into Equation (6.31), yielding

$$\mathrm{Var}_N[\hat{\theta}^*|\theta^*] = \sum_{j=0}^{m} \gamma_j^2 \, \frac{\Delta^2 R_{\lambda_j}(\theta^*)}{N_j(\partial_\theta R(\theta)|_{\theta=\theta^*})^2} . \tag{B.4}$$

Compared to the looser bound on the variance in Equation (6.35), I have omitted the approximation of the gradient by its Lipschitz constant, as well as the assumption of equal variances $\Delta^2 R_{\lambda_j}(\theta^*) \approx \Delta^2 R_\lambda(\theta^*)$. Nonetheless, I observed numerically that the latter approximation holds quite well in this regime, justifying its use in the main text. The values of $\gamma_j$ are chosen from those selected in the hyperparameter optimization of Section 6.1.4, and thus are the same as those used in numerical experiments. The bias term is approximated as

$$\mathrm{Bias}_N[\hat{\theta}^*|\theta^*] \approx \frac{\lambda^{m+1}}{\partial_\theta R(\theta)|_{\theta=\theta^*}} , \tag{B.5}$$

where, compared to Equation (6.30), I have approximated the response function mitigation bias as $|R_M(\theta^*) - R(\theta^*)| = \Theta(\lambda^{m+1}) \approx \lambda^{m+1}$, and not approximated the denominator.

## B.4 Inference-based noisy sensing

As outlined in Section 6.2.4, the CMSE is given as

$$\mathrm{CMSE}[\hat{\theta}^*|\theta^*] = (\mathrm{Bias}[\hat{\theta}^*|\theta^*])^2 + \underbrace{\mathbb{E}_{N_I}\left[\mathrm{Var}_{N_E}\left[\hat{\theta}^*\middle|\theta^*, f\right]\,\middle|\,\theta^*\right] + \mathrm{Var}_{N_I}\left[\mathbb{E}_{N_E}\left[\hat{\theta}^*\middle|\theta^*, f\right]\,\middle|\,\theta^*\right]}_{\mathrm{Var}[\hat{\theta}^*|\theta^*]}. \tag{B.6}$$

Following from the third line of Equation 6.42, the bias term is given as

$$\mathrm{Bias}[\hat{\theta}^*|\theta^*] \approx \frac{\mathbb{E}_{N_I}[|R_\lambda(\theta^*) - \tilde{R}_\lambda(\theta^*)|]}{|\partial_\theta R_\lambda(\theta)|_{\theta=\theta^*}|} \leq \frac{\delta}{|\partial_\theta R_\lambda(\theta)|_{\theta=\theta^*}|}, \tag{B.7}$$

where the last inequality is satisfied with high probability due to Corollary 3. I note that this is a very pessimistic estimate: Equation B.7 assumes that the worst-case bound $|R_\lambda(\theta^*) - \tilde{R}_\lambda(\theta^*)| < \delta$ is saturated everywhere, but in general the average-case inference error $|R_\lambda(\theta^*) - \tilde{R}_\lambda(\theta^*)|$ will be much smaller. Thus, I expect the analytic approximations (i.e. Figure 6.6) to overestimate the error of inference-based methods, an intuition that I numerically verify in Section 6.3. Corollary 3 demonstrates that one requires $N_k \in \Omega(\log^3(n)/\delta^2)$ shots to achieve an inference error bound of $\delta$ with high probability. In approximating this term numerically, I therefore assume that $\delta \approx \sqrt{\log^3(n)/N_k}$. The estimation variance (second term in Equation B.6) follows from the second line of Equation 6.44 as

$$\mathbb{E}_{N_I}\left[\mathrm{Var}_{N_E}\left[\hat{\theta}^*\middle|\theta^*, f\right]\,\middle|\,\theta^*\right] \approx \frac{\Delta^2 R_\lambda(\theta^*)}{N_E(\partial_\theta R_\lambda(\theta)|_{\theta=\theta^*})^2}. \tag{B.8}$$

The inferred response function variance (third term in Equation B.6) follows from the first line of Equation 6.46 as

$$\mathrm{Var}_{N_I}\left[\mathbb{E}_{N_E}\left[\hat{\theta}^*\middle|\theta^*, f\right]\,\middle|\,\theta^*\right] \leq \frac{\delta^2}{(\partial_\theta R_\lambda(\theta)|_{\theta=\theta^*})^2}, \tag{B.9}$$

which is computed in the same manner as the bias term.

## B.5 Inference-based noisy sensing mitigated by zero-noise extrapolation

As shown in Section 6.2.5, adding error mitigation to inference leads to error bounds of a similar functional form (see Equations 6.47 and 6.48). This bound is evaluated in the same manner as Section B.4, with the following changes:

- The estimation variance term is scaled by a factor of $\Lambda^2$ (see Equation 33 in our full manuscript [401], c.f. Equation (6.44)).
- The bound on inference error $\delta$ is estimated from the result of Corollary 2 in our full manuscript [401], and therefore I assume that it may be approximated as $\delta \approx \sqrt{\frac{\Lambda^3 \log^3(n)}{\max_j |\gamma_j| N_k}}$. This is a factor of $\sqrt{\Lambda^3 / \max_j |\gamma_j|}$ worse than the standard inference error reported in Corollary 3.

# References


[1] Douglas Adams. *The Restaurant at the End of the Universe*. Hitchhiker's Guide to the Galaxy. Pan Books, 1980.

[2] Alan Mathison Turing. "On computable numbers, with an application to the Entscheidungsproblem". In: *Proceedings of the London Mathematical Society* 58.345-363 (1936), p. 5.

[3] George Boole. *The mathematical analysis of logic*. 1847.

[4] Claude E Shannon. "A symbolic analysis of relay and switching circuits". In: *Electrical Engineering* 57.12 (1938), pp. 713–723.

[5] Michael Oser Rabin. "Degree of difficulty of computing a function and a partial ordering of recursive sets". In: (1960).

[6] Juris Hartmanis and Richard E Stearns. "On the computational complexity of algorithms". In: *Transactions of the American Mathematical Society* 117 (1965), pp. 285–306.

[7] Paul Benioff. "The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines". In: *Journal of statistical physics* 22 (1980), pp. 563–591. URL: https://link.springer.com/article/10.1007/BF01011339.

[8] Richard P. Feynman. "Simulating physics with computers". In: *International Journal of Theoretical Physics* 21.6 (June 1982), pp. 467–488. URL: https://doi.org/10.1007/BF02650179.

[9] Seth Lloyd. "Universal quantum simulators". In: *Science* (1996), pp. 1073–1078. URL: https://doi.org/10.1126/science.273.5278.1073.

[10] David Deutsch. "Quantum theory, the Church–Turing principle and the universal quantum computer". In: *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 400.1818 (1985), pp. 97–117. URL: https://royalsocietypublishing.org/doi/10.1098/rspa.1985.0070.

[11] David Deutsch and Richard Jozsa. "Rapid solution of problems by quantum computation". In: *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences* 439.1907 (1992), pp. 553–558.

[12] Peter W Shor. "Algorithms for quantum computation: discrete logarithms and factoring". In: *Proceedings 35th annual symposium on foundations of computer science*. Ieee. 1994, pp. 124–134. URL: https://ieeexplore.ieee.org/document/365700.

[13] Ronald L Rivest, Adi Shamir, and Leonard Adleman. "A method for obtaining digital signatures and public-key cryptosystems". In: *Communications of the ACM* 21.2 (1978), pp. 120–126. URL: https://dl.acm.org/doi/10.1145/359340.359342.

[14] John D Dixon. "Asymptotically fast factorization of integers". In: *Mathematics of computation* 36.153 (1981), pp. 255–260. URL: https://www.ams.org/journals/mcom/1981-36-153/S0025-5718-1981-0595059-1/.

[15] Lov K Grover. "A fast quantum mechanical algorithm for database search". In: *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. 1996, pp. 212–219. URL: https://doi.org/10.1145/237814.237866.

[16] A Yu Kitaev. "Quantum measurements and the Abelian stabilizer problem". In: *arXiv preprint quant-ph/9511026* (1995).

[17] Lisa Hales and Sean Hallgren. "An improved quantum Fourier transform algorithm and applications". In: *Proceedings 41st Annual Symposium on Foundations of Computer Science*. IEEE. 2000, pp. 515–525. URL: https://ieeexplore.ieee.org/document/892139.

[18] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge: Cambridge University Press, 2000.

[19] Gilles Brassard et al. "Quantum amplitude amplification and estimation". In: *Contemporary Mathematics* 305 (2002), pp. 53–74. URL: http://www.ams.org/books/conm/305/.

[20] Daniel S Abrams and Seth Lloyd. "Quantum algorithm providing exponential speed increase for finding eigenvalues and eigenvectors". In: *Physical Review Letters* 83.24 (1999), p. 5162. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.83.5162.

[21] Alán Aspuru-Guzik et al. "Simulated Quantum Computation of Molecular Energies". In: *Science* 309.5741 (2005), pp. 1704–1707. URL: https://science.sciencemag.org/content/309/5741/1704.

[22] Andrew Lucas. "Ising formulations of many NP problems". In: *Frontiers in Physics* 2 (2014), p. 5. URL: https://www.frontiersin.org/articles/10.3389/fphy.2014.00005/full.

[23] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. "A quantum approximate optimization algorithm". In: *arXiv preprint arXiv:1411.4028* (2014). URL: https://arxiv.org/abs/1411.4028.

[24] Kostas Blekos et al. "A review on quantum approximate optimization algorithm and its variants". In: *Physics Reports* 1068 (2024), pp. 1–66. URL: https://www.sciencedirect.com/science/article/abs/pii/S0370157324001078.

[25] Dominic W Berry et al. "Efficient quantum algorithms for simulating sparse Hamiltonians". In: *Communications in Mathematical Physics* 270 (2007), pp. 359–371. URL: https://link.springer.com/article/10.1007/s00220-006-0150-x.

[26] Dominic W Berry et al. "Simulating Hamiltonian dynamics with a truncated Taylor series". In: *Physical Review Letters* 114.9 (2015), p. 090502. URL: https://doi.org/10.1103/PhysRevLett.114.090502.

[27] Guang Hao Low and Isaac L Chuang. "Hamiltonian simulation by qubitization". In: *Quantum* 3 (2019), p. 163. URL: https://doi.org/10.22331/q-2019-07-12-163.

[28] Guang Hao Low and Isaac L. Chuang. "Optimal Hamiltonian Simulation by Quantum Signal Processing". In: *Phys. Rev. Lett.* 118 (1 Jan. 2017), p. 010501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.118.010501.

[29] Y. Li and S. C. Benjamin. "Efficient Variational Quantum Simulator Incorporating Active Error Minimization". In: *Phys. Rev. X* 7 (2 June 2017), p. 021050. URL: https://link.aps.org/doi/10.1103/PhysRevX.7.021050.

[30] Hirsh Kamakari et al. "Digital quantum simulation of open quantum systems using quantum imaginary–time evolution". In: *PRX Quantum* 3.1 (2022), p. 010320. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.3.010320.

[31] Aram W Harrow, Avinatan Hassidim, and Seth Lloyd. "Quantum algorithm for linear systems of equations". In: *Physical Review Letters* 103.15 (2009), p. 150502. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.103.150502.

[32] András Gilyén et al. "Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics". In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. 2019, pp. 193–204. URL: https://dl.acm.org/doi/abs/10.1145/3313276.3316366.

[33] John M. Martyn et al. "Grand Unification of Quantum Algorithms". In: *PRX Quantum* 2 (4 Dec. 2021), p. 040203. URL: https://link.aps.org/doi/10.1103/PRXQuantum.2.040203.

[34] Ashley Montanaro. "Quantum algorithms: an overview". In: *npj Quantum Information* 2.1 (2016), pp. 1–8. URL: https://www.nature.com/articles/npjqi201523.

[35] Alexander M Dalzell et al. "Quantum algorithms: A survey of applications and end-to-end complexities". In: *arXiv preprint arXiv:2310.03011* (2023). URL: https://arxiv.org/abs/2310.03011.

