Applications of Gaussian boson sampling in graph theory

By

NAOMI ROSE SOLOMONS



School of Physics
UNIVERSITY OF BRISTOL

A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of DOCTOR OF PHILOSOPHY in the Faculty of Science.

FEBRUARY 2024

Supervisor: Anthony Laing



Word count: 42 232

Abstract

Quantum mechanics is a formalism for describing nature at a fundamental level, which becomes more relevant for particular systems at a very small scale. These can be used as an architecture for information technology, with unique properties that can be exploited to perform certain tasks with computational power beyond what was originally conceived to be mathematically possible. However, practical realisations that demonstrate this benefit have been difficult to construct. In the long term, it is believed that quantum computers will run digital quantum algorithms with error correction to solve classically intractable problems. In the immediate term, there is a search for experiments that show quantum advantage, in solving specific problems more efficiently than classical supercomputers.

Quantum optics, which encodes information in the states of light, is privileged to have complexity-theoretic evidence supporting a scheme for quantum advantage that is native to the platform – boson sampling, and its variant Gaussian boson sampling – providing a natural choice for early proof-of-concept quantum computing experiments. Hence we are now in an exciting era, with impressive experimental achievements which continually surpass new milestones for quantum computing capabilities, but which are countered by theoretical challenges demonstrating the existing power of classical computers.

Beyond the objective of quantum advantage, the known applications of Gaussian boson sampling are limited. A useful framework for developing and analysing potential uses arises from its description in the language of graph theory, which is the basis for many well-known mathematically hard problems, including dense subgraph finding, a problem which represents difficult tasks in various fields, from finance to drug discovery.

In this thesis, we consider evidence for Gaussian boson sampling showing a performance or time advantage for certain applications. We use numerical simulations of realistic sources of error in Gaussian boson sampling, when applied to dense subgraph finding, which disagree with the possibility of quantum advantage for this use case; we show how a similar performance in sampling high-density subgraphs can be achieved by classical sampling techniques, and assess the performance of various sampling schemes for graph problems; we provide evidence that sampling from displaced Gaussian states also cannot be efficiently classically simulated (that is, in polynomial time) in the case of sufficiently low displacement, and we use the connection between the loop hafnian and the matching polynomial to identify regimes in which simulating displaced Gaussian boson sampling is classically efficient. Although these techniques do not show any additional uses of Gaussian boson sampling, they provide important insights into its value as a computational tool and improve our understanding of the features expected in further applications.

Acknowledgements

No one who started a PhD in 2019 had the experience they expected, but despite everything these were truly the best years of my life[1], mostly because of the people that I have been surrounded by. I have many people to thank, and I can only apologise for any omissions, of which I'm sure there will be many.

Firstly, thank you to Anthony, for your ambition and motivation, and the Laing group, for creating a great atmosphere which I was very glad to join halfway through my PhD.

Thank you to Dara for your supervision, and for its continuation long after you had any obligation to do so. Thank you for your good humour and patience. I am inspired by your motivation to find the answers to scientific questions simply because they're interesting!

An enormous thank you to Oli. Without you, this thesis would be much shorter, and much harder to write. Thank you for teaching me some small amount of your huge knowledge of coding and quantum optics, and doing so in a cheerful and fun way. Thank you for your sage (and patient) wisdom, such as 'computers don't run programs when they're turned off'. Thank you as well to the rest of Club++ (John, Lana, and Alex) for the good (virtual) company and for making the early days of learning C++ less onerous.

Thank you to the collaborators who I worked with throughout this PhD. In particular, thank you to Siddarth for encouraging me to do my first paper, poster and conference talk – working with you was a great start to my PhD. Thank you to Jake and Ryan, for sharing your considerable knowledge and your eagerness to answer questions, and a huge thank you to Zhenghao for your hard work, patience, and perseverance through some mind-melting email chains.

Thank you to Dara McCutcheon and Jake Bulmer for giving feedback on earlier drafts of this thesis. Thank you to Alex Jones for proof-reading, constructive conversations, and much more besides.

Thank you to QETLabs and the CDT management for this opportunity, for which I am very grateful, and to Sorrel, Lin, and Holly for making our lives so much easier! In particular, thank you for the many wonderful outreach opportunities, that have been the highlights of my time in Bristol.

Aside from my PhD work, it has been a brilliant opportunity to be able to work with some exciting startups. Thank you Alex Moylett, the Deltaflow Run team, and everyone I was able to work with at Riverlane, it was a fantastic experience being able to see how the company runs and I learned a lot. Thank you as well to the whole team at Duality, for some very fun, interesting, and educational years.

There are many other people that I have had the pleasure to meet, talk to and work with. Here is an incomplete list of the people that have given me academic help, or career advice, that I am very grateful for: Jorge Barreto, Jake Bulmer, Neil Gillespie, Sorrel Johnson, Alex

[1] so far

Jones, Jonathan Matthews, Sam Morley-Short, Lawrence Rosenfeld, Andrew Royall, Paul Skrzypczyk, and Patrick Yard.

Thank you as well to the many people whose company has improved my life in Bristol. Thank you to the best CDT cohort of them all, cohort 6 – I'm so lucky we all started at the same time – especially Seb and Sam (and Tarini!), for sharing some particularly difficult months and filling them with good memories. Thank you to everyone else, whom I hope to see at many more pub trips. I'm lucky to say that there's too many of you to name, so I will take the coward's way out and not even try. We're often told to expect that a PhD can be quite a lonely process – I'm very glad that wasn't the case.

Thank you to the people bringing excitement to my life in Bristol outside of Physics: the Cube Microplex, Bristol Chamber Choir, Shaftesbury Avenue Ladies' Football Team, and the environmental activists I have had the pleasure to meet.

I am so thankful for my wonderful friends outside of Bristol, and I'm very grateful to you all. Thank you especially to Rosa and Daze, who let me vent endlessly about my impostor syndrome.

Thank you to my family, who asked specifically to be mentioned in these acknowledgements: thank you to Mum, Dad, Rachel, and Harry. Thank you for always letting me know it's fine to quit at any time, which made it much easier to keep going.

This thesis is dedicated to my grandad, Mike Higgins, who taught me the endless joy of scientific curiosity, and the value of my education – I don't take it for granted.

Author's declaration

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED: ..... DATE: .....

Table of Contents

Page

List of Figures **xi**

List of publications and presentations **xiii**

List of abbreviations, acronyms and initialisms **xv**

List of symbols **xvii**

1 Introduction **1**

2 Background **3**

2.1 Computers and quantum physics . . . 3
2.1.1 Computers . . . 3
2.1.2 Computational complexity . . . 6
2.1.3 Quantum physics . . . 14
2.1.4 Quantum information . . . 19
2.2 Quantum computing . . . 24
2.3 Quantum optics . . . 27
2.4 Linear optical quantum computing . . . 32
2.5 Graphs . . . 36
2.6 Boson sampling . . . 40
2.6.1 The model . . . 40
2.6.2 Computational complexity . . . 44
2.7 Continuous variable quantum information . . . 48
2.7.1 Phase space representation . . . 48
2.7.2 Gaussian states . . . 56
2.7.3 Implementation . . . 58
2.8 Gaussian boson sampling . . . 59
2.9 Simulation, emulation, and approximation . . . 65

3 Effect of photonic errors on quantum-enhanced dense subgraph finding **69**

3.1 Preface . . . 69
3.2 Introduction . . . 70
3.3 Background . . . 71
3.3.1 Dense subgraph finding . . . 71
3.3.2 Classical algorithms for dense subgraph finding . . . 72
3.3.3 Hafnians and density . . . 73
3.3.4 Using GBS to find dense subgraphs . . . 74
3.3.5 Sampling algorithms based on GBS . . . 77
3.3.6 Complexity considerations . . . 80
3.3.7 Other work . . . 81
3.4 Simulation methods . . . 82
3.4.1 Adding noise . . . 82
3.4.2 Sampling method . . . 85
3.4.3 Choosing the correct scaling parameter . . . 86
3.5 Results . . . 88
3.5.1 Random sampling with error . . . 88
3.5.2 Simulated annealing with error . . . 89
3.5.3 Sampling without postselection . . . 91
3.5.4 Average density of samples . . . 93
3.6 Discussion . . . 94
3.7 Outlook and further work . . . 98