[36] G Massimo Palma, Kalle-Antti Suominen, and Artur Ekert. "Quantum computers and dissipation". In: *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 452.1946 (1996), pp. 567–584.

[37] Serge Haroche and Jean-Michel Raimond. "Quantum computing: dream or nightmare?" In: *Physics Today* 49.8 (1996), pp. 51–52.

[38] Christopher Monroe et al. "Future of quantum computing proves to be debatable". In: *Physics Today* 49.11 (1996), pp. 107–108.

[39] Richard W Hamming. "Error detecting and error correcting codes". In: *The Bell system technical journal* 29.2 (1950), pp. 147–160.

[40] Peter W Shor. "Scheme for reducing decoherence in quantum computer memory". In: *Physical review A* 52.4 (1995), R2493.

[41] Andrew M Steane. "Error correcting codes in quantum theory". In: *Physical Review Letters* 77.5 (1996), p. 793. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.77.793.

[42] Daniel Gottesman. *Stabilizer codes and quantum error correction*. California Institute of Technology, 1997.

[43] Daniel Gottesman and Isaac L Chuang. "Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations". In: *Nature* 402.6760 (1999), pp. 390–393. URL: https://www.nature.com/articles/46503.

[44] Sergey B Bravyi and A Yu Kitaev. "Quantum codes on a lattice with boundary". In: *arXiv preprint quant-ph/9811052* (1998). URL: https://arxiv.org/abs/quant-ph/9811052.

[45] Austin G. Fowler et al. "Surface codes: Towards practical large-scale quantum computation". In: *Physical Review A* 86 (3 Sept. 2012), p. 032324. URL: https://link.aps.org/doi/10.1103/PhysRevA.86.032324.

[46] David P DiVincenzo and Peter W Shor. "Fault-tolerant error correction with efficient quantum codes". In: *Physical review letters* 77.15 (1996), p. 3260. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.77.3260.

[47] Emanuel Knill, Raymond Laflamme, and Wojciech Zurek. "Threshold accuracy for quantum computation". In: *arXiv preprint quant-ph/9610011* (1996). URL: https://arxiv.org/abs/quant-ph/9610011.

[48] Christof Zalka. "Threshold estimate for fault tolerant quantum computation". In: *arXiv preprint quant-ph/9612028* (1996). URL: https://arxiv.org/abs/quant-ph/9612028.

[49] Peter W Shor. "Fault-tolerant quantum computation". In: *Proceedings of 37th conference on foundations of computer science*. IEEE. 1996, pp. 56–65. URL: https://ieeexplore.ieee.org/document/548464.

[50] Emanuel Knill, Raymond Laflamme, and Wojciech H Zurek. "Resilient quantum computation: error models and thresholds". In: *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 454.1969 (1998), pp. 365–384. URL: https://royalsocietypublishing.org/doi/10.1098/rspa.1998.0166.

[51] D Aharonov and M Ben-Or. "Fault-Tolerant Quantum Computation with Constant Error Rate online". In: *arXiv preprint quant-ph/9906129* (1999). URL: https://arxiv.org/abs/quant-ph/9906129.

[52] Youngseok Kim et al. "Evidence for the utility of quantum computing before fault tolerance". In: *Nature* 618.7965 (2023), pp. 500–505. URL: https://www.nature.com/articles/s41586-023-06096-3.

[53] Rajeev Acharya et al. "Quantum error correction below the surface code threshold". In: *arXiv preprint arXiv:2408.13687* (2024). URL: https://arxiv.org/abs/2408.13687.

[54] Ben W Reichardt et al. "Demonstration of quantum computation and error correction with a tesseract code". In: *arXiv preprint arXiv:2409.04628* (2024). URL: https://arxiv.org/abs/2409.04628.

[55] Dolev Bluvstein et al. "Logical quantum processor based on reconfigurable atom arrays". In: *Nature* (2023), pp. 1–3. URL: https://www.nature.com/articles/s41586-023-06927-3#article-info.

[56] Lars S Madsen et al. “Quantum computational advantage with a programmable photonic processor”. In: *Nature* 606.7912 (2022), pp. 75–81. URL: https://www.nature.com/articles/s41586-022-04725-x.

[57] Aaron J Weinstein et al. “Universal logic with encoded spin qubits in silicon”. In: *Nature* 615.7954 (2023), pp. 817–822. URL: https://www.nature.com/articles/s41586-023-05777-3.

[58] Sebastian Krinner et al. “Realizing repeated quantum error correction in a distance-three surface code”. In: *Nature* 605.7911 (2022), pp. 669–674.

[59] Youwei Zhao et al. “Realization of an error-correcting surface code with superconducting qubits”. In: *Physical Review Letters* 129.3 (2022), p. 030501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.129.030501.

[60] Rajeev Acharya et al. “Suppressing quantum errors by scaling a surface code logical qubit”. In: *Nature* 614.7949 (2023), pp. 676–681. URL: https://www.nature.com/articles/s41586-022-05434-1.

[61] Neereja Sundaresan et al. “Demonstrating multi-round subsystem quantum error correction using matching and maximum likelihood decoders”. In: *Nature Communications* 14.1 (2023), p. 2852. URL: https://www.nature.com/articles/s41467-023-38247-5.

[62] Harald Putterman et al. *Hardware-efficient quantum error correction using concatenated bosonic qubits*. 2024. URL: https://arxiv.org/abs/2409.13025.

[63] Antonio deMarti iOlius et al. “Decoding algorithms for surface codes”. In: *Quantum* 8 (2024), p. 1498. URL: https://quantum-journal.org/papers/q-2024-10-10-1498/.

[64] John Preskill. “Quantum Computing in the NISQ era and beyond”. In: *Quantum* 2 (2018), p. 79. URL: https://quantum-journal.org/papers/q-2018-08-06-79/.

[65] Frank Arute et al. “Quantum supremacy using a programmable superconducting processor”. In: *Nature* 574.7779 (Oct. 2019), pp. 505–510. URL: https://doi.org/10.1038/s41586-019-1666-5.

[66] Yong Liu et al. “Closing the "Quantum Supremacy" Gap: Achieving Real-Time Simulation of a Random Quantum Circuit Using a New Sunway Supercomputer”. In: *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. SC ’21. St. Louis, Missouri: Association for Computing Machinery, 2021. URL: https://doi.org/10.1145/3458817.3487399.

[67] Feng Pan, Keyang Chen, and Pan Zhang. “Solving the sampling problem of the sycamore quantum circuits”. In: *Physical Review Letters* 129.9 (2022), p. 090502. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.129.090502.

[68] Yulin Wu et al. “Strong quantum computational advantage using a superconducting quantum processor”. In: *Physical Review Letters* 127.18 (2021), p. 180501. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.127.180501.

[69] Qingling Zhu et al. "Quantum computational advantage via 60-qubit 24-cycle random circuit sampling". In: *Science bulletin* 67.3 (2022), pp. 240–245. URL: https://www.sciencedirect.com/science/article/pii/S2095927321006733.

[70] Joseph Tindall et al. "Efficient tensor network simulation of ibm's eagle kicked ising experiment". In: *PRX Quantum* 5.1 (2024), p. 010308. URL: https://link.aps.org/doi/10.1103/PRXQuantum.5.010308.

[71] Tomislav Begušić, Johnnie Gray, and Garnet Kin-Lic Chan. "Fast and converged classical simulations of evidence for the utility of quantum computing before fault tolerance". In: *Science Advances* 10.3 (2024). URL: https://www.science.org/doi/abs/10.1126/sciadv.adk4321.

[72] Kostyantyn Kechedzhi et al. "Effective quantum volume, fidelity and computational cost of noisy quantum processing experiments". In: *Future Generation Computer Systems* 153 (2024), pp. 431–441. URL: https://www.sciencedirect.com/science/article/pii/S0167739X23004569.

[73] Emanuele G Dalla Torre and Mor M Roses. "Dissipative mean-field theory of IBM utility experiment". In: *arXiv preprint arXiv:2308.01339* (2023). URL: https://arxiv.org/abs/2308.01339.

[74] Hai-Jun Liao et al. "Simulation of IBM's kicked Ising experiment with Projected Entangled Pair Operator". In: *arXiv preprint arXiv:2308.03082* (2023). URL: https://arxiv.org/abs/2308.03082.

[75] Tomislav Begušić, Johnnie Gray, and Garnet Kin-Lic Chan. "Fast and converged classical simulations of evidence for the utility of quantum computing before fault tolerance". In: *Science Advances* 10.3 (2024). URL: https://www.science.org/doi/abs/10.1126/sciadv.adk4321.

[76] Manuel S Rudolph et al. "Classical surrogate simulation of quantum systems with LOWESA". In: *arXiv preprint arXiv:2308.09109* (2023). URL: https://arxiv.org/abs/2308.09109.

[77] Siddhartha Patra et al. "Efficient tensor network simulation of IBM's largest quantum processors". In: *arXiv preprint arXiv:2309.15642* (2023). URL: https://doi.org/10.48550/arXiv.2309.15642.

[78] Anonymous. "Quantum Disadvantage: Or, simulating IBM's 'quantum utility' experiment with a Commodore 64". In: *Proceedings of SIGBOVIK 2024*. The Association for Computational Heresy. 2024, pp. 199–205. URL: https://sigbovik.org/2024/.

[79] M. Cerezo et al. "Variational quantum algorithms". In: *Nature Reviews Physics* 3.1 (2021), pp. 625–644. URL: https://www.nature.com/articles/s42254-021-00348-9.

[80] Amara Katabarwa et al. "Early fault-tolerant quantum computing". In: *PRX Quantum* 5.2 (2024), p. 020101. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.5.020101.

[81] Riki Toshio et al. "Practical quantum advantage on partially fault-tolerant quantum computer". In: *arXiv preprint arXiv:2408.14848* (2024). URL: https://arxiv.org/abs/2408.14848.

[82] Michael Reck et al. "Experimental realization of any discrete unitary operator". In: *Physical review letters* 73.1 (1994), p. 58. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.73.58.

[83] Adriano Barenco et al. "Elementary gates for quantum computation". In: *Physical review A* 52.5 (1995), p. 3457. URL: https://arxiv.org/abs/quant-ph/9503016.

[84] P Oscar Boykin et al. "On universal and fault-tolerant quantum computing: a novel basis and a new constructive proof of universality for shor's basis". In: *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)*. IEEE. 1999, pp. 486–494. URL: https://ieeexplore.ieee.org/document/814621.

[85] Xinlan Zhou, Debbie W Leung, and Isaac L Chuang. "Methodology for quantum logic gate construction". In: *Physical Review A* 62.5 (2000), p. 052316. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.62.052316.

[86] Sergey Bravyi and Alexei Kitaev. "Universal quantum computation with ideal Clifford gates and noisy ancillas". In: *Phys. Rev. A* 71 (2 Feb. 2005), p. 022316. URL: https://link.aps.org/doi/10.1103/PhysRevA.71.022316.

[87] Andrew D Ludlow et al. "Optical atomic clocks". In: *Reviews of Modern Physics* 87.2 (2015), pp. 637–701. URL: https://journals.aps.org/rmp/abstract/10.1103/RevModPhys.87.637.

[88] Alexander D Cronin, Jörg Schmiedmayer, and David E Pritchard. "Atom interferometers". In: *arXiv preprint arXiv:0712.3703* (2007). URL: https://arxiv.org/abs/0712.3703.

[89] Remi Geiger et al. "High-accuracy inertial measurements with cold-atom sensors". In: *AVS Quantum Science* 2.2 (2020). URL: https://pubs.aip.org/avs/aqs/article-abstract/2/2/024702/997275/High-accuracy-inertial-measurements-with-cold-atom.

[90] C. L. Degen, F. Reinhard, and P. Cappellaro. "Quantum sensing". In: *Rev. Mod. Phys.* 89.3 (2017), p. 035002. URL: https://journals.aps.org/rmp/abstract/10.1103/RevModPhys.89.035002.

[91] Matteo GA Paris. "Quantum estimation for quantum technology". In: *International Journal of Quantum Information* 7.supp01 (2009), pp. 125–137. URL: https://www.worldscientific.com/doi/abs/10.1142/S0219749909004839.