4 Sampling the density of subgraphs 99
4.1 Preface . . . 99
4.2 Introduction . . . 99
4.3 Background . . . 100
4.3.1 Density of weighted graphs . . . 100
4.3.2 The Max-Haf problem . . . 101
4.3.3 Verification of sampling problems . . . 102
4.3.4 Rejection sampling . . . 105
4.3.5 Previous results . . . 107
4.4 Other sampling tasks . . . 108
4.4.1 Max Haf . . . 108
4.4.2 Complex-weighted DkS . . . 110
4.5 Density distribution . . . 112
4.5.1 Normalisation . . . 112
4.5.2 Verification with density as the ground truth . . . 113
4.6 Rejection sampling . . . 116
4.6.1 Complexity of rejection sampling . . . 116

4.6.2 Rejection sampling and anti-concentration . . . 118
4.6.3 Results . . . 119
4.7 Discussion . . . 122
4.8 Outlook and further work . . . 124

5 A complexity transition in displaced Gaussian boson sampling 125
5.1 Preface . . . 125
5.2 Introduction . . . 125
5.3 Background . . . 126
5.3.1 Evidence for the computational complexity of GBS . . . 126
5.3.2 Displaced Gaussian boson sampling . . . 127
5.3.3 The matching polynomial . . . 129
5.3.4 Noise and displacement . . . 132
5.4 Approximation schemes to the loop Hafnian . . . 134
5.4.1 Taylor expansion approximation . . . 134
5.4.2 Non-zero regions . . . 135
5.5 The computational complexity of DGBS . . . 139
5.5.1 Complexity of exact simulation . . . 140
5.5.2 Complexity of approximate simulation . . . 141
5.6 Loss and DGBS . . . 145
5.6.1 Lossy GBS as DGBS . . . 145
5.6.2 Loss tolerance of DGBS . . . 151
5.7 Discussion . . . 153
5.8 Outlook and further work . . . 154

6 Discussion and outlook 157

A Notation differences 161

B Range of scaling parameter 163

C Computational complexity of linear optics 165
C.1 Exact case . . . 165
C.2 Approximate case . . . 165

D Anticoncentration evidence 167

Bibliography 171

Index 191

List of Figures

FIGURE **Page**

2.1 Example graph. . . . 36
2.2 Example bipartite graph. . . . 37
2.3 Perfect matching of a bipartite graph. . . . 38
2.4 Perfect matchings of a general graph. . . . 38
2.5 Boson sampling. . . . 41
2.6 Boson sampling complexity proof structure. . . . 45
2.7 BS vs GBS comparison. . . . 60
2.8 Graph with loops. . . . 62
2.9 Perfect matchings with loops. . . . 63

3.1 Perfect matchings and density. . . . 75
3.2 DkS example graphs. . . . 88
3.3 Random sampling. . . . 90
3.4 Simulated annealing. . . . 91
3.5 Random sampling without postselection. . . . 92
3.6 Density of sampled subgraphs for different initial graph sizes. . . . 94
3.7 GBS distributions. . . . 96
3.8 Gaussian state transformation matrices. . . . 97

4.1 Complex-valued 24-mode matrix. . . . 108
4.2 Max Haf random sampling. . . . 109
4.3 Average square hafnian of samples. . . . 110
4.4 Complex-weighted DkS with error. . . . 111
4.5 Distribution of subgraph densities and hafnians. . . . 112
4.6 HOG test. . . . 114
4.7 CHOG test examples. . . . 115
4.8 CHOG test. . . . 116
4.9 HOG tests for rejection sampling. . . . 120
4.10 CHOG test for rejection sampling. . . . 121
4.11 Rejection sampling efficiency. . . . 122

5.1 Univariate matching polynomial. 131
5.2 Distribution of matching polynomial roots with w. 139
5.3 Photon number ratio of w. 140
5.4 Growth of $|w|$ with different loss values. 152

D.1 Anti-concentration of functions of Gaussian matrices. 168
D.2 Anti-concentration probability of matrix functions. 169

List of publications and presentations

The work presented in Chapter 3 was published in: **Naomi R. Solomons**, Oliver. F. Thomas and Dara P. S. McCutcheon, *Effect of photonic errors on quantum enhanced dense-subgraph finding,* Phys. Rev. Applied **20**, 054043 (2023) [1].

It was presented by N. R. S. at the following conferences (as *Using Gaussian boson sampling for dense subgraph finding with impure sources and loss*): QCTiP 2023 (talk, under the title *Gaussian-boson-sampling-enhanced dense subgraph finding shows limited advantage over efficient classical algorithms*), ICIQP 2022 (talk), Paraty 2023 (poster), QIP 2023 (poster), BQIT 2022 (poster, winner of the 'best presenter' prize).

The work presented in Chapter 5 was presented at BQIT 2023 by Zhenghao Li as a poster. It is in preparation as a paper entitled *A Complexity Transition in Displaced Gaussian Boson Sampling.*

Additionally, work done during this PhD was published in: **Naomi R. Solomons**, Alasdair I. Fletcher, Djeylan Aktas, Natarajan Venkatachalam, Sören Wengerowsky, Martin Lončarić, Sebastian P. Neumann, Bo Liu, Željko Samec, Mario Stipčević, Rupert Ursin, Stefano Pirandola, John G. Rarity, and Siddarth Koduru Joshi, *Scalable Authentication and Optimal Flooding in a Quantum Network*, PRX Quantum 3, (2022). This work is not included in this thesis.

It was presented by N. R. S. as a talk at BQIT 2021 and as a poster at Q-Turn 2020.

List of abbreviations, acronyms and initialisms

BS	boson sampling
CDF	cumulative distribution function
CEP	circular error probable
CHOG	Chen-HOG
CV	continuous variable
DGBS	displaced Gaussian boson sampling
DkS	densest k-subgraph
DV	discrete variable
ECT	extended Church-Turing (thesis)
FBQC	fusion-based quantum computation
FPRAS	fully polynomial-time randomised approximation scheme
FPTAS	fully polynomial-time approximation scheme
GBS	Gaussian boson sampling
GKP	Gottesman, Kitaev and Preskill
HOG	heavy output generation
i.i.d.	independent and identically distributed
JSA	joint spectral amplitude
KLM	Knill, Laflamme and Milburn
MBQC	measurement-based quantum computation
NTM	nondeterministic Turing machine
PNR	photon number resolving
PTAS	polynomial time approximation scheme
TM	Turing machine
TVD	total variation distance
XEB	cross-entropy benchmarking

Complexity classes are defined in 2.1.2. Those used in other sections of this work are:

BPP	bounded-error probabilistic polynomial time
BQP	bounded-error quantum polynomial time
FBPP	function problem **BPP**
NP	nondeterministic polynomial time
P	polynomial time
$\#\mathbf{P}$	counting problem polynomial time

List of symbols

$\hat{a}$	ladder/annihilation operator
A	A matrix: $(X \otimes \mathbb{1})(\mathbb{1} - \sigma_Q^{-1})$
$\mathcal{A}$	adjacency matrix
B	B matrix (for pure states): $A = B^* \oplus B$
c	speed of light OR graph scaling parameter
$d(G)$	density of graph G
$\mathbf{d}$	displacement vector (real basis)
E	set of edges in a graph
G	graph
H	rejection sampling scaling coefficient
$\hat{H}$	Hamiltonian operator
h	Planck's constant
$\hbar$	$h/2\pi$
k	subgraph size
K	number of squeezed modes
L	loss
m	number of modes
N	number of samples drawn
n	number of photons
$\bar{n}$	mean photon number
$\hat{n}$	(photon) number operator
$p(x)$	probability of outcome x
P	purity
$\hat{p}$	momentum quadrature operator
$q(x)$	probability of outcome x (proposal or simulation distribution)
r	squeezing parameter
S	subgraph
$\mathcal{S}_n$	symmetric group (over n elements)
t	time

T	quantum component of the covariance matrix
U	unitary matrix
V	set of vertices in a graph OR covariance matrix (real basis)
w	displacement/squeezing ratio
W	Wigner function OR classical noise of the covariance matrix
$\mathcal{W}$	total weight of graph
$\hat{x}$	position quadrature operator
$\mathbf{x}$	position
X	matrix with Gaussian i.i.d. elements OR Pauli X matrix
β	displacement parameter OR anti-concentration probability
$\boldsymbol{\gamma}$	modified displacement vector, $\boldsymbol{\gamma} = \boldsymbol{\delta}^\dagger \sigma_Q^{-1}$
$\boldsymbol{\delta}$	displacement vector (complex basis)
ϵ	error
η	transmission
ρ	density matrix
σ	covariance matrix (complex basis)
ψ	wavefunction
$\mathbb{1}$	the identity matrix: $\mathbb{1}_{ij} = \delta_{ij}$

CHAPTER 1

Introduction

It makes no damn sense.
Compels me though.