[92] Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone. "Advances in quantum metrology". In: *Nat. Photonics* 5.4 (2011), pp. 222–229. URL: https://www.nature.com/articles/nphoton.2011.35.

[93] Géza Tóth and Iagoba Apellaniz. "Quantum metrology from a quantum information science perspective". In: *Journal of Physics A: Mathematical and Theoretical* 47.42 (2014), p. 424006. URL: https://iopscience.iop.org/article/10.1088/1751-8113/47/42/424006/meta.

[94] Luca Pezzè et al. "Quantum metrology with nonclassical states of atomic ensembles". In: *Rev. Mod. Phys.* 90 (3 Sept. 2018), p. 035005. URL: https://link.aps.org/doi/10.1103/RevModPhys.90.035005.

[95] Stuart S Szigeti, Onur Hosten, and Simon A Haine. "Improving cold-atom sensors with quantum entanglement: Prospects and challenges". In: *Applied Physics Letters* 118.14 (2021). URL: https://pubs.aip.org/aip/apl/article-abstract/118/14/140501/40134/Improving-cold-atom-sensors-with-quantum.

[96] Jun Ye and Peter Zoller. "Essay: Quantum Sensing with Atomic, Molecular, and Optical Platforms for Fundamental Physics". In: *Physical Review Letters* 132.19 (2024), p. 190001. URL: https://link.aps.org/doi/10.1103/PhysRevLett.132.190001.

[97] Jiahao Huang, Min Zhuang, and Chaohong Lee. "Entanglement-enhanced quantum metrology: From standard quantum limit to Heisenberg limit". In: *Applied Physics Reviews* 11 (2024). URL: https://doi.org/10.1063/5.0204102.

[98] Johannes Jakob Meyer et al. "Quantum metrology in the finite-sample regime". In: *arXiv preprint arXiv:2307.06370* (2023). URL: https://arxiv.org/abs/2307.06370.

[99] Samuel P Nolan, Stuart S Szigeti, and Simon A Haine. "Optimal and robust quantum metrology using interaction-based readouts". In: *Physical Review Letters* 119.19 (2017), p. 193601. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.119.193601.

[100] Matthias M Müller, Stefano Gherardini, and Filippo Caruso. "Noise-robust quantum sensing via optimal multi-probe spectroscopy". In: *Scientific reports* 8.1 (2018), p. 14278. URL: https://www.nature.com/articles/s41598-018-32434-x.

[101] Lukas J Fiderer, Julien ME Fraïsse, and Daniel Braun. "Maximal quantum Fisher information for mixed states". In: *Physical Review Letters* 123.25 (2019), p. 250502. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.123.250502.

[102] Sisi Zhou, Spyridon Michalakis, and Tuvia Gefen. "Optimal protocols for quantum metrology with noisy measurements". In: *PRX Quantum* 4.4 (2023), p. 040305. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.4.040305.

[103] Tyson Jones et al. "QuEST and high performance simulation of quantum computers". In: *Scientific reports* 9.1 (2019), p. 10736. URL: https://www.nature.com/articles/s41598-019-47174-9.

[104] Tyson Jones and Simon Benjamin. "QuESTlink—Mathematica embiggened by a hardware-optimised quantum emulator". In: *Quantum Science and Technology* 5.3 (2020), p. 034012. URL: https://iopscience.iop.org/article/10.1088/2058-9565/ab8506/meta.

[105] Michael Broughton et al. "Tensorflow quantum: A software framework for quantum machine learning". In: *arXiv preprint arXiv:2003.02989* (2020). URL: https://arxiv.org/abs/2003.02989.

[106] Tyson Jones. "Simulation of, and with, first generation quantum computers". PhD thesis. University of Oxford, 2022. URL: https://ora.ox.ac.uk/objects/uuid:b666f280-249c-4218-8112-f88d348f198f.

[107] Scott Aaronson and Daniel Gottesman. "Improved simulation of stabilizer circuits". In: *Physical Review A* 70.5 (2004), p. 052328. URL: https://link.aps.org/doi/10.1103/PhysRevA.70.052328.

[108] Daniel Gottesman. "The heisenberg representation of quantum computers, talk at". In: *International Conference on Group Theoretic Methods in Physics*. Citeseer. 1998. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.252.9446.

[109] Leslie G Valiant. "Quantum computers that can be simulated classically in polynomial time". In: *Proceedings of the thirty-third annual ACM symposium on Theory of computing*. 2001, pp. 114–123. URL: https://doi.org/10.1145/380752.380785.

[110] Barbara M Terhal and David P DiVincenzo. "Classical simulation of noninteracting-fermion quantum circuits". In: *Physical Review A* 65.3 (2002), p. 032325. URL: https://link.aps.org/doi/10.1103/PhysRevA.65.032325.

[111] Richard Jozsa and Akimasa Miyake. "Matchgates and classical simulation of quantum circuits". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 464.2100 (2008), pp. 3089–3106. URL: https://royalsocietypublishing.org/doi/10.1098/rspa.2008.0189.

[112] Rolando D Somma. "Quantum computation, complexity, and many-body physics". In: *arXiv preprint quant-ph/0512209* (2005). URL: https://arxiv.org/abs/quant-ph/0512209.

[113] Rolando Somma et al. "Efficient solvability of Hamiltonians and limits on the power of some quantum computational models". In: *Physical Review Letters* 97.19 (2006), p. 190501. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.97.190501.

[114] Guifré Vidal. "Efficient Classical Simulation of Slightly Entangled Quantum Computations". In: *Phys. Rev. Lett.* 91 (14 Oct. 2003), p. 147902. URL: https://link.aps.org/doi/10.1103/PhysRevLett.91.147902.

[115] Román Orús. "A practical introduction to tensor networks: Matrix product states and projected entangled pair states". In: *Annals of Physics* 349 (2014), pp. 117–158.

[116] Jacob C Bridgeman and Christopher T Chubb. "Hand-waving and interpretive dance: an introductory course on tensor networks". In: *Journal of physics A: Mathematical and theoretical* 50.22 (2017), p. 223001. URL: https://iopscience.iop.org/article/10.1088/1751-8121/aa6dc3.

[117] Emanuel Knill et al. "Randomized benchmarking of quantum gates". In: *Physical Review A* 77.1 (2008), p. 012307. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.77.012307.

[118] Jonas Helsen et al. "Matchgate benchmarking: Scalable benchmarking of a continuous family of many-qubit gates". In: *Quantum* 6 (2022), p. 657. URL: https://quantum-journal.org/papers/q-2022-02-21-657/.

[119] Piotr Czarnik et al. "Error mitigation with Clifford quantum-circuit data". In: *Quantum* 5 (Nov. 2021), p. 592. URL: https://doi.org/10.22331/q-2021-11-26-592.

[120] Armands Strikis et al. "Learning-based quantum error mitigation". In: *PRX Quantum* 2.4 (2021), p. 040330.

[121] Frank Arute et al. "Observation of separated dynamics of charge and spin in the Fermi-Hubbard model". In: *arXiv preprint arXiv:2010.07965* (2020). URL: https://arxiv.org/abs/2010.07965.

[122] Ashley Montanaro and Stasja Stanisic. "Error mitigation by training with fermionic linear optics". In: *arXiv preprint arXiv:2102.02120* (2021). URL: https://arxiv.org/abs/2102.02120.

[123] Ewin Tang. "A quantum-inspired classical algorithm for recommendation systems". In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. 2019, pp. 217–228. URL: https://dl.acm.org/doi/10.1145/3313276.3316310.

[124] Ewin Tang. "Quantum principal component analysis only achieves an exponential speedup because of its state preparation assumptions". In: *Physical Review Letters* 127.6 (2021), p. 060503.

[125] András Gilyén, Seth Lloyd, and Ewin Tang. "Quantum-inspired low-rank stochastic regression with logarithmic dependence on the dimension". In: *arXiv preprint arXiv:1811.04909* (2018). URL: https://arxiv.org/abs/1811.04909.

[126] Nai-Hui Chia, Han-Hsuan Lin, and Chunhao Wang. "Quantum-inspired sublinear classical algorithms for solving low-rank linear systems". In: *arXiv preprint arXiv:1811.04852* (2018). URL: https://arxiv.org/abs/1811.04852.

[127] Jielun Chen, EM Stoudenmire, and Steven R White. "Quantum Fourier transform has small entanglement". In: *PRX Quantum* 4.4 (2023), p. 040318. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.4.040318.

[128] Larry Huynh et al. "Quantum-inspired machine learning: a survey". In: *arXiv preprint arXiv:2308.11269* (2023). URL: https://arxiv.org/abs/2308.11269.

[129] Hsin-Yuan Huang et al. "Power of data in quantum machine learning". In: *Nature Communications* 12.1 (2021), pp. 1–9. URL: https://www.nature.com/articles/s41467-021-22539-9.

[130] Robert M Parrish et al. "Quantum computation of electronic transitions using a variational quantum eigensolver". In: *Physical review letters* 122.23 (2019), p. 230401. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.122.230401.

[131] Sofiene Jerbi et al. "Shadows of quantum machine learning". In: *Nature Communications* 15.1 (2024), p. 5676. URL: https://www.nature.com/articles/s41467-024-49877-8.

[132] Casper Gyurik, Riccardo Molteni, and Vedran Dunjko. "Limitations of measure-first protocols in quantum machine learning". In: *arXiv preprint arXiv:2311.12618* (2023). URL: https://arxiv.org/abs/2311.12618.

[133] Andreas Elben et al. "The randomized measurement toolbox". In: *Nature Review Physics* (2022). URL: https://www.nature.com/articles/s42254-022-00535-2.

[134] M Cerezo et al. "Does provable absence of barren plateaus imply classical simulability? Or, why we need to rethink variational quantum computing". In: *arXiv preprint arXiv:2312.09121* (2023). URL: https://arxiv.org/abs/2312.09121.

[135] Martin Larocca et al. "A Review of Barren Plateaus in Variational Quantum Computing". In: *arXiv preprint arXiv:2405.00781* (2024). URL: https://arxiv.org/abs/2405.00781.

[136] Rolando D Somma et al. "Shadow Hamiltonian Simulation". In: *arXiv preprint arXiv:2407.21775* (2024). URL: https://arxiv.org/abs/2407.21775.

[137] Suguru Endo et al. "Hybrid quantum-classical algorithms and quantum error mitigation". In: *Journal of the Physical Society of Japan* 90.3 (2021), p. 032001. URL: https://doi.org/10.7566/JPSJ.90.032001.

[138] Alberto Peruzzo et al. "A variational eigenvalue solver on a photonic quantum processor". In: *Nature Communications* 5.1 (2014), pp. 1–7. URL: https://www.nature.com/articles/ncomms5213#citeas.

[139] Harper R Grimsley et al. "An adaptive variational algorithm for exact molecular simulations on a quantum computer". In: *Nature Communications* 10.1 (2019), pp. 1–9. URL: https://www.nature.com/articles/s41467-019-10988-2.

[140] Ho Lun Tang et al. "qubit-adapt-vqe: An adaptive algorithm for constructing hardware-efficient ansätze on a quantum processor". In: *PRX Quantum* 2.2 (2021), p. 020310.

[141] Arthur G Rattew et al. "A domain-agnostic, noise-resistant, hardware-efficient evolutionary variational quantum eigensolver". In: *arXiv preprint arXiv:1910.09694* (2019). URL: https://arxiv.org/abs/1910.09694.

[142] Richard Meister, Cica Gustiani, and Simon C Benjamin. "Exploring ab initio machine synthesis of quantum circuits". In: *New Journal of Physics* 25.7 (2023), p. 073018. URL: https://iopscience.iop.org/article/10.1088/1367-2630/ace077.

[143] Kosuke Mitarai et al. "Quantum circuit learning". In: *Physical Review A* 98.3 (2018), p. 032309. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.98.032309.

[144] Maria Schuld et al. "Evaluating analytic gradients on quantum hardware". In: *Physical Review A* 99.3 (2019), p. 032331. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.99.032331.

[145] Leonardo Banchi and Gavin E. Crooks. "Measuring Analytic Gradients of General Quantum Evolution with the Stochastic Parameter Shift Rule". In: *Quantum* 5 (2021), p. 386. URL: https://doi.org/10.22331/q-2021-01-25-386.