Benoit Blanc, *Knives Out*

This work is concerned with Gaussian boson sampling (GBS), a simplified, non-universal framework for linear optical quantum computing. Due to the limitations in available operations, it cannot realise the algorithms that have generated the considerable recent excitement in quantum computing. Nonetheless, it exploits the enhanced power of quantum systems for information processing, and is considerably more accessible to experimental implementations in the near future, having been the basis of several recent claims of quantum advantage. Not only does it require the implementation of many components necessary for full-scale universal quantum computers, making it an excellent experimental testbed, but it also reveals the computational power native to photonic systems within a more natural setting, and is thus well deserving of further study.

The paradigm of sampling problems presents unique challenges in determining the practical value of this scheme – it represents an unusual style of computation quite different from well-studied problem statements and algorithms, it presents difficulties in verification and evidence for the complexity of simulation, and it provides new vulnerabilities to being 'spoofed' by well-designed classical algorithms.

This thesis is motivated partly by the objective of examining regimes in which the computational power of GBS may fail to exceed efficient classical algorithms, and the evidence for this, and partly by the search for practical uses of GBS. There is considerable overlap in these topics, and in this work, we find that the formalism of graph theory is particularly useful as a basis of this study.

The thesis is structured as follows. In Chapter 2, we review the background theory needed to understand the topic, including computational complexity (Section 2.1.2), a brief introduction to graph theory (Section 2.5), and continuous variable quantum optics (Section 2.7). Gaussian boson sampling is introduced in Section 2.8. In Chapter 3, we consider the application of GBS to the problem of dense subgraph finding, using numerical simulation results to determine the impact on this application by different sources of error. We find that this application is very robust to error, even in regimes of error in which we do not expect quantum advantage to hold, which may imply that the improvement to be gained from GBS is not due to quantum effects. We extend this study in Chapter 4, by considering the difficulty of sampling directly from the distribution in which we are interested in this problem, the densities of subgraphs, and find that this is provably efficient in the case of unweighted graphs. We also consider other methods of assessing the performance of GBS for this problem, again finding that simulations with high levels of error still perform well, and also finding limited quantum advantage (by these metrics) in the case of complex-weighted graphs. Both chapters support the evidence that the negative terms corresponding to quantum interference are important signatures to indicate the potential for quantum advantage. Finally, in Chapter 5, we consider the impact of adding displacement to the state input to GBS on the computational complexity of the experiment. We do this based on previous results in graph theory concerning the matching polynomial, closely related to the loop hafnian, used to calculate the probabilities of different outcomes. We present an efficient classical algorithm that can be used to simulate GBS with displacements, and describe the regime in which it is applicable, as well as the possible use of this in simulating GBS with loss. We also consider the evidence for the hardness of displaced GBS, and why this does not hold for higher displacements.

Chapter 2 is a review of background information and not original work. Further chapters giving the results of this work begin with a statement describing the author's contributions and the contributions of other collaborators. Thank you to Victor Brena, who wrote the LaTeX template used for this thesis.

This is an 'annotated copy' of this thesis. This is not the formal, finalised thesis copy, but has further notes and derivations that might be of interest for readers who are less familiar with the topic. These purple notes are annotations, so if you can read this, you're reading the annotated copy! However all of the results of the work and the relevant background is included in the normal copy.

CHAPTER 2

Background

It's behind you
and in front of you
at the same time!

Ergo Phizmiz, *The Quantum Horse: A Cubist Pantomime*

2.1 Computers and quantum physics

In this section, we give some background context to motivate the study of the overlap of computer science and quantum physics, which may be familiar to most readers. Section 2.1.2 contains a list of computational complexity classes which will be used in this work.

2.1.1 Computers

The first recorded use of the word 'computer', in 1613, refers to a human (an 'arithmetician') skilled at doing mathematical calculations by hand [2–4]. This usage continued well into the 20th century, when 'computer' was a recognised profession, typically carried out by women, in fields including aeronautics, wartime data management, and scientific research [5]. Many of these people became early computer programmers when electronic computers became available [6].

The history of mechanical devices used for calculations also starts long before what we now recognise as a 'computer'. Early examples include the abacus, a frame holding beads that has been used for millennia [7], or slide rules, which were popular for school students before the widespread availability of the pocket calculator [8]. The remains of a mechanical orrery

dating back to approximately 100 BCE have been found (the *Antikythera mechanism*), a device which is used to model the orbits of celestial bodies with a high degree of accuracy [9].

There is an important distinction between, e.g. an abacus, or even a pocket calculator, and an orrery. An abacus can carry out a restricted set of calculations, due to a set of rules that describes how a variable (i.e. the number on beads of the right hand side of a wire) changes when certain operations are applied. By carrying out the appropriate sequence of operations, the variable changes in a well understood way, and the result will be an answer to a particular problem. The orrery also behaves in a well understood way. It can also be 'programmed' to begin in a particular input configuration, and there is a variable of interest (the position of objects representing the planets) which can be read out. This variable changes as the calculation proceeds, which is done via progressing a mechanism by hand or by clockwork. In both cases, we want to know the output of a particular function applied to the input. However, in the first case the machine is programmed to carry out an operation that implements that function, out of a space of possible functions limited by the architecture of the device. This is a broad generalisation of a model of *computation*, which we will formalise soon. On the other hand, in the latter case, the function is implemented as part of the physical model that describes the device, which behaves in a similar way as the system we wish to model, and hence it is realised as part of the intrinsic evolution of the machine's internal state. This is a *simulation* of the system of interest, which is closely related to computation, and can usually be carried out by computers.

Consider, for example, that you want to know how far an object travelling at velocity v will travel in time t. You could build a machine capable of doing the computation, $v \times t$, or you could build a simulation, a robot capable of travelling at velocity v, and measure how far it travels in time t. Doing the former calculation may seem more simple, but if the system of interest is very complicated, then programming and evaluating this function may be prohibitively difficult, in which case the latter experiment may be more useful.

The behaviour of modern computers (from this point on we will use the contemporary meaning of devices, not people) is based on the rules that govern electrical circuits. In theory, it is possible to implement any program that can be broken down into the *gates* that make up computational logic [10]. In order to be effectively processed, the data handled are stored as discrete variables. In accordance with the intrinsic architecture of a *classical* – which we use to mean non-quantum – computer, information is stored as binary numbers, which are strings of *bits*, digits which can take the value 0 or 1.

This is in contrast to analogue computers which store information in continuous physical quantities, e.g. voltage. Having access to a continuous variable space is very powerful, and if it were possible to control and measure this with arbitrary precision, analogue computers could carry out calculations with an efficiency that is considered mathematically infeasible by digital computers (assuming current beliefs about the complexity of these problems are

true) [11]. Nonetheless, this is not realistically the case.

Using discrete variables is preferred for several reasons. Increasing the number of bits allows calculations to be done with arbitrary precision, and the simplicity of this storage method allows for greater accuracy. Digital information storage and programming is more versatile and facilitates simpler error correction schemes [12]. Hence, due to their limited usefulness, analogue computers (of which the orrery is an early example) have not gained the same relevance in modern technology.