[146] Sam McArdle et al. "Variational ansatz-based quantum simulation of imaginary time evolution". In: *npj Quantum Information* 5.1 (2019), pp. 1–6. URL: https://www.nature.com/articles/s41534-019-0187-2.

[147] James Stokes et al. "Quantum natural gradient". In: *Quantum* 4 (2020), p. 269. URL: https://quantum-journal.org/papers/q-2020-05-25-269/.

[148] Bálint Koczor and Simon C Benjamin. "Quantum natural gradient generalized to noisy and nonunitary circuits". In: *Physical Review A* 106.6 (2022), p. 062416. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.106.062416.

[149] Shun-Ichi Amari. "Natural gradient works efficiently in learning". In: *Neural computation* 10.2 (1998), pp. 251–276. URL: http://cognet.mit.edu/journal/10.1162/089976698300017746.

[150] Ken M Nakanishi, Keisuke Fujii, and Synge Todo. "Sequential minimal optimization for quantum-classical hybrid algorithms". In: *Physical Review Research* 2.4 (2020), p. 043158. URL: https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.2.043158.

[151] Bálint Koczor and Simon C Benjamin. "Quantum analytic descent". In: *arXiv preprint arXiv:2008.13774* (2020). URL: https://arxiv.org/abs/2008.13774.

[152] Gregory Boyd and Bálint Koczor. "Training variational quantum circuits with CoVaR: covariance root finding with classical shadows". In: *arXiv preprint arXiv:2204.08494* (2022). URL: https://arxiv.org/abs/2204.08494.

[153] Tyson Jones et al. "Variational quantum algorithms for discovering Hamiltonian spectra". In: *Physical Review A* 99.6 (2019), p. 062304. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.99.062304.

[154] Oscar Higgott, Daochen Wang, and Stephen Brierley. "Variational quantum computation of excited states". In: *Quantum* 3 (2019), p. 156. URL: https://quantum-journal.org/papers/q-2019-07-01-156/.

[155] Ken M Nakanishi, Kosuke Mitarai, and Keisuke Fujii. "Subspace-search variational quantum eigensolver for excited states". In: *Physical Review Research* 1.3 (2019), p. 033062. URL: https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.1.033062.

[156] Lila Cadi Tazi and Alex JW Thom. "Folded Spectrum VQE: A quantum computing method for the calculation of molecular excited states". In: *Journal of Chemical Theory and Computation* 20.6 (2024), pp. 2491–2504. URL: https://pubs.acs.org/doi/10.1021/acs.jctc.3c01378.

[157] Pauline J Ollitrault et al. "Quantum equation of motion for computing molecular excitation energies on a noisy quantum processor". In: *Physical Review Research* 2.4 (2020), p. 043140. URL: https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.2.043140.

[158] Ayush Asthana et al. "Quantum self-consistent equation-of-motion method for computing molecular excitation energies, ionization potentials, and electron affinities on a quantum computer". In: *Chemical Science* 14.9 (2023), pp. 2405–2418. URL: https://pubs.rsc.org/en/content/articlelanding/2023/sc/d2sc05371c.

[159] Dan-Bo Zhang et al. "Variational quantum eigensolvers by variance minimization". In: *Chinese Physics B* 31.12 (2022), p. 120301. URL: https://iopscience.iop.org/article/10.1088/1674-1056/ac8a8d.

[160] Wooseop Hwang et al. "Preparing Ground and Excited States Using Adiabatic CoVaR". In: *arXiv preprint arXiv:2409.16194* (2024). URL: https://arxiv.org/abs/2409.16194.

[161] Bela Bauer et al. “Hybrid quantum-classical approach to correlated materials”. In: *Physical Review X* 6.3 (2016), p. 031045. URL: 10.1103/PhysRevX.6.031045.

[162] Ivan Rungger et al. “Dynamical mean field theory algorithm and experiment on quantum computers”. In: *arXiv preprint arXiv:1910.04735* (2019). URL: https://arxiv.org/abs/1910.04735.

[163] Suguru Endo, Iori Kurata, and Yuya O Nakagawa. “Calculation of the Green’s function on near-term quantum computers”. In: *Physical Review Research* 2.3 (2020), p. 033281. URL: https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.2.033281.

[164] Francois Jamet et al. “Krylov variational quantum algorithm for first principles materials simulations”. In: *arXiv preprint arXiv:2105.13298* (2021). URL: https://arxiv.org/abs/2105.13298.

[165] Francois Jamet et al. “Anderson impurity solver integrating tensor network methods with quantum computing”. In: *arXiv preprint arXiv:2304.06587* (2023).

[166] Xiaosi Xu et al. “Variational algorithms for linear algebra”. In: *Science Bulletin* 66.21 (2021), pp. 2181–2188. URL: https://www.sciencedirect.com/science/article/pii/S2095927321004631.

[167] Suguru Endo et al. “Variational quantum simulation of general processes”. In: *Physical Review Letters* 125.1 (2020), p. 010501. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.125.010501.

[168] Hsin-Yuan Huang, Kishor Bharti, and Patrick Rebentrost. “Near-term quantum algorithms for linear systems of equations with regression loss functions”. In: *New Journal of Physics* 23.11 (2021), p. 113021. URL: https://iopscience.iop.org/article/10.1088/1367-2630/ac325f.

[169] Carlos Bravo-Prieto et al. “Variational quantum linear solver”. In: *Quantum* 7 (2023), p. 1188. URL: https://quantum-journal.org/papers/q-2023-11-22-1188/.

[170] Bálint Koczor et al. “Variational-State Quantum Metrology”. In: *New Journal of Physics* (2020). URL: https://iopscience.iop.org/article/10.1088/1367-2630/ab965e.

[171] Jacob L Beckey et al. “Variational quantum algorithm for estimating the quantum Fisher information”. In: *Physical Review Research* 4.1 (2022), p. 013083. URL: https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.4.013083.

[172] Raphael Kaubruegger et al. “Quantum variational optimization of Ramsey interferometry and atomic clocks”. In: *Physical Review X* 11.4 (2021), p. 041045.

[173] Ziqi Ma et al. “Adaptive circuit learning for quantum metrology”. In: *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE. 2021, pp. 419–430. URL: https://ieeexplore.ieee.org/document/9605341.

[174] Tyler G Thurtell and Akimasa Miyake. “Optimizing one-axis twists for variational Bayesian quantum metrology”. In: *Physical Review Research* 6.2 (2024), p. 023179. URL: https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.6.023179.

[175] Ran Liu et al. "Variational Quantum Metrology with Loschmidt Echo". In: *arXiv preprint arXiv:2211.12296* (2022). URL: https://arxiv.org/abs/2211.12296.

[176] Trung Kien Le, Hung Q. Nguyen, and Le Bin Ho. "Variational quantum metrology for multiparameter estimation under dephasing noise". In: *Scientific Reports* 13 (2023), p. 17775. URL: https://doi.org/10.1038/s41598-023-44786-0.

[177] Johannes Jakob Meyer, Johannes Borregaard, and Jens Eisert. "A variational toolbox for quantum multi-parameter estimation". In: *NPJ Quantum Information* 7.1 (2021), pp. 1–5. URL: https://www.nature.com/articles/s41534-021-00425-y.

[178] Raphael Kaubruegger et al. "Optimal and Variational Multiparameter Quantum Metrology and Vector-Field Sensing". In: *PRX Quantum* 4.2 (2023), p. 020333. URL: https://link.aps.org/doi/10.1103/PRXQuantum.4.020333.

[179] Christian D Marciniak et al. "Optimal metrology with programmable quantum sensors". In: *Nature* 603.7902 (2022), pp. 604–609. URL: https://doi.org/10.1038/s41586-022-04435-4.

[180] Su Direkci et al. "Heisenberg-limited Bayesian phase estimation with low-depth digital quantum circuits". In: *arXiv preprint arXiv:2407.06006* (2024). URL: https://arxiv.org/abs/2407.06006.

[181] Juan C. Zuñiga Castro et al. *Variational quantum state preparation for quantum-enhanced metrology in noisy systems*. 2024. URL: https://arxiv.org/abs/2406.01859.

[182] Richard M Karp. *Reducibility among combinatorial problems*. Springer, 1972. URL: https://doi.org/10.1007/978-1-4684-2001-2_9.

[183] Michel X Goemans and David P Williamson. "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming". In: *Journal of the ACM (JACM)* 42.6 (1995), pp. 1115–1145. URL: https://dl.acm.org/doi/10.1145/227683.227684.

[184] Asier Ozaeta, Wim van Dam, and Peter L McMahon. "Expectation values from the single-layer quantum approximate optimization algorithm on Ising problems". In: *Quantum Science and Technology* 7.4 (2022), p. 045036. URL: https://dx.doi.org/10.1088/2058-9565/ac9013.

[185] Leo Zhou et al. "Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices". In: *Physical Review X* 10.2 (2020), p. 021067. URL: https://journals.aps.org/prx/abstract/10.1103/PhysRevX.10.021067.

[186] Gereon Koßmann et al. "Deep-Circuit QAOA". In: *arXiv preprint arXiv:2210.12406* (2022). URL: https://arxiv.org/abs/2210.12406.

[187] Xiao Yuan et al. "Theory of variational quantum simulation". In: *Quantum* 3 (2019), p. 191. URL: https://quantum-journal.org/papers/q-2019-10-07-191/.

[188] Cristina Cirstoiu et al. "Variational fast forwarding for quantum simulation beyond the coherence time". In: *npj Quantum Information* 6.1 (2020), pp. 1–10. URL: https://doi.org/10.1038/s41534-020-00302-0.

[189] Stefano Barison, Filippo Vicentini, and Giuseppe Carleo. "An efficient quantum algorithm for the time evolution of parameterized circuits". In: *Quantum* 5 (2021), p. 512. URL: https://doi.org/10.22331/q-2021-07-28-512.

[190] Sheng-Hsuan Lin et al. "Real-and imaginary-time evolution with compressed quantum circuits". In: *PRX Quantum* 2.1 (2021), p. 010342. URL: https://doi.org/10.1103/PRXQuantum.2.010342.

[191] Noah F. Berthusen et al. "Quantum dynamics simulations beyond the coherence time on noisy intermediate-scale quantum hardware by variational Trotter compression". In: *Phys. Rev. Res.* 4 (2 May 2022), p. 023097. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.4.023097.

[192] Joe Gibbs et al. "Dynamical simulation via quantum machine learning with provable generalization". In: *Physical Review Research* 6.1 (2024), p. 013241. URL: https://doi.org/10.1103/PhysRevResearch.6.013241.

[193] Matthew L Goh and Bálint Koczor. "Direct Estimation of the Density of States for Fermionic Systems". In: *arXiv preprint arXiv:2407.03414* (2024). URL: https://arxiv.org/abs/2407.03414.

[194] Benjamin Commeau et al. "Variational Hamiltonian diagonalization for dynamical quantum simulation". In: *arXiv preprint arXiv:2009.02559* (2020). URL: https://arxiv.org/abs/2009.02559.

[195] Kentaro Heya et al. "Variational quantum gate optimization". In: *arXiv preprint arXiv:1810.12745* (2018). URL: https://arxiv.org/abs/1810.12745.

[196] Sumeet Khatri et al. "Quantum-assisted quantum compiling". In: *Quantum* 3 (2019), p. 140. URL: https://quantum-journal.org/papers/q-2019-05-13-140/.

[197] Matthias C Caro et al. "Out-of-distribution generalization for learning quantum dynamics". In: *Nature Communications* 14.1 (2023), p. 3751. URL: https://www.nature.com/articles/s41467-023-39381-w.

[198] Jonathan Romero, Jonathan P Olson, and Alan Aspuru-Guzik. "Quantum autoencoders for efficient compression of quantum data". In: *Quantum Science and Technology* 2.4 (2017), p. 045001. URL: https://iopscience.iop.org/article/10.1088/2058-9565/aa8072.

[199] Alex Pepper, Nora Tischler, and Geoff J Pryde. "Experimental realization of a quantum autoencoder: The compression of qutrits via machine learning". In: *Physical review letters* 122.6 (2019), p. 060501.

[200] Chenfeng Cao and Xin Wang. "Noise-assisted quantum autoencoder". In: *Physical Review Applied* 15.5 (2021), p. 054012. URL: https://journals.aps.org/prapplied/abstract/10.1103/PhysRevApplied.15.054012.

[201] Guillaume Verdon, Michael Broughton, and Jacob Biamonte. "A quantum algorithm to train neural networks using low-depth circuits". In: *arXiv preprint arXiv:1712.05304* (2017). URL: https://arxiv.org/abs/1712.05304.

[202] Marcello Benedetti et al. "A generative modeling approach for benchmarking and training shallow quantum circuits". In: *npj Quantum Information* 5.1 (2019), p. 45. URL: https://www.nature.com/articles/s41534-019-0157-8.

[203] Yuxuan Du et al. "Expressive power of parametrized quantum circuits". In: *Physical Review Research* 2.3 (2020), p. 033125. URL: https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.2.033125.

[204] Jonathan Romero and Alán Aspuru-Guzik. "Variational quantum generators: Generative adversarial quantum machine learning for continuous distributions". In: *Advanced Quantum Technologies* 4.1 (2021), p. 2000003.

[205] Iris Cong, Soonwon Choi, and Mikhail D Lukin. "Quantum convolutional neural networks". In: *Nature Physics* 15.12 (2019), pp. 1273–1278. URL: https://www.nature.com/articles/s41567-019-0648-8.

[206] Maria Schuld et al. "Circuit-centric quantum classifiers". In: *Physical Review A* 101.3 (2020), p. 032308. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.101.032308.

[207] Matthew L Goh et al. "Lie-algebraic classical simulations for variational quantum computing". In: *arXiv preprint arXiv:2308.01432* (2023). URL: https://arxiv.org/abs/2308.01432.

[208] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. "An introduction to quantum machine learning". In: *Contemporary Physics* 56.2 (2015), pp. 172–185. URL: https://www.tandfonline.com/doi/full/10.1080/00107514.2014.964942.

[209] Jacob Biamonte et al. "Quantum machine learning". In: *Nature* 549.7671 (2017), pp. 195–202. URL: https://www.nature.com/articles/nature23474.

[210] M Cerezo et al. "Challenges and opportunities in quantum machine learning". In: *Nature Computational Science* (2022). URL: https://www.nature.com/articles/s43588-022-00311-3.

[211] Jarrod R McClean et al. "Barren plateaus in quantum neural network training landscapes". In: *Nature Communications* 9.1 (2018), pp. 1–6. URL: https://doi.org/10.1038/s41467-018-07090-4.

[212] Zoë Holmes et al. "Connecting ansatz expressibility to gradient magnitudes and barren plateaus". In: *PRX Quantum* 3 (1 Jan. 2022), p. 010313. URL: https://doi.org/10.1103/PRXQuantum.3.010313.

[213] Martin Larocca et al. "Diagnosing Barren Plateaus with Tools from Quantum Optimal Control". In: *Quantum* 6 (Sept. 2022), p. 824. URL: https://doi.org/10.22331/q-2022-09-29-824.

[214] Carlos Ortiz Marrero, Mária Kieferová, and Nathan Wiebe. "Entanglement-induced barren plateaus". In: *PRX Quantum* 2.4 (2021), p. 040316. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.2.040316.

[215] Taylor L Patti et al. "Entanglement devised barren plateau mitigation". In: *Physical Review Research* 3.3 (2021), p. 033090. URL: https://par.nsf.gov/servlets/purl/10328786.

[216] Stefan H Sack et al. "Avoiding barren plateaus using classical shadows". In: *PRX Quantum* 3.2 (2022), p. 020365. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.3.020365.

[217] M. Cerezo et al. “Cost function dependent barren plateaus in shallow parametrized quantum circuits”. In: *Nature Communications* 12.1 (2021), pp. 1–12. URL: https://www.nature.com/articles/s41467-021-21728-w.

[218] Samson Wang et al. “Noise-induced barren plateaus in variational quantum algorithms”. In: *Nature Communications* 12.1 (2021), pp. 1–11. URL: https://doi.org/10.1038/s41467-021-27045-6.

[219] Zoë Holmes et al. “Barren plateaus preclude learning scramblers”. In: *Physical Review Letters* 126.19 (2021), p. 190501. URL: https://doi.org/10.1103/PhysRevLett.126.190501.

[220] Andrew Arrasmith et al. “Equivalence of quantum barren plateaus to cost concentration and narrow gorges”. In: *Quantum Science and Technology* 7.4 (2022), p. 045015. URL: https://doi.org/10.1088/2058-9565/ac7d06.

[221] Andrew Arrasmith et al. “Effect of barren plateaus on gradient-free optimization”. In: *Quantum* 5 (2021), p. 558. URL: https://quantum-journal.org/papers/q-2021-10-05-558/.

[222] Supanut Thanasilp et al. “Exponential concentration in quantum kernel methods”. In: *Nature Communications* 15.1 (2024), p. 5200. URL: https://doi.org/10.1038/s41467-024-49287-w.

[223] Alicia B Magann et al. “From pulses to circuits and back again: A quantum optimal control perspective on variational quantum algorithms”. In: *PRX Quantum* 2.1 (2021), p. 010101. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.2.010101.

[224] Xiaozhen Ge, Re-Bing Wu, and Herschel Rabitz. “The optimization landscape of hybrid quantum–classical algorithms: From quantum control to NISQ applications”. In: *Annual Reviews in Control* (2022). URL: https://www.sciencedirect.com/science/article/pii/S1367578822000840.

[225] Martin Larocca et al. “Theory of overparametrization in quantum neural networks”. In: *Nature Computational Science* 3.6 (2023), pp. 542–551. URL: https://www.nature.com/articles/s43588-023-00467-6.

[226] Jamie Heredge et al. “Prospects of Privacy Advantage in Quantum Machine Learning”. In: *arXiv preprint arXiv:2405.08801* (2024). URL: https://arxiv.org/abs/2405.08801.

[227] Enrico Fontana et al. “Characterizing barren plateaus in quantum ansätze with the adjoint representation”. In: *Nature Communications* 15.1 (2024), p. 7171. URL: https://www.nature.com/articles/s41467-024-49910-w.

[228] Michael Ragone et al. “A Lie algebraic theory of barren plateaus for deep parameterized quantum circuits”. In: *Nature Communications* 15.1 (2024), p. 7172. URL: https://www.nature.com/articles/s41467-024-49909-3#citeas.

[229] Louis Schatzki et al. “Theoretical guarantees for permutation-equivariant quantum neural networks”. In: *npj Quantum Information* 10.1 (2024), p. 12. URL: https://www.nature.com/articles/s41534-024-00804-1.

[230] Eric R Anschuetz. “A Unified Theory of Quantum Neural Network Loss Landscapes”. In: *arXiv preprint arXiv:2408.11901* (2024). URL: https://arxiv.org/abs/2408.11901.

[231] N L Diaz et al. "Showcasing a Barren Plateau Theory Beyond the Dynamical Lie Algebra". In: *arXiv preprint arXiv:2310.11505* (2023). URL: https://arxiv.org/abs/2310.11505.

[232] Arthur Pesah et al. "Absence of Barren Plateaus in Quantum Convolutional Neural Networks". In: *Physical Review X* 11.4 (2021), p. 041011. URL: https://journals.aps.org/prx/abstract/10.1103/PhysRevX.11.041011.

[233] Léo Monbroussou et al. "Trainability and Expressivity of Hamming-Weight Preserving Quantum Circuits for Machine Learning". In: *arXiv preprint arXiv:2309.15547* (2023). URL: https://arxiv.org/abs/2309.15547.

[234] Alistair Letcher, Stefan Woerner, and Christa Zoufal. "Tight and Efficient Gradient Bounds for Parameterized Quantum Circuits". In: *arXiv preprint arXiv:2309.12681* (2023). URL: https://arxiv.org/abs/2309.12681.

[235] Manuel S Rudolph et al. "Trainability barriers and opportunities in quantum generative modeling". In: *arXiv preprint arXiv:2305.02881* (2023). URL: https://arxiv.org/abs/2305.02881.

[236] Kaining Zhang et al. "Escaping from the Barren Plateau via Gaussian Initializations in Deep Variational Quantum Circuits". In: *Advances in Neural Information Processing Systems*. 2022. URL: https://openreview.net/forum?id=jXgbJdQ2YIy.

[237] Chae-Yeun Park and Nathan Killoran. "Hamiltonian variational ansatz without barren plateaus". In: *Quantum* 8 (2024), p. 1239. URL: https://quantum-journal.org/papers/q-2024-02-01-1239/.

[238] Yabo Wang et al. "Trainability enhancement of parameterized quantum circuits via reduced-domain parameter initialization". In: *Physical Review Applied* 22.5 (2024), p. 054005. URL: https://journals.aps.org/prapplied/abstract/10.1103/PhysRevApplied.22.054005.

[239] Pablo Bermejo et al. "Quantum Convolutional Neural Networks are (Effectively) Classically Simulable". In: *arXiv preprint arXiv:2408.12739* (2024). URL: https://arxiv.org/abs/2408.12739.

[240] Jack Cunningham and Jun Zhuang. "Investigating and Mitigating Barren Plateaus in Variational Quantum Circuits: A Survey". In: *arXiv preprint arXiv:2407.17706* (2024). URL: https://arxiv.org/abs/2407.17706.

[241] Lennart Bittel and Martin Kliesch. "Training Variational Quantum Algorithms Is NP-Hard". In: *Phys. Rev. Lett.* 127 (12 Sept. 2021), p. 120502. URL: https://doi.org/10.1103/PhysRevLett.127.120502.

[242] Enrico Fontana et al. "Non-trivial symmetries in quantum landscapes and their resilience to quantum noise". In: *Quantum* 6 (2022), p. 804. URL: https://quantum-journal.org/papers/q-2022-09-15-804/.

[243] Eric R Anschuetz and Bobak T Kiani. "Quantum variational algorithms are swamped with traps". In: *Nature Communications* 13.1 (2022), p. 7760. URL: https://doi.org/10.1038/s41467-022-35364-5.

[244] Xuchen You and Xiaodi Wu. "Exponentially many local minima in quantum neural networks". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 12144–12155. URL: https://proceedings.mlr.press/v139/you21c.html.

[245] Zeyuan Allen-Zhu, Yuanzhi Li, and Yingyu Liang. "Learning and generalization in overparameterized neural networks, going beyond two layers". In: *Advances in neural information processing systems* (2019). URL: https://proceedings.neurips.cc/paper/2019/file/62dad6e273d32235ae02b7d321578ee8-Paper.pdf.

[246] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. "A convergence theory for deep learning via over-parameterization". In: *International Conference on Machine Learning*. PMLR. 2019, pp. 242–252. URL: http://proceedings.mlr.press/v97/allen-zhu19a.html.

[247] Rares-Darius Buhai et al. "Empirical study of the benefits of overparameterization in learning latent variable models". In: *International Conference on Machine Learning*. PMLR. 2020, pp. 1211–1219. URL: https://openreview.net/forum?id=rkg0_eHtDr.

[248] Alon Brutzkus et al. "SGD Learns Over-parameterized Networks that Provably Generalize on Linearly Separable Data". In: *International Conference on Learning Representations*. 2018. URL: https://openreview.net/forum?id=rJ33wwxRb.

[249] Simon Du et al. "Gradient descent finds global minima of deep neural networks". In: *International Conference on Machine Learning*. PMLR. 2019, pp. 1675–1685. URL: http://proceedings.mlr.press/v97/du19c.html.

[250] Xiao Shi and Yun Shang. "Avoiding barren plateaus via Gaussian Mixture Model". In: *arXiv preprint arXiv:2402.13501* (2024). URL: https://arxiv.org/abs/2402.13501.

[251] Guillaume Verdon et al. "Learning to learn with quantum neural networks via classical neural networks". In: *arXiv preprint arXiv:1907.05415* (2019). URL: https://arxiv.org/abs/1907.05415.