In fact, beyond using discrete variables, for most modern computation, using variables that can only take two values – Boolean variables – is preferred. These are composed using logical operators such as *and* and *not*. There are many texts to understand how the rules describing electrical circuits can be composed to create machines operating according to Boolean logic – I particularly recommend [13].

To better understand the limits of certain sorts of computers, there are many proposed models of computation, such as the Turing machine (TM) [14]. In brief, this machine is an automaton, whose internal state is one of a possible set of states, which receives a length of tape, with each box on the tape containing one of a set of symbols. The combination of the state and symbol decides the proceeding action of the Turing machine: it prints a new symbol onto the same box, and then moves to the left or right, or halts the computation. A full description of these models is omitted here but can be found in, e.g., [15, 16]. Turing's description was proposed to probe the question of computability: whether the solution to a particular problem can be found with finite resources (in terms of time or memory). However, the problem of the scale or efficiency of a machine to solve a particular problem requires further consideration.

It may not be immediately clear how this links to computers as we understand them, nor how this can capture the full range of possible processes. Nonetheless, the Church-Turing thesis [17] links TMs with the (informally defined) 'effective procedure', which is intended to encompass the mathematical (but not creative) steps a human being can carry out to solve a task. Essentially, it states that if you are able to propose an informal algorithm to solve a certain problem, well-described with an appropriate (finite) series of steps, then this can be programmed into an appropriately defined mathematical model of computation such as a TM [16]. This cannot be proven due to the lack of a rigorous definition for this concept, however the Church-Turing thesis is widely accepted as a foundation on which to base statements about the power of different forms of computation [15]. This also links together different models of computation, and shows that the set of computable problems is independent of the model chosen. However, we need more detail to tackle the problem of efficiency.

2.1.2 Computational complexity

Unless otherwise indicated, most of the information in this section is from [18]. However, the Wikipedia entries for most major complexity classes are also very helpful and detailed.

Informally, an *algorithm* is a set of instructions that can be carried out by a particular device (or program) to realise a certain computational process. A formal definition using the TM model of computation is a program that leads a TM from the input data of a problem to the output data representing the solution [15]. On the other hand, we can also describe a *heuristic*, which has some probability of failure, or may give a non-optimal solution. This is particularly useful in the case that 'optimal' is not well-defined.

When considering a particular algorithm, it is useful to describe its *computational complexity* (or just complexity) – roughly speaking, the amount of resources required to realise it. In this work, we are particularly interested in the time (or number of steps) required, although memory (space) is also frequently taken into account when discussing the complexity of algorithms. Trade-offs in in space and time are possible to identify, although they are not trivial.

In general, as the size of the input to an algorithm increases, so does its complexity. Hence, for an input of size n, the complexity will usually be given as a function of n. The complexity of the algorithm will also depend on the exact details of the input, so it is often useful to consider both the worst-case complexity (the largest possible amount of time required for any input of size n) and the average-case complexity (whether the problem can be solved within a certain amount of time, as a function of n, for some proportion of the inputs).

It is also particularly useful to consider the asymptotic complexity. Realistic details of the exact implementation of the algorithm and the physical machine will vary, so it is most interesting to consider the behaviour as n gets very large. This is of particular relevance in quantum computing where we will be focusing on problems that become unfeasible for classical computers. This is usually expressed with 'big O notation', in which $f(n) = O(g(n))$ indicates that [19]:

$$\exists\, M \in \mathbb{R}_+, n_0 \in \mathbb{R}_+,\ s.t.\ |f(n)| \leq M g(n)\ \forall\, n \geq n_0. \tag{2.1}$$

This says: 'there exists a positive real number, which we call M, and a positive real number which we call n_0, such that when the input n is bigger than n_0, the absolute value of $f(n)$ is upper bounded by $Mg(n)$'. In general, the complexity of an algorithm will be expressed as a function $f : \mathbb{N} \to \mathbb{N}$, which takes the size (e.g. number of bits required to write it) of the input and gives the number of steps in the algorithm, so n_0 will be in $\mathbb{N}$ (the 'natural numbers'). That is, it is possible to find a constant multiple of $g(n)$ such that, past a certain value of n, it is always greater than or equal to $f(n)$. The constant M is introduced to disregard overheads that may come from details in which we are less interested, e.g. how the information is stored, which introduce a constant factor to the program. Otherwise, this tells us that $g(n)$ 'grows

faster' than $|f(n)|$; if we disregard smaller terms that may make a difference when n is small, then we can consider $g(n)$ to upper bound $|f(n)|$ as long as n is sufficiently large. There are further variations on big O notation (including lower bounds) that won't be required in this thesis. In general, you can just take the part of the equation that grows the fastest (for example, exponential terms will dominate over polynomial terms, will dominate over linear terms), and take out any coefficients that don't depend on n. So if $f(n) = 2n^2 3^{1.5n} + n^{10} + 100n + 5$, then $O(f(n)) = n^2 3^{1.5n}$.

In general, 'efficient' is used to describe algorithms which run in polynomial time – that is, with running time $O(n^k)$ for some real, positive k. This may not seem like an intuitive definition, but at very large n, there is a significant difference between polynomial and exponential time. Furthermore, this allows us to consider an algorithm efficient if it calls an efficient subroutine a polynomial number of times (as $(n^k)^l = n^{kl}$), which is particularly useful, compared to e.g. defining efficient to mean linear [18]. This definition becomes more complicated when introducing error, which we consider later on. Generally, n is taken to be the size of the representation of the input, e.g. if the input is some number N, then n increases as $\log(N)$ (although this depends on the system, and base, of the representation of the number).

Thus far, we have not been clear on what 'number of steps' means. It seems as though this is a potential difficulty in defining the complexity of an algorithm, as this may differ depending on the device or model of computation used. In fact, this is well-defined in the case of TMs, and it is widely believed that this is the only model that needs to be considered. This is due to the extended Church-Turing thesis (ECT), which is the (more surprising) statement that 'every physically realisable computation model can be simulated by a TM with polynomial overhead' (introduced in [20], although here we have used the statement from [18]).

It isn't really possible to 'prove' either the Church-Turing thesis or the ECT, because of the necessity of making certain assumptions. Nonetheless, they can be proven if you carefully define limits on the models of computation you assume. One example is [21], discussed in [22]. What is important is that there is no need for ingenuity or creativity, and that the rules can be followed mechanically.

For that reason, if we are interested in the question of whether an algorithm is efficient, we can rigorously define the complexity classes of interest in terms of its implementation on a TM. Alternatively, we can describe another programming language, and count the number of steps in terms of the 'basic operations' used by that language (within reason), and we will not gain any advantage in efficiency.

As we will see, a major possible contradiction to the ECT comes from the development of quantum computers, which use techniques outside of the scope of what was originally considered in the theoretical description of computation. There are further possible issues

with the ECT, described in [18], but they are omitted here. Another particularly interesting potential challenge to the ECT is the possibility of efficient derandomisation, that is, whether $\mathbf{BPP} = \mathbf{P}$.

As well as the efficiency of algorithms, we will also be interested in the difficulty of particular mathematical problems. Problems are placed in a complexity class depending on the type of problem, the computational method of solving it, and the resources required to solve it. The complexity is defined by the resource requirements of the best algorithm that can solve it. There are a large number of different complexity classes, which are enumerated in the 'complexity zoo' [23].

For a complexity class $\mathbf{C}$, as well as identifying whether or not a problem is in $\mathbf{C}$, a problem can be $\mathbf{C}$-hard, if every problem in $\mathbf{C}$ can be (polynomial-time)[1] reduced to it – that is, given an algorithm that solves the given problem, any problem in $\mathbf{C}$ can be solved efficiently using this algorithm as a subroutine [15]. A problem is $\mathbf{C}$-complete if it is $\mathbf{C}$-hard and in $\mathbf{C}$.

Another useful piece of notation is $\mathbf{C}^{\mathbf{D}}$, the class of problems solvable by an algorithm in $\mathbf{C}$ with access to an *oracle* in $\mathbf{D}$. An oracle is a machine (a 'black box') that can solve a certain problem (or class of problems) with a single operation.

For example, $\mathbf{NP}$-hard: this problem (H) might not be in $\mathbf{NP}$, but every problem (L) in $\mathbf{NP}$ reduces to it. Equivalently, L can be solved in polynomial time with access to an oracle for H. $\mathbf{NP}$-complete: this problem is in $\mathbf{NP}$, and every problem in $\mathbf{NP}$ reduces to it.

It may be surprising that we can find such a problem. However, to give an example, we must first introduce some complexity classes. Some important complexity classes are (here we use n to mean the length of the input bitstring):

- $\mathbf{P}$: this is a class of decision problems, meaning Boolean functions – functions in which the output is one bit. Specifically, $\mathbf{P}$ contains problems that are computable in time cn^k for some $c, k \in \mathbb{R}_+$. These are the problems that a deterministic TM can solve in polynomial computation time. That is, the worst case running time for an input of size n is upper bounded by a polynomial in n, as opposed to exponential like $O(2^n)$, or combinatorial like $O(n!)$. Decision problems are a common framework for considering complexity due to their simplicity. Decision problems are often defined as formal languages. In this framework, the letters are {0,1}, and the words are inputs for which the solution is 1, or the TM accepts.

  Examples of problems in $\mathbf{P}$ include the connectivity problem (given two vertices of a graph, decide whether they are connected by a path of edges in the graph), and determining whether a number is prime [24]. Remember that the input, the number which may be prime, is a bitstring of length n, so the method of checking every number less than that number to see if it is a factor (or even half of that number) would take

[1] If not otherwise specified, 'reduction' in this work will refer to polynomial-time reduction.

at worst $2^n - 1$ steps (if we consider 'check if x is a factor' to be one step), which is exponential.

- **NP**: the class of decision problems for which, if the solution to the problem is 'yes' or 1, it has a *proof* which is verifiable in polynomial time. **coNP** is the class of problems with such verifiers for proofs where the solution is 'no'. It is clear that $\mathbf{NP} \supset \mathbf{P}$ - as these can be solved in polynomial time, solutions can be verified by solving the problem. This is equivalent to the class of problems solvable by a nondeterministic Turing machine (NTM; hence the name **NP**). These are equivalent because the NTM could make a good guess at a proof (e.g. if the question is 'does there exist an object A that does B', the proof would be guessing object A and then spending polynomial time to verify it does B). This may seem like cheating, because this is a 'best case' scenario for the NTM, whereas complexity arguments tend to consider worst case scenarios - however the question is whether there exists some NTM that happens to solve the problem in polynomial time, for any input, which is the 'luckiest guesser'.
  Examples of problems in **NP** include whether a graph contains a clique (which will be discussed further later on) and the Boolean satisfiability problem – these are both **NP**-complete.

Here we have seen examples of decision problems, the most important structure for computational problems. Alongside decision problems of a certain complexity, there are also *undecidable* problems, for which it is is impossible to construct an algorithm (an effective method) which gives the correct answer. A classic example proposed by Turing is the halting problem, although a more practical example is the matrix mortality problem: the 3×3 case is undecidable, and the decidability of the 2×2 case is currently an unanswered question [25].

A particularly interesting set of problems are the **NP**-complete problems. It is an open (and very famous) question in computer science whether $\mathbf{P} = \mathbf{NP}$. If any of the **NP**-complete problems permit a polynomial-time solution, that will show that $\mathbf{P} = \mathbf{NP}$ – however, most computer scientists do not believe this to be true [26, 27].

The theory of **NP**-completeness was developed independently by Cook and Levin [28, 29], focusing on the Boolean satisfiability problem. A Boolean formula consists of a statement made up of single bits $\{x_i\}$, composed by logical operators (AND, NOT and OR). The satisfiability problem asks whether there exists an assignment of $\{x_i\}$ such that the statement is true (or equal to 1). The proof that Boolean satisfiability is **NP**-complete involves transforming the strings defining TM computations into Boolean formulae. More details are given in [18].

Following this result, there is a well-known list of 21 computational problems which were shown to be **NP**-complete by Karp, including several problems in graph theory [30].

We introduce further complexity classes that will be relevant in this thesis:

- $\#\mathbf{P}$ (pronounced 'sharp P'): the class of counting problems associated with $\mathbf{NP}$ - that is, if the $\mathbf{NP}$ problem is of the form "are there any..." the $\#\mathbf{P}$ problem asks "how many...". This is equivalent to asking how many proofs for the existence of a solution there are, given that these proofs can be verified in polynomial time. Alternatively, this is how many paths of a polynomial-time NTM accept. An example is counting the number of satisfying assignments of a Boolean formula, or calculating the permanent of a binary matrix (which is $\#\mathbf{P}$-complete). $\#\mathbf{P} \supseteq \mathbf{NP}$ (if you know how many satisfying solutions there are, you know whether one exists).

- $\mathbf{Gap-P}$ [31]: this is the closure of $\#\mathbf{P}$ under subtraction; problems which count the number of accepting paths minus the number of rejecting paths. This allows us to consider, e.g. the permanents of matrices with negative values, as we can express this as the difference between two $\#\mathbf{P}$ problems. The hardness is therefore similar to $\#\mathbf{P}$, however we now have additional questions such as asking the sign.

- $\mathbf{BPP}$: the class of decision problems solvable in polynomial time, with error $\leq 1/3$ (or possibility of failure), by a probabilistic Turing machine[2] (more generally, this encompasses algorithms that have some element of randomness, for example by making a random choice between two different actions at a certain stage, with a certain probability). For example, this could be an algorithm that flips a coin at every stage, and implements one of two possible actions depending on the outcome. A probabilistic Turing machine is a type of non-deterministic machine, but without the 'luckiest guesser' attribute. It's also required that the TM runs in polynomial time for every input. $\mathbf{BPP}$ problems are considered much more feasible for realistic computers. This machine could just have all probabilities set to 1 or 0, which would make it a deterministic Turing machine - hence we have $\mathbf{P} \subseteq \mathbf{BPP}$. However, it is conjectured that $\mathbf{P} = \mathbf{BPP}$, which would suggest there is no extra advantage to be gained from introducing randomness into a computation, but this is an open question, as there still remain some problems which are in $\mathbf{BPP}$ but are not known to be in $\mathbf{P}$.

- $\mathbf{BQP}$: the class of decision problems solvable by a quantum TM (or quantum computer) in polynomial time, with error $\leq 1/3$. This is the quantum analogue to $\mathbf{BPP}$; because of the inherent randomness involved in quantum computers, this is more appropriate to use than a quantum analogue to $\mathbf{P}$.

- $\mathbf{BPP_{path}}$: consider the possible 'paths' that a nondeterministic quantum computation can take. These may have different probabilities. In the complexity class $\mathbf{BPP}$, if the input has solution 1, it's required that this is the output $\geq 2/3$ of the time (and vice versa). $\mathbf{BPP_{path}}$ requires that this is the case for $\geq 2/3$ of paths. This is more powerful

[2]The choice of value for the allowed error is arbitrary, as long as it is below $1/2$; these definitions are equivalent, as lower error bounds can be achieved by running an algorithm multiple times.

than **BPP**, as it includes some problems that may have a high probability branch (in some frameworks this is one that terminates after fewer steps) which gives the wrong output.
We can include postselection (defined in the next bullet point, or [32] for classical computers). We find that $\mathbf{BPP_{path}} = \mathbf{PostBPP}$ (by using postselection to average over random choices of paths). More detail, and a proof that $\mathbf{BPP_{path}} = \mathbf{PostBPP}$ is in Appendix B of [33].

- **PostBQP** [34]: we now introduce postselection, a process by which we only keep some subset of the outcomes of a computation (or, e.g. outcomes of an experiment), conditioned on a certain property. More formally, this can be represented by an ancilla qubit which marks whether the computation succeeds or fails. Then, as with **BQP**, this includes decision problems solvable in polynomial time by a quantum TM, and, given that the computation succeeds (or 'postselecting' only on outcomes with the ancilla qubit in the state $|1\rangle$), the answer is correct with probability $\geq 2/3$. This is more powerful than **BQP** – for example, the answer only has to have a bounded probability of error if a certain condition holds (the one we postselect on), not over all cases. Also, it doesn't restrict how often the computation succeeds – so for example, it may take an exponential number of runs until the computation succeeds, but each of these only takes polynomial time.