[252] Frederic Sauvage et al. "FLIP: A flexible initializer for arbitrarily-sized parametrized quantum circuits". In: *arXiv preprint arXiv:2103.08572* (2021). URL: https://arxiv.org/abs/2103.08572.

[253] Ali Rad, Alireza Seif, and Norbert M Linke. "Surviving The Barren Plateau in Variational Quantum Circuits with Bayesian Learning Initialization". In: *arXiv preprint arXiv:2203.02464* (2022). URL: https://arxiv.org/abs/2203.02464.

[254] Huan-Yu Liu et al. "Mitigating barren plateaus with transfer-learning-inspired parameter initializations". In: *New Journal of Physics* 25.1 (2023), p. 013039. URL: https://iopscience.iop.org/article/10.1088/1367-2630/acb58e.

[255] Harper R Grimsley et al. "Adaptive, problem-tailored variational quantum eigensolver mitigates rough parameter landscapes and barren plateaus". In: *npj Quantum Information* 9.1 (2023), p. 19. URL: https://www.nature.com/articles/s41534-023-00681-0.

[256] James Dborin et al. "Matrix product state pre-training for quantum machine learning". In: *Quantum Science and Technology* 7.3 (2022), p. 035014. URL: https://iopscience.iop.org/article/10.1088/2058-9565/ac7073/meta.

[257] Manuel S Rudolph et al. "Synergistic pretraining of parametrized quantum circuits via tensor networks". In: *Nature Communications* 14.1 (2023), p. 8367. URL: https://doi.org/10.1038/s41467-023-43908-6.

[258] Antonio A Mele et al. "Avoiding barren plateaus via transferability of smooth solutions in a Hamiltonian variational ansatz". In: *Physical Review A* 106.6 (2022), p. L060401. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.106.L060401.

[259] Xia Liu et al. "Mitigating Barren Plateaus of Variational Quantum Eigensolvers". In: *IEEE Transactions on Quantum Engineering* (2024), pp. 1–19. URL: https://ieeexplore.ieee.org/abstract/document/10485449.

[260] Kosuke Mitarai et al. "Quadratic clifford expansion for efficient benchmarking and initialization of variational quantum algorithms". In: *Physical Review Research* 4.3 (2022), p. 033012. URL: https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.4.033012.

[261] GS Ravi et al. "CAFQA: A classical simulation bootstrap for variational quantum algorithms". In: *arXiv preprint arXiv:2202.12924* (2022). URL: https://arxiv.org/abs/2202.12924.

[262] MH Cheng et al. "Clifford Circuit Initialisation for Variational Quantum Algorithms". In: *arXiv preprint arXiv:2207.01539* (2022). URL: https://arxiv.org/abs/2207.01539.

[263] Ricard Puig et al. "Variational quantum simulation: a case study for understanding warm starts". In: *arXiv preprint arXiv:2404.10044* (2024). URL: https://arxiv.org/abs/2404.10044.

[264] MP Da Silva et al. "Demonstration of logical qubits and repeated error correction with better-than-physical error rates". In: *arXiv preprint arXiv:2404.02280* (2024). URL: https://arxiv.org/abs/2404.02280.

[265] Zhenyu Cai et al. "Quantum error mitigation". In: *Reviews of Modern Physics* 95.4 (2023), p. 045005. URL: https://journals.aps.org/rmp/abstract/10.1103/RevModPhys.95.045005.

[266] Kristan Temme, Sergey Bravyi, and Jay M. Gambetta. "Error Mitigation for Short-Depth Quantum Circuits". In: *Physical review letters* 119 (18 Nov. 2017), p. 180509. URL: https://link.aps.org/doi/10.1103/PhysRevLett.119.180509.

[267] Tudor Giurgica-Tiron et al. "Digital zero noise extrapolation for quantum error mitigation". In: *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)* (2020), pp. 306–316.

[268] Abhinav Kandala et al. "Error mitigation extends the computational reach of a noisy quantum processor". In: *Nature* 567.7749 (Mar. 2019), pp. 491–495. URL: http://dx.doi.org/10.1038/s41586-019-1040-7.

[269] Zhenyu Cai. "A practical framework for quantum error mitigation". In: *arXiv preprint arXiv:2110.05389* (2021). URL: https://arxiv.org/abs/2110.05389.

[270] Michael Krebsbach, Björn Trauzettel, and Alessio Calzona. "Optimization of Richardson extrapolation for quantum error mitigation". In: *Physical Review A* 106.6 (2022), p. 062436. URL: https://link.aps.org/doi/10.1103/PhysRevA.106.062436.

[271] Suguru Endo, Simon C Benjamin, and Ying Li. "Practical quantum error mitigation for near-future applications". In: *Physical Review X* 8.3 (2018), p. 031027. URL: https://journals.aps.org/prx/abstract/10.1103/PhysRevX.8.031027.

[272] Ewout Van Den Berg et al. "Probabilistic error cancellation with sparse Pauli–Lindblad models on noisy quantum processors". In: *Nature Physics* (2023), pp. 1–6. URL: https://doi.org/10.1038/s41567-023-02042-2.

[273] Xavi Bonet-Monroig et al. "Low-cost error mitigation by symmetry verification". In: *Physical Review A* 98.6 (2018), p. 062339. URL: https://doi.org/10.1103/PhysRevA.98.062339.

[274] Sam McArdle, Xiao Yuan, and Simon Benjamin. "Error-Mitigated Digital Quantum Simulation". In: *Phys. Rev. Lett.* 122 (18 May 2019), p. 180501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.122.180501.

[275] William J Huggins et al. "Virtual distillation for quantum error mitigation". In: *Physical Review X* 11.4 (2021), p. 041036.

[276] Bálint Koczor. "Exponential error suppression for near-term quantum devices". In: *Physical Review X* 11.3 (2021), p. 031057.

[277] Jarrod R McClean et al. "Hybrid quantum-classical hierarchy for mitigation of decoherence and determination of excited states". In: *Physical Review A* 95.4 (2017), p. 042308. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.95.042308.

[278] Tyler Takeshita et al. "Increasing the representation accuracy of quantum simulations of chemistry without extra quantum resources". In: *Physical Review X* 10.1 (2020), p. 011004. URL: https://journals.aps.org/prx/abstract/10.1103/PhysRevX.10.011004.

[279] Enrico Fontana et al. "Classical simulations of noisy variational quantum circuits". In: *arXiv preprint arXiv:2306.05400* (2023). URL: https://arxiv.org/abs/2306.05400.

[280] Thomas Schuster et al. "A polynomial-time classical algorithm for noisy quantum circuits". In: *arXiv preprint arXiv:2407.12768* (2024). URL: https://arxiv.org/abs/2407.12768.

[281] Kaoru Yamamoto et al. "Error-mitigated quantum metrology via virtual purification". In: *Physical Review Letters* 129.25 (2022), p. 250503. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.129.250503.

[282] Hyukgun Kwon et al. "Efficacy of virtual purification-based error mitigation on quantum metrology". In: *arXiv preprint arXiv:2303.15838* (2023). URL: https://arxiv.org/abs/2303.15838.

[283] Yusuke Hama and Hirofumi Nishi. "Quantum-Error-Mitigation Circuit Groups for Noisy Quantum Metrology". In: *arXiv preprint arXiv:2303.01820* (2023). URL: https://arxiv.org/abs/2303.01820.

[284] Peng Chen and Jun Jing. *Qubit-assisted quantum metrology*. 2024. arXiv: 2404.12649 [quant-ph].

[285] John S. Van Dyke, Zackary White, and Gregory Quiroz. "Mitigating Errors in DC Magnetometry via Zero-Noise Extrapolation". In: *arXiv preprint arXiv:2402.16949* (2024). URL: https://arxiv.org/abs/2402.16949.

[286] C. Huerta Alderete et al. "Inference-Based Quantum Sensing". In: *Phys. Rev. Lett.* 129 (19 Oct. 2022), p. 190501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.129.190501.

[287] Seunghoon Lee et al. "Evaluating the evidence for exponential quantum advantage in ground-state quantum chemistry". In: *Nature Communications* 14.1 (2023), p. 1952. URL: https://www.nature.com/articles/s41467-023-37587-6.

[288] Adam D. Bookatz. "QMA-complete problems". In: *Quantum Information and Computation* 14 (2014), p. 361. URL: https://doi.org/10.26421/QIC14.5-6-1.

[289] Miroslav Dobšíček et al. "Arbitrary accuracy iterative quantum phase estimation algorithm using a single ancillary qubit: A two-qubit benchmark". In: *Physical Review A—Atomic, Molecular, and Optical Physics* 76.3 (2007), p. 030306. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.76.030306.

[290] Krysta M Svore, Matthew B Hastings, and Michael Freedman. "Faster phase estimation". In: *arXiv preprint arXiv:1304.0741* (2013). URL: https://arxiv.org/abs/1304.0741.

[291] Nathan Wiebe and Chris Granade. "Efficient Bayesian phase estimation". In: *Physical review letters* 117.1 (2016), p. 010503. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.117.010503.

[292] Cassandra Granade and Nathan Wiebe. "Using random walks for iterative phase estimation". In: *arXiv preprint arXiv:2208.04526* (2022). URL: https://arxiv.org/abs/2208.04526.

[293] Alessandro Lumino et al. "Experimental phase estimation enhanced by machine learning". In: *Physical Review Applied* 10.4 (2018), p. 044033. URL: https://journals.aps.org/prapplied/abstract/10.1103/PhysRevApplied.10.044033.

[294] Ian D Kivlichan, Christopher E Granade, and Nathan Wiebe. "Phase estimation with randomized Hamiltonians". In: *arXiv preprint arXiv:1907.10070* (2019). URL: https://arxiv.org/abs/1907.10070.

[295] Rolando Somma et al. "Simulating physical phenomena by quantum networks". In: *Physical Review A* 65.4 (2002), p. 042323. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.65.042323.

[296] Laura Clinton et al. "Phase estimation of local Hamiltonians on NISQ hardware". In: *New Journal of Physics* 25.3 (2023), p. 033027. URL: https://iopscience.iop.org/article/10.1088/1367-2630/acc26d.

[297] Guoming Wang et al. "Quantum algorithm for ground state energy estimation using circuit depth with exponentially improved dependence on precision". In: *Quantum* 7 (2023), p. 1167. URL: https://quantum-journal.org/papers/q-2023-11-06-1167/.

[298] Lin Lin and Yu Tong. "Heisenberg-limited ground-state energy estimation for early fault-tolerant quantum computers". In: *PRX Quantum* 3.1 (2022), p. 010318. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.3.010318.

[299] Zhiyan Ding and Lin Lin. "Simultaneous estimation of multiple eigenvalues with short-depth quantum circuit on early fault-tolerant quantum computers". In: *Quantum* 7 (2023), p. 1136. URL: https://quantum-journal.org/papers/q-2023-10-11-1136/.

[300] Alexander Weiße et al. "The kernel polynomial method". In: *Reviews of modern physics* 78.1 (2006), p. 275. URL: https://journals.aps.org/rmp/abstract/10.1103/RevModPhys.78.275.

[301] Rolando Elio Belardinelli and Victor Daniel Pereyra. "Fast algorithm to calculate density of states". In: *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics* 75.4 (2007), p. 046701. URL: https://journals.aps.org/pre/abstract/10.1103/PhysRevE.75.046701.

[302] M Scott Shell, Pablo G Debenedetti, and Athanassios Z Panagiotopoulos. "An improved Monte Carlo method for direct calculation of the density of states". In: *The Journal of chemical physics* 119.18 (2003), pp. 9406–9411.

[303] Shufeng Kong et al. "Density of states prediction for materials discovery via contrastive learning from probabilistic embeddings". In: *Nature communications* 13.1 (2022), p. 949.

[304] Brielin Brown, Steven T Flammia, and Norbert Schuch. "Computational difficulty of computing the density of states". In: *Physical review letters* 107.4 (2011), p. 040501. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.107.040501.

[305] Patrick Rall. "Quantum algorithms for estimating physical quantities using block encodings". In: *Physical Review A* 102.2 (2020), p. 022408. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.102.022408.