  Postselection is particularly relevant in the case of boson sampling, in which we can't *deterministically* do two-qubit gates. However, if we can postselect on certain outcomes, we can implement these gates – meaning that with postselection, boson sampling is a universal quantum computing model (this will be explained more in Section 2.4).

- **FP**: the set of problems where the output is more than one bit (i.e. non-Boolean functions) solvable in polynomial time by a deterministic TM. These problems can still be efficiently computed by classical computers, but they are not decision problems.

- **FBPP**: a generalisation of **BPP** to function problems - that is, the algorithm gives an output which has more than one bit, with error $\leq 1/3$.

We note the existence of other quantum complexity classes intended as quantum analogues to classical classes, such as **QMA**, which are not included in this thesis. Finally, we introduce the polynomial hierarchy (**PH**). This is going to be useful when we have yes/no problems, but where the structure is slightly more complicated.

- $\Sigma_0^{\mathbf{P}} \equiv \mathbf{P}$.

- $\Sigma_{i+1}^{\mathbf{P}}$: the class $\mathbf{NP}^{\Sigma_i^{\mathbf{P}}}$, so problems for which the solution can be verified in polynomial time with access to a $\Sigma_i^{\mathbf{P}}$ oracle. This means $\mathbf{\Sigma_1^P}$ contains **NP**.

- $\Delta_{i+1}^{\mathbf{P}} = \mathbf{P}^{\Sigma_i^{\mathbf{P}}}$.
- $\Pi_{i+1}^{\mathbf{P}} = \mathbf{coNP}^{\Sigma_i^{\mathbf{P}}}$.
- **PH**: this is the union of all classes in the polynomial hierarchy - that is, the union of all $\Sigma_i^{\mathbf{P}}$, as well as all $\Delta_i^{\mathbf{P}}$ and $\Pi_i^{\mathbf{P}}$.

Several methods of understanding the polynomial hierarchy are given in [18]. Informally, this extends problems of the form “do there exist cases such that...” to "do there exist cases such that, for every other case..." or even “do there exist solutions such that, for every other case, there exists...". A more detailed pedagogical introduction is also given in [35]. Proving that $\mathbf{P} = \mathbf{NP}$ is one method of showing that the polynomial hierarchy collapses (i.e. that the levels are not distinct, beyond a certain stage) but, similarly to $\mathbf{P} =? \mathbf{NP}$, it is widely thought that it does not collapse [26].

Let’s give a more detailed explanation (as well as Arora and Barak, I used Eshan Chattopadhyay’s 2021 lecture notes, which give more formal detail). Firstly, we’ll express binary problems as *languages*, where x is in a language if the answer is yes, as we saw previously. For example, consider the k-clique decision problem: the words are graphs of n, and they are in the language if they contain a clique of size k (you can skip ahead to Section 2.5 for a graph theory introduction). This is a problem which is expressed through the quantifiers $\exists$:

$$x \in L \iff \exists\, u \in \{0,1\}^{q(|x|)} \;:\; M(x,u) = 1, \tag{2.2}$$

where in this case, x is a graph, u is a subgraph (in this case it is the *certificate* for x), and M is a (Turing) machine. If q is a polynomial and M runs in polynomial time, then the language is in **NP** (it can be verified in polynomial time).

Contrast this with **coNP**, which is the complement to **NP** – that is, if the answer is no, there is a proof which is verifiable in polynomial time. For example, “is it true that there are no cliques of size k in this graph?”, for which the proof would be a clique. A more natural example is: given numbers m and k, is there a factor of m which is less than k and greater than 1? In this case, if the answer is no, a proof would consist of all the prime factors of m: in polynomial time, you can check that these are all greater than k, and that they do indeed multiply to m. **coNP** problems can be written in the following way:

$$x \in L \iff \forall\, u \in \{0,1\}^{q(|x|)} \;:\; M(x,u) = 1. \tag{2.3}$$

This structure can be used for a lot of different decision problems (they may be more complex than **NP**, depending on the properties of M and q). However, some have a more complex structure. Let’s introduce the problem of graph colouring. A k-colouring is a way of assigning each vertex of a graph one of k different colours, so that no neighbouring vertices have the same colour, and the problem: does this graph admit a k-colouring? is **NP**-complete for $k \geq 3$ (so we can write this as in Eq. 2.2).

Now consider a list colouring, which is where every vertex is given a list of allowed colours, and then a colouring is made where the colour of each vertex comes from the appropriate list. Again, colourings are acceptable only if neighbouring vertices have different colours. A graph is k-choosable if a graph colouring is possible given any choice of lists of length k. It may seem as through a graph being k-colourable means it is k-choosable, but this is not the case – for a counterexample, see the Wikipedia page for 'list coloring'.

The problem of list colouring can be expressed as:

$$x \in L \iff \forall\, u \in \{0,1\}^{q(|x|)}\ \exists\, v \in \{0,1\}^{q'(|x|)}\ :\ M(x,u,v) = 1. \tag{2.4}$$

That is, a graph is in L if it is k-colourable; for any assignment of lists of length k, u (q is polynomial in $|x|$), there exists a colouring v (ditto for q') such that $M(x,u,v) = 1$ if the colouring is valid (satisfies the list constraints and has no neighbours the same colour). Note the alternation of the qualifiers $\forall$ and $\exists$ – repeating qualifiers doesn't give us any extra power.

This problem is in the complexity class $\mathbf{coNP^{NP}}$. The graph colouring problem is in $\mathbf{NP}$, and if the answer is no you can efficiently verify a proof (a list assignment) that produces the answer 'no' when calling this oracle. Nonetheless, just being able to solve the graph colouring problem is not itself sufficient to solve the problem. Using the rules introduced above, this is in the second level of the polynomial hierarchy: $\Pi_2^{\mathbf{P}}$.

For a level i of the polynomial hierarchy, a problem can be expressed as a set of i terms quantified alternately by $\forall$ and $\exists$, where problems in Π_i start with $\exists$, and Σ_i start with $\forall$.

There's much more to say about the polynomial hierarchy. There's a chance it will collapse down to some finite level like a telescope, and if it collapses all the way to the bottom, this is $\mathbf{P} = \mathbf{PH} \iff \mathbf{P} = \mathbf{NP}$.

Proving the complexity of a particular problem is difficult, and many results rely on unproven (but widely accepted) conjectures, such as $\mathbf{P} \neq \mathbf{NP}$. If you can find an algorithm that runs in polynomial time that solves the problem, that means that the problem is in $\mathbf{P}$; on the other hand, to show that the problem is *not* in $\mathbf{P}$ is more complicated (which is not to suggest that finding polynomial-time algorithms for tricky problems isn't hard!). One standard procedure is to show that, if a polynomial-time solution to the problem existed, this would allow a reduction to another known problem, and the fact of that problem being in e.g. $\mathbf{P}$ would lead to a contradiction (see, for example, [36]).

One theorem that is important to later proofs described in this thesis is:

Theorem 2.1 (Toda, [37])**.** $\mathbf{PH} \subseteq \mathbf{P}^{\#\mathbf{P}}$.

The proof is described in [18]. This means that a decision problem that can be solved in polynomial time with access to a $\#\mathbf{P}$ oracle must be at least as hard as any level of the polynomial hierarchy.

2.1.3 Quantum physics

Unless otherwise indicated, most of the information in this section is from [38].

Now we must introduce quantum physics, which describes the mechanics of the physical systems that we will be interested in. Quantum physics is a vast subject that cannot be covered in its entirety in the introduction to this thesis (nor is it necessary to), but we will give some context to better understand the field that gave rise to quantum computing.