[306] Sirui Lu, Mari Carmen Banuls, and J Ignacio Cirac. "Algorithms for quantum simulation at finite energies". In: *PRX Quantum* 2.2 (2021), p. 020321. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.2.020321.

[307] Alexander Schuckert et al. "Probing finite-temperature observables in quantum simulators of spin systems with short-time dynamics". In: *Physical Review B* 107.14 (2023), p. L140410. URL: https://journals.aps.org/prb/abstract/10.1103/PhysRevB.107.L140410.

[308] Khaldoon Ghanem, Alexander Schuckert, and Henrik Dreyer. "Robust Extraction of Thermal Observables from State Sampling and Real-Time Dynamics on Quantum Computers". In: *Quantum* 7 (2023), p. 1163. URL: https://quantum-journal.org/papers/q-2023-11-03-1163/.

[309] Alessandro Summer et al. "Calculating the many-body density of states on a digital quantum computer". In: *Physical Review Research* 6.1 (2024), p. 013106. URL: https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.6.013106.

[310] Henrik Bruus and Karsten Flensberg. *Many-body quantum theory in condensed matter physics: an introduction*. Oxford University Press, 2004.

[311] G Giuliani and G Vignale. *Quantum Theory of the Electron Liquid Cambridge, UK: Cambridge Univ*. 2005.

[312] Guillaume Verdon et al. "Quantum Hamiltonian-based models and the variational quantum thermalizer algorithm". In: *arXiv preprint arXiv:1910.02071* (2019). URL: https://arxiv.org/abs/1910.02071.

[313] Jingxiang Wu and Timothy H Hsieh. "Variational thermal quantum simulation via thermofield double states". In: *Physical review letters* 123.22 (2019), p. 220502. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.123.220502.

[314] Jonathan Foldager, Arthur Pesah, and Lars Kai Hansen. "Noise-assisted variational quantum thermalization". In: *Scientific reports* 12.1 (2022), p. 3862. URL: https://www.nature.com/articles/s41598-022-07296-z.

[315] JKL MacDonald. "On the modified Ritz variation method". In: *Physical Review* 46.9 (1934), p. 828.

[316] DJ Rowe. "Equations-of-motion method and the extended shell model". In: *Reviews of Modern Physics* 40.1 (1968), p. 153. URL: https://journals.aps.org/rmp/abstract/10.1103/RevModPhys.40.153.

[317] Ksenija Glusac. "What has light ever done for chemistry?" In: *Nature Chemistry* 8.8 (2016), pp. 734–735. URL: https://www.nature.com/articles/nchem.2582.

[318] Antoine Georges et al. "Dynamical mean-field theory of strongly correlated fermion systems and the limit of infinite dimensions". In: *Reviews of Modern Physics* 68.1 (1996), p. 13. URL: https://journals.aps.org/rmp/abstract/10.1103/RevModPhys.68.13.

[319] Atanu Rajak et al. "Quantum annealing: An overview". In: *Philosophical Transactions of the Royal Society A* 381.2241 (2023), p. 20210417. URL: https://royalsocietypublishing.org/doi/10.1098/rsta.2021.0417.

[320] Antonio E Russo et al. "Evaluating energy differences on a quantum computer with robust phase estimation". In: *Physical review letters* 126.21 (2021), p. 210501. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.126.210501.

[321] Kh P Gnatenko, HP Laba, and VM Tkachuk. "Energy levels estimation on a quantum computer by evolution of a physical quantity". In: *Physics Letters A* 424 (2022), p. 127843. URL: https://www.sciencedirect.com/science/article/abs/pii/S0375960121007076.

[322] Yuichiro Matsuzaki et al. "Direct estimation of the energy gap between the ground state and excited state with quantum annealing". In: *Japanese Journal of Applied Physics* 60.SB (2021), SBBI02. URL: https://iopscience.iop.org/article/10.35848/1347-4065/abdf20.

[323] Ilia Zintchenko and Nathan Wiebe. "Randomized gap and amplitude estimation". In: *Physical Review A* 93.6 (2016), p. 062306. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.93.062306.

[324] Hsin-Yuan Huang, Richard Kueng, and John Preskill. "Predicting many properties of a quantum system from very few measurements". In: *Nature Physics* 16.10 (2020), pp. 1050–1057. URL: https://www.nature.com/articles/s41567-020-0932-7.

[325] Scott Aaronson. "Shadow tomography of quantum states". In: *SIAM Journal on Computing* 49.5 (2019), STOC18–368. URL: https://dl.acm.org/doi/abs/10.1145/3188745.3188802.

[326] Ewout Van Den Berg. "A simple method for sampling random clifford operators". In: *2021 ieee international conference on quantum computing and engineering (qce)*. IEEE. 2021, pp. 54–59. URL: https://www.computer.org/csdl/proceedings-article/qce/2021/169100a054/1yEZdUCHkNa.

[327] Maxwell West et al. "Real classical shadows". In: *arXiv preprint arXiv:2410.23481* (2024). URL: https://arxiv.org/abs/2410.23481.

[328] Christian Bertoni et al. "Shallow shadows: Expectation estimation using low-depth random Clifford circuits". In: *Physical Review Letters* 133.2 (2024), p. 020602. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.133.020602.

[329] Andrew Zhao, Nicholas C Rubin, and Akimasa Miyake. "Fermionic partial tomography via classical shadows". In: *Physical Review Letters* 127.11 (2021), p. 110504. URL: https://link.aps.org/doi/10.1103/PhysRevLett.127.110504.

[330] Kianna Wan et al. "Matchgate shadows for fermionic quantum simulation". In: *Communications in Mathematical Physics* 404 (2023), p. 629. URL: https://link.springer.com/article/10.1007/s00220-023-04844-0.

[331] Valentin Heyraud, Héloise Chomet, and Jules Tilly. "Unified Framework for Matchgate Classical Shadows". In: *arXiv preprint arXiv:2409.03836* (2024). URL: https://arxiv.org/abs/2409.03836.

[332] Ahmed A Akhtar, Hong-Ye Hu, and Yi-Zhuang You. "Scalable and flexible classical shadow tomography with tensor networks". In: *Quantum* 7 (2023), p. 1026. URL: https://quantum-journal.org/papers/q-2023-06-01-1026/.

[333] J.R.R. Tolkien. *The Fellowship of the Ring*. London: George Allen & Unwin, 1954.

[334] Hans Hon Sang Chan et al. "Algorithmic shadow spectroscopy". In: *PRX Quantum* 6.1 (2025), p. 010352. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.6.010352.

[335] George EP Box and David A Pierce. "Distribution of residual autocorrelations in autoregressive-integrated moving average time series models". In: *Journal of the American Statistical Association* 65.332 (1970), pp. 1509–1526.

[336] Greta M Ljung and George EP Box. "On a measure of lack of fit in time series models". In: *Biometrika* 65.2 (1978), pp. 297–303.

[337] Virginia Vassilevska Williams et al. "New bounds for matrix multiplication: from alpha to omega". In: *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM. 2024, pp. 3792–3835. URL: https://epubs.siam.org/doi/abs/10.1137/1.9781611977912.134.

[338] James W Cooley and John W Tukey. "An algorithm for the machine calculation of complex Fourier series". In: *Mathematics of computation* 19.90 (1965), pp. 297–301. URL: https://www.jstor.org/stable/2003354.

[339] Bálint Koczor. "The dominant eigenvector of a noisy quantum state". In: *New Journal of Physics* 23.12 (2021), p. 123047.

[340] Michael R Geller and Zhongyuan Zhou. "Efficient error models for fault-tolerant architectures and the Pauli twirling approximation". In: *Physical Review A—Atomic, Molecular, and Optical Physics* 88.1 (2013), p. 012314. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.88.012314.

[341] Zhenyu Cai and Simon C Benjamin. "Constructing smaller pauli twirling sets for arbitrary error channels". In: *Scientific reports* 9.1 (2019), p. 11281. URL: https://www.nature.com/articles/s41598-019-46722-7.

[342] David J Luitz, Nicolas Laflorencie, and Fabien Alet. "Many-body localization edge in the random-field Heisenberg chain". In: *Physical Review B* 91.8 (2015), p. 081103.

[343] Rahul Nandkishore and David A Huse. "Many-body localization and thermalization in quantum statistical mechanics". In: *Annu. Rev. Condens. Matter Phys.* 6.1 (2015), pp. 15–38.

[344] Andrew M Childs et al. "Toward the first quantum simulation with quantum speedup". In: *Proceedings of the National Academy of Sciences* 115.38 (2018), pp. 9456–9461. URL: https://www.pnas.org/doi/abs/10.1073/pnas.1801723115.

[345] Jutho Haegeman et al. "Elementary excitations in gapped quantum spin systems". In: *Physical review letters* 111.8 (2013), p. 080401. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.111.080401.

[346] Dominik Hangleiter, Jacques Carolan, and Karim PY Thébault. *Analogue Quantum Simulation: A New Instrument for Scientific Understanding*. Tech. rep. Springer, 2022.

[347] Christian Kokail et al. "Entanglement Hamiltonian tomography in quantum simulation". In: *Nature Physics* 17.8 (2021), pp. 936–942. URL: https://www.nature.com/articles/s41567-021-01260-w.

[348] Ian D Kivlichan et al. "Quantum simulation of electronic structure with linear depth and connectivity". In: *Physical review letters* 120.11 (2018), p. 110501. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.120.110501.

[349] Frank Arute et al. "Hartree-Fock on a superconducting qubit quantum computer". In: *Science* 369.6507 (2020), pp. 1084–1089. URL: https://science.sciencemag.org/content/369/6507/1084.

[350] D.L. Goodstein. *States of Matter*. Prentice-Hall, 1975.

[351] Matthew L. Goh. "Feedback control of atomic Fermi gases". Honours Thesis. The Australian National University, 2019. URL: https://openresearch-repository.anu.edu.au/items/87bda595-833b-4dfb-8530-b58d26ae77ca.

[352] Kaelyn J Ferris et al. "Exploiting Maximally Mixed States for Spectral Estimation by Time Evolution". In: *arXiv preprint arXiv:2312.00687* (2023).

[353] Emanuel Knill and Raymond Laflamme. "Power of one bit of quantum information". In: *Physical Review Letters* 81.25 (1998), p. 5672. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.81.5672.

[354] Andreas Bärtschi and Stephan Eidenbenz. "Deterministic preparation of Dicke states". In: *International Symposium on Fundamentals of Computation Theory*. Springer. 2019, pp. 126–139. URL: https://link.springer.com/chapter/10.1007/978-3-030-25027-0_9.

[355] Alan V Oppenheim. *Discrete-time signal processing*. Prentice Hall, 1999.

[356] Fredric J Harris. "On the use of windows for harmonic analysis with the discrete Fourier transform". In: *Proceedings of the IEEE* 66.1 (1978), pp. 51–83.

[357] Vadym Kliuchnikov et al. "Shorter quantum circuits via single-qubit gate approximation". In: *Quantum* 7 (2023), p. 1208. URL: https://quantum-journal.org/papers/q-2023-12-18-1208/.

[358] Bálint Koczor. "Sparse Probabilistic Synthesis of Quantum Operations". In: *arXiv preprint arXiv:2402.15550* (2024). URL: https://arxiv.org/abs/2402.15550.

[359] Zhenyu Cai. "Resource estimation for quantum variational simulations of the Hubbard model". In: *Physical Review Applied* 14.1 (2020), p. 014059. URL: https://journals.aps.org/prapplied/abstract/10.1103/PhysRevApplied.14.014059.

[360] Frank Verstraete, J Ignacio Cirac, and José I Latorre. "Quantum circuits for strongly correlated quantum systems". In: *Physical Review A* 79.3 (2009), p. 032316. URL: https://doi.org/10.1103/PhysRevA.79.032316.

[361] Jonathan Foldager and Bálint Koczor. "Can shallow quantum circuits scramble local noise into global white noise?" In: *Journal of Physics A: Mathematical and Theoretical* 57.1 (2023), p. 015306. URL: https://iopscience.iop.org/article/10.1088/1751-8121/ad0ac7.

[362] Yilun Yang et al. "Phase-sensitive quantum measurement without controlled operations". In: *Physical Review Letters* 132.22 (2024), p. 220601. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.132.220601.