Quantum mechanics is mostly relevant at very small scales (on the atomic or subatomic scale), and hence what we tend to observe in the macroscopic universe obeys the rules of classical (non-quantum) mechanics, which includes the laws of Newton and Maxwell, as well as relativity, introduced by Einstein. These laws become increasingly relevant at larger scales due to the process of decoherence – more specifically, this refers the loss of quantum coherence. Coherence corresponds to the capability of wavefunctions to interfere with each other, and can be understood as the ability of a quantum system to maintain a superposition. It has been shown to be operationally equivalent to entanglement [39]. Environmental interactions cause decoherence, which can be considered to be caused by entanglement with the environment. The study of decoherence, as well as the closely related study of open quantum systems, are active areas of research, but outside of the scope of this thesis.

The term *quantum* is related to *quantisation*, the realisation that physical quantities which were previously thought to be continuous must instead take discrete values. This originates in the early 20th century with Planck's studies of black body radiation.

Planck's original area of expertise was in thermodynamics. His work led him to consider black body radiation, which is the study of the radiation emitted by an object that perfectly absorbs radiation, at thermal equilibrium. The power of this radiation (per unit area) depends on both the temperature of the body and the wavelength at which it is emitted. At a particular temperature, there is a wavelength at which this power peaks, and then it decays as the wavelength approaches zero/infinity. For very small wavelengths, the theoretical predictions for the amount of energy radiated by a black body (at a certain temperature) did not match experimental results, and grew to infinity at wavelengths approaching zero, in a problem known as the 'ultraviolet catastrophe'.

These predictions were based on considering the behaviour of the individual atoms in the black body (typically considered to be a cavity), which act as electric dipoles and can be treated as (classical) linear harmonic oscillators. That is, the separate positive and negative charges vibrate (the distance between them oscillates) with a frequency $\nu = c/\lambda$, where c is the speed of light and λ corresponds to the frequency of the electromagnetic radiation being absorbed/emitted [40]. The energy of these oscillators can then take any value, which means that the average energy (using classical thermodynamics) of the oscillators is kT, where T is the temperature and k is Boltzmann's constant. As a result the spectral distribution function (the distribution of emission power in terms of wavelength and temperature)

becomes $\rho(\lambda, T) = 8\pi kT/\lambda^4$ (see [38] for further details). Planck's theory solves the ultraviolet catastrophe by only allowing the energy of an oscillator to take discrete values, which are integer multiples of some energy ε. ε is a finite *quantum* of energy, possibly dependent on the frequency of the oscillator. At long wavelengths, the differences between these energy levels are so small that the quantum behaviour is essentially indistinguishable from the classical theory.

Then, instead of an integral over the possible energies, it becomes a discrete sum (see [38] for details). This means that the average energy of the assemblage of oscillators then becomes $\varepsilon/(\exp(\varepsilon/kT) - 1)$, and the spectral distribution function becomes

$$\rho(\lambda, T) = 8\pi\varepsilon/\lambda^4(\exp(\varepsilon/kT) - 1).$$

Using the previously known thermodynamic properties of the system, we find that $\varepsilon = hc/\lambda$, for some constant h, where $c = 2.998 \times 10^8\,\mathrm{m\,s^{-1}}$ is the speed of light. Using experimental data, it has been found that $h = 6.626 \times 10^{-34}\,\mathrm{J\,s}$, now known as Planck's constant. Note how small this is – this is in agreement with our observation that quantum effects are too small (in scale) to have a major impact on the macroscopic world that we inhabit. The concept of quantisation was later extended to the electromagnetic field itself, leading to Einstein's work on the photoelectric effect, and further developing the understanding of black-body radiation. We will continue the quantisation of the electromagnetic field further in Section 2.3. Further experiments confirmed the quantisation of other quantities, such as spin, although this will be less relevant in this work.

Throughout the history of the study of light, different theories seemed to point towards light exhibiting properties of either a wave or a particle. Planck's theory of quantisation, and further experimental evidence, seems to definitively describe electromagnetic radiation as particles. De Broglie proposed that other subatomic particles, including the constituents of matter, can concurrently display properties of both waves and particles, a principle known as wave-particle duality.

The canonical experiment displaying this phenomenon is the double slit experiment, first performed as a demonstration of classical optics but later used to show principles associated with quantum mechanics. A single particle is produced(e.g. with an electron gun), which travels through a screen with two slits in it, and then its position is measured when it is incident on a second screen. If this is repeated many times and the positions of these particles are recorded, the cumulative results show a distinctive interference pattern, similarly to a water wave travelling through the two slits.

Two important conclusions can be drawn from this experiment. The first conclusion of this is that the wave-like nature of the particles plays a role in their dynamics. To describe particles using this formalism, we introduce the wavefunction, $\psi(\mathbf{x}, t)$, which is a function of position and time. The wavefunction is complex-valued(which is required in order to get

the correct interference outcomes), but can be interpreted as a probability amplitude that governs a real observable probability density $P(\mathbf{x},t) \propto |\psi(\mathbf{x},t)|^2$ – a relationship known as the Born rule. The Born rule arises from the statistics associated with quantum behaviour, and is analogous to similar results in the interference of classical waves.

Quantum behaviour can be derived from a set of 'postulates' (see, e.g. [41]), in which Born's rule is generally included, i.e. it is an axiom of the theory and not derived from first principles. However, the choice of postulates is not unique, and by beginning with a different set of axioms it is possible to derive the Born rule [42]. It can also be derived from other assumptions, or even interpretations [43]. Alongside these postulates, some ongoing research aims to determine what other features of quantum mechanics are strictly necessary [44].

The second conclusion is the principle of superposition. We can consider the experimental results for if one slit at a time were closed, while the other remained open. If we then combine these, we would get a different outcome to what happens when both are open simultaneously. Hence, the trajectory of the particle depends on the wavefunction having some amplitude in each slit, and therefore they cannot be considered independently, nor can the outcome be seen as a statistical artefact from a single slit being chosen at each run of the experiment.

Mathematically, superposition arises from allowing any linear combination (suitable normalised, as we will see) of two possible states as a description of another possible state. This is generally included in the postulates of quantum mechanics, which is particularly clear when considering the (equivalent) matrix formalism.

This may not be particularly surprising when we consider the wave-like nature of photons or electrons. However, the consequences are more clear when we consider properties that are described by the wavefunction which, in a classical view of the universe, should be fixed for the object of interest, such as the energy/number of photons, or path taken. Further examples will be considered in Section 2.3.

Wavefunctions that allow properties of interest (e.g. photon number) to take a superposition over different values may lead to the correct outcomes for statistical averages, but it does not match our perception of reality. Notably, as much as the wavefunction may contain some amplitude of an electron being present in multiple locations, it is only measured in exactly one of them (assuming no experimental errors).

When a measurement occurs, the wavefunction *collapses*. This is a spontaneous and probabilistic process by which a property of a quantum system becomes well-defined as an eigenstate of the measurement operator, as we will formalise in the following section. The wavefunction of the system is deformed in a non-continuous, random way to reflect the particular value given to that property.

An interesting feature of quantum theory is that the collapse of the wavefunction shows objective randomness – that is, the wavefunction is taken to be a complete description of

the system, but the result of a measurement cannot be predicted deterministically, but only the statistics of different outcomes. This is one of the most surprising features of the theory, and the counter-intuitive nature of phenomena such as wavefunction collapse are a major contributor to the reputation of quantum physics as being difficult to understand[3].

This has lead to a considerable history of discussion on how to interpret quantum mechanics. Popular interpretations include:

- the Copenhagen interpretation: this accepts the indeterminism of quantum mechanics, and generally understands that the only meaningful information that can be known about an object are from the result of measurements. Certain variations align with the positivist philosophy which was particularly popular at the time, although now less so [45].
- the many-worlds (Everettian) interpretation: this states that the wavefunction is objectively real and deterministic, and that wavefunction collapse does not occur, but that different measurement outcomes correspond to the different worlds in which an observer can exist, all present within the universe [46].
- the pilot wave (de Broglie-Bohm/Bohmian mechanics) interpretation: this is a deterministic, non-local theory that states that all particles have well-defined positions, but that their motion is determined by an underlying guiding equation with the same form as the associated wavefunction [47].

The Copenhagen interpretation aligns with the view that is mostly commonly taught in undergraduate lecture courses, but there is no general scientific consensus on what is 'correct'.