[363] Laura Clinton et al. "Quantum Phase Estimation without Controlled Unitaries". In: *arXiv preprint arXiv:2410.21517* (2024). URL: https://arxiv.org/abs/2410.21517.

[364] Sacha Baron Cohen. *Borat: Cultural Learnings of America for Make Benefit Glorious Nation of Kazakhstan*. 20th Century Fox, 2006.

[365] Ruvi Lecamwasam, Tatiana Iakovleva, and Jason Twamley. "Quantum metrology with linear Lie algebra parameterisations". In: *arXiv preprint arXiv:2311.12446* (2023). URL: https://arxiv.org/abs/2311.12446.

[366] Ville Bergholm et al. "Pennylane: Automatic differentiation of hybrid quantum-classical computations". In: *arXiv preprint arXiv:1811.04968* (2018). URL: https://arxiv.org/abs/1811.04968.

[367] Brian C Hall. *Lie groups, Lie algebras, and representations*. Springer, 2013.

[368] Alexander Kirillov Jr. *An introduction to Lie groups and Lie algebras*. 113. Cambridge University Press, 2008.

[369] D. D'Alessandro. *Introduction to Quantum Control and Dynamics*. Chapman & Hall/CRC Applied Mathematics & Nonlinear Science. Taylor & Francis, 2007. URL: https://books.google.sm/books?id=HbMYmAEACAAJ.

[370] Robert Zeier and Thomas Schulte-Herbrüggen. "Symmetry principles in quantum systems theory". In: *Journal of mathematical physics* 52.11 (2011), p. 113510. URL: https://aip.scitation.org/doi/pdf/10.1063/1.3657939.

[371] Gerard Aguilar et al. *Full classification of Pauli Lie algebras*. 2024. arXiv: 2408.00081. URL: https://arxiv.org/abs/2408.00081.

[372] Tyson Jones and Julien Gacon. "Efficient calculation of gradients in classical simulations of variational quantum algorithms". In: *arXiv preprint arXiv:2009.02823* (2020). URL: https://arxiv.org/abs/2009.02823.

[373] Eric R Anschuetz et al. "Efficient classical algorithms for simulating symmetric quantum systems". In: *Quantum* 7 (2023), p. 1189. URL: https://quantum-journal.org/papers/q-2023-11-28-1189/.

[374] Seth Lloyd. "Almost any quantum logic gate is universal". In: *Physical Review Letters* 75.2 (1995), p. 346. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.75.346.

[375] Sujay Kazi, Martin Larocca, and M Cerezo. "On the universality of $S_n$-equivariant $k$-body gates". In: *New Journal of Physics* 26.5 (May 2024), p. 053030. URL: https://dx.doi.org/10.1088/1367-2630/ad4819.

[376] Xavier Bonet-Monroig, Ryan Babbush, and Thomas E O'Brien. "Nearly optimal measurement scheduling for partial tomography of quantum states". In: *Physical Review X* 10.3 (2020), p. 031064. URL: https://journals.aps.org/prx/abstract/10.1103/PhysRevX.10.031064.

[377] Efekan Kökcü et al. "Fixed depth Hamiltonian simulation via Cartan decomposition". In: *Physical Review Letters* 129.7 (2022), p. 070501. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.129.070501.

[378] Sujay Kazi et al. "Analyzing the quantum approximate optimization algorithm: ansätze, symmetries, and Lie algebras". In: *arXiv preprint arXiv:2410.05187* (2024). URL: https://arxiv.org/abs/2410.05187.

[379] Matthew L. Goh. *g-sim*. https://github.com/gohmat/g-sim. 2023.

[380] Martin Hebenstreit et al. "All pure fermionic non-gaussian states are magic states for matchgate computations". In: *Physical review letters* 123.8 (2019), p. 080503. URL: https://doi.org/10.1103/PhysRevLett.123.080503.

[381] Oliver Reardon-Smith. "The fermionic linear optical extent is multiplicative for 4 qubit parity eigenstates". In: *arXiv preprint arXiv:2407.20934* (2024). URL: https://arxiv.org/abs/2407.20934.

[382] Efekan Kökcü et al. "Fixed depth Hamiltonian simulation via Cartan decomposition". In: *Physical Review Letters* 129.7 (2022), p. 070501. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.129.070501.

[383] Gabriel Matos et al. "Characterization of variational quantum algorithms using free fermions". In: *Quantum* 7 (2023), p. 966. URL: https://quantum-journal.org/papers/q-2023-03-30-966/.

[384] Dave Wecker, Matthew B. Hastings, and Matthias Troyer. "Progress towards practical quantum variational algorithms". In: *Physical Review A* 92 (4 Oct. 2015), p. 042303. URL: https://link.aps.org/doi/10.1103/PhysRevA.92.042303.

[385] Roeland Wiersema et al. "Exploring entanglement and optimization within the Hamiltonian variational ansatz". In: *PRX Quantum* 1.2 (2020), p. 020319. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.1.020319.

[386] Wen Wei Ho and Timothy H. Hsieh. "Efficient variational simulation of non-trivial quantum states". In: *SciPost Phys.* 6 (3 2019), p. 29. URL: https://scipost.org/10.21468/SciPostPhys.6.3.029.

[387] Chris Cade et al. "Strategies for solving the Fermi-Hubbard model on near-term quantum computers". In: *Physical Review B* 102.23 (2020), p. 235122. URL: https://journals.aps.org/prb/abstract/10.1103/PhysRevB.102.235122.

[388] Michal Oszmaniec et al. "Fermion sampling: a robust quantum computational advantage scheme using fermionic linear optics and magic input states". In: *PRX Quantum* 3.2 (2022), p. 020328. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.3.020328.

[389] Kunal Sharma et al. "Noise resilience of variational quantum compiling". In: *New Journal of Physics* 22.4 (2020), p. 043006. URL: https://iopscience.iop.org/article/10.1088/1367-2630/ab784c.

[390] Philip W Anderson. "Absence of diffusion in certain random lattices". In: *Physical review* 109.5 (1958), p. 1492. URL: https://journals.aps.org/pr/abstract/10.1103/PhysRev.109.1492.

[391] Valmir Bucaj. "On the Kunz-Souillard approach to localization for the discrete one dimensional generalized Anderson model". In: *arXiv preprint arXiv:1608.01379* (2016). URL: https://arxiv.org/abs/1608.01379.

[392] Paolo Zanardi and Mario Rasetti. "Noiseless quantum codes". In: *Physical Review Letters* 79.17 (1997), p. 3306. URL: https://link.aps.org/doi/10.1103/PhysRevLett.79.3306.

[393] Bryan Eastin and Emanuel Knill. "Restrictions on transversal encoded quantum gate sets". In: *Physical review letters* 102.11 (2009), p. 110502. URL: https://link.aps.org/doi/10.1103/PhysRevLett.102.110502.

[394] Navin Khaneja, Roger Brockett, and Steffen J Glaser. "Time optimal control in spin systems". In: *Physical Review A* 63.3 (2001), p. 032308. URL: https://link.aps.org/doi/10.1103/PhysRevA.63.032308.

[395] Zoltán Zimborás et al. "Symmetry criteria for quantum simulability of effective interactions". In: *Physical Review A* 92.4 (2015), p. 042309. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.92.042309.

[396] Iman Marvian. "Restrictions on realizable unitary operations imposed by symmetry and locality". In: *Nature Physics* 18.3 (2022), pp. 283–289. URL: https://www.nature.com/articles/s41567-021-01464-0.

[397] Tzu-Ching Yen and Artur F Izmaylov. "Cartan subalgebra approach to efficient measurements of quantum observables". In: *PRX Quantum* 2.4 (2021), p. 040320. URL: https://link.aps.org/doi/10.1103/PRXQuantum.2.040320.

[398] Thomas Steckmann et al. "Simulating the Mott transition on a noisy digital quantum computer via Cartan-based fast-forwarding circuits". In: *arXiv preprint arXiv:2112.05688* (2021). URL: https://arxiv.org/abs/2112.05688.

[399] Diego García-Martín, Martin Larocca, and M. Cerezo. "Effects of noise on the overparametrization of quantum neural networks". In: *Phys. Rev. Res.* 6 (1 Mar. 2024), p. 013295. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.6.013295.

[400] Don Siegel. *Dirty Harry*. Warner Bros., 1971.

[401] Aroosa Ijaz et al. "More buck-per-shot: Why learning trumps mitigation in noisy quantum sensing". In: *Materials Today Quantum* (2025), p. 100042. URL: https://www.sciencedirect.com/science/article/pii/S2950257825000204.

[402] BM Escher, Ruynet Lima de Matos Filho, and Luiz Davidovich. "General framework for estimating the ultimate precision limit in noisy quantum-enhanced metrology". In: *Nature Physics* 7.5 (2011), pp. 406–411. URL: https://doi.org/10.1038/nphys1958.

[403] Rafał Demkowicz-Dobrzański, Jan Kołodyński, and Mădălin Guţă. "The elusive Heisenberg limit in quantum-enhanced metrology". In: *Nature communications* 3.1 (2012), p. 1063. URL: https://www.nature.com/articles/ncomms2067.

[404] Simon A Haine et al. "Heisenberg-limited metrology with information recycling". In: *Physical Review A* 91.4 (2015), p. 041802. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.91.041802.

[405] Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone. "Quantum metrology". In: *Physical Review Letters* 96.1 (2006), p. 010401. URL: https://doi.org/10.1103/PhysRevLett.96.010401.

[406] Daniel Stilck França and Raul Garcia-Patron. "Limitations of optimization algorithms on noisy quantum devices". In: *Nature Physics* 17.11 (2021), pp. 1221–1227.

[407] Alexander Müller-Hermes, Daniel Stilck França, and Michael M Wolf. "Relative entropy convergence for depolarizing channels". In: *Journal of Mathematical Physics* 57.2 (2016), p. 022202. URL: https://aip.scitation.org/doi/10.1063/1.4939560.

[408] Kendall Atkinson. *An introduction to numerical analysis*. John Wiley & Sons, 1991.

[409] PG Hoel and A Levine. "Optimal spacing and weighting in polynomial prediction". In: *The Annals of Mathematical Statistics* (1964), pp. 1553–1560. URL: https://projecteuclid.org/journals/annals-of-mathematical-statistics/volume-35/issue-4/Optimal-Spacing-and-Weighting-in-Polynomial-Prediction/10.1214/aoms/1177700379.full.

[410] Jack C Saywell et al. "Enhancing the sensitivity of atom-interferometric inertial sensors using robust control". In: *Nature Communications* 14.1 (2023), p. 7626. URL: https://www.nature.com/articles/s41467-023-43374-0.

[411] Vicent J. Borràs et al. "A quantum sensing metrology for magnetic memories". In: *npj Spintronics* 2.1 (2024). URL: https://doi.org/10.1038/s44306-024-00016-5.

[412] Bálint Koczor, John JL Morton, and Simon C Benjamin. "Probabilistic interpolation of quantum rotation angles". In: *Physical Review Letters* 132.13 (2024), p. 130602. URL: https://arxiv.org/abs/2305.19881.

[413] Jesús Rubio and Jacob Dunningham. "Quantum metrology in the presence of limited data". In: *New Journal of Physics* 21.4 (Apr. 2019), p. 043037. URL: https://dx.doi.org/10.1088/1367-2630/ab098b.

[414] Dunham Jackson. "On the accuracy of trigonometric interpolation". In: *Transactions of the American Mathematical Society* 14.4 (1913), pp. 453–461. URL: https://www.ams.org/journals/tran/1913-014-04/S0002-9947-1913-1500957-1/.

[415] Charles W Champ and Andrew V Sills. "The Generalized Law of Total Covariance". In: *arXiv preprint arXiv:2205.14525* (2022). URL: https://arxiv.org/abs/2205.14525.

[416] Rina Foygel Barber. "Hoeffding and Bernstein inequalities for weighted sums of exchangeable random variables". In: *arXiv preprint arXiv:2404.06457* (2024). URL: https://arxiv.org/abs/2404.06457.

[417] J.R.R. Tolkien. *The Return of the King*. London: George Allen & Unwin, 1955.