The wavefunction of a quantum particle, as well as describing its state, is the basis of an equation of motion which we can use to describe its time evolution. In particular, we will introduce a partial differential equation, known as Schrödinger's equation, the solutions of which describe the wavefunction of the quantum state: in order to understand where this comes from, it is useful to compare to wave mechanics (although we note this is not a derivation). We follow the argument in [38].

The equation of motion of a plane wave is given by:

$$\psi(\mathbf{x},t) = Ae^{i(\mathbf{k}\cdot\mathbf{x}-\omega t)}, \tag{2.5}$$

[3] Certain aspects of quantum physics – such as quantum field theory – do have particularly challenging maths, but this is certainly not unique to quantum physics, and the linear algebra techniques required in quantum information theory are comparatively straightforward. Furthermore, many fields of science contain philosophical or cognitive difficulties, or even paradoxes; consider the simplicity of the human brain/genome, the vast time and distance scales involved in evolutionary biology or astrophysics, or any field that attempts to understand human behaviour. My personal belief is that a characterisation of quantum physics as being 'impossible to understand' not only dissuades too many people, from the mildly interested to the potential physicist, in engaging with a fascinating and potentially useful subject, but does a disservice to the many scientists, philosophers, and educators who have devoted so much of their time to helping us understand it.

where A is a constant, ω is the angular frequency $\omega = 2\pi f$, and $E = hf$ following the Planck relation. We introduce $\hbar = h/2\pi$, so that $E = \hbar\omega$. $\mathbf{k} = \mathbf{p}/\hbar$ is the wavevector, where $\mathbf{p}$ is the momentum of the particle associated with the wavefunction. $|\mathbf{p}| = h/\lambda$, from the de Broglie relation confirming the wavelike nature of matter. This relation allows us to introduce an equation that encompasses both the wavelike and particle-like properties of the object.

In analogy to the classical equation $E = \mathbf{p}^2/2m$, where m is the mass of the particle, we have:

$$\omega = \frac{\hbar k^2}{2m}. \tag{2.6}$$

We note that the wavefunction satisfies the relations:

$$-i\hbar\nabla\psi = \mathbf{p}\psi \tag{2.7}$$

$$i\hbar\frac{\partial}{\partial t}\psi = E\psi, \tag{2.8}$$

which are the quantum mechanical operators for momentum and energy respectively.

We now introduce a potential $V(\mathbf{x}, t)$ to represent a field of some force in which the particle is located, which means the total energy of the particle is:

$$E = \frac{\mathbf{p}^2}{2m} + V(\mathbf{x}, t), \tag{2.9}$$

where the first term represents the kinetic energy. Hence, we find that the wave satisfies the equation:

$$i\hbar\frac{\partial}{\partial t}\psi(\mathbf{x}, t) = \left[-\frac{\hbar^2}{2m}\nabla^2 + V(\mathbf{x}, t)\right]\psi(\mathbf{x}, t). \tag{2.10}$$

In the time-independent case, this leads to the eigenvalue equation:

$$E\psi = \hat{H}\psi, \tag{2.11}$$

where $\hbar = h/2\pi$, $\mathbf{x}$ and t are the position and time at which ψ is evaluated, which may be omitted if they are not required to be specified in the context, m is the mass of the particle, V is the potential of a field that the particle is in, and E is the total energy. In accordance with the superposition principle, any linear combination of solutions to Eq. 2.10 is also a solution.

This introduces the Hamiltonian operator, $\hat{H}$, which gives the energy of the state. In classical mechanics, the Hamiltonian formalism is a useful method of describing the evolution of a physical system. The total energy of the system is given by the Hamiltonian, H, which is a function of the position and momentum of all of the constituent bodies. This is useful in describing the system, as by solving $i\hbar\partial_t\psi = \hat{H}\psi$, we can describe the time evolution of the system, $\psi(t_1) = \mathrm{e}^{-i\hat{H}(t_1-t_0)/\hbar}\psi(t_0)$. We note that this is a non-relativistic equation, which

will be fine for our purposes – a description of relativistic wave equations, such as the Dirac equation, is ommitted in this work. We will return to this in the following section.

Another property of quantum systems that is particularly relevant to quantum information theory becomes apparent when considering many-body systems. It was identified as a feature of quantum theory by Einstein, Podolsky and Rosen, but proposed as a likely contradiction with reality [48]. We can consider a system in which two particles are emitted, with a particular total energy, in a superposition of different frequencies. By conservation of energy, measuring the frequency of one particle will cause the wavefunction of the other to collapse instantaneously. This apparent action at a distance is caused by entanglement, the superposition of a many-body system into non-separable terms, which we will illustrate in more detail in the following section. Bell's theorem showed that the correlations of entangled quantum systems cannot be produced by any local hidden variable model, that is, any model where the properties of objects are well-defined before measurement [49], and where the only influences on an object come from causes with its light-cone ('locality'). The existence of entanglement was confirmed experimentally by Bell tests [50].

When considering the quantum nature of atoms and molecules, it is particularly useful to study the angular momentum of particles. This is also quantised, with the different possible values being indicated by the orbital angular momentum quantum number. It is also useful to separately consider the angular momentum in the z-direction, which is defined by the magnetic quantum number (due to its interaction with magnetic fields). However, experimental results showed that this was not sufficient to describe atomic spectra, and other results including the Stern-Gerlach experiment. A more detailed introduction to these concepts is given in [38].

Particles also have a property known as spin, corresponding to an intrinsic magnetic moment, or intrinsic angular momentum, with possible values represented by the spin quantum number, s. Particles with integer spin are known as bosons, whereas particles with spin $1/2, 3/2, 5/2...$ are known as fermions. The wavefunction ψ is dependent on this value, as well as position and time. Further differences between bosons and fermions will be given in the following section, once we have introduced more of the tools that we require for describing quantum systems.

2.1.4 Quantum information

Now we have seen the concepts underlying quantum theory, we can detail the formalism that underpins information processing tasks that use quantum states. Unless indicated otherwise, information in this chapter can be found in [38, 51, 52]. We will not introduce the mathematical apparatus of linear algebra that is necessary from this point onwards, but this is also described in [52].

The previous section briefly introduced the wave formalism of quantum mechanics, but this work will generally represent quantum systems using the matrix formalism. Although equivalent, these do not obviously appear the same, and Dirac notation is particularly successful in uniting them.

We denote a quantum state by a 'ket', $|\psi\rangle$. It will also be useful to introduce its dual, a 'bra', $\langle\psi|$. In the wave formalism, we can describe a quantum state in a particular basis e.g. using the position representation (where $\psi(\mathbf{x}) = \langle\mathbf{x}|\psi\rangle$) and then the dual is the complex conjugate of $\psi(\mathbf{x})$, and we have the inner product ('braket'):

$$\langle\psi_1(\mathbf{x})|\psi_2(\mathbf{x})\rangle = \int \psi_1^*(\mathbf{x})\psi_2(\mathbf{x})\mathrm{d}\mathbf{x}. \tag{2.12}$$

We generally require quantum states to be normalised, so $\langle\psi|\psi\rangle = 1$ (so that the probability of finding the particle *somewhere* is one). Two quantum states are orthogonal if $\langle\psi_1|\psi_2\rangle = 0$.

A superposition of quantum states corresponds to a linear sum, $|\psi\rangle = c_1|\psi_1\rangle + c_2|\psi_2\rangle$, $c_1, c_2 \in \mathbb{C}$. Consider a complete (possibly infinite) set of orthonormal ($\langle\psi_i|\psi_j\rangle = \delta_{ij}$) wavefunctions $\{\psi_i\}$ – that is, the wavefunction of any dynamical state can represented as a linear combination of these basis functions:

$$\psi = \sum_i c_i\psi_i,\ c_i \in \mathbb{C}. \tag{2.13}$$

Generally, these are the eigenfunctions of a linear, Hermitian operator. For a finite, discrete set of $\{\psi_i\}$, these can be used as a basis by which any wavefunction can be fully defined by the column vector of $\mathbf{c} = \{c_i\}$. Basis choices can be infinite, continuous, and even non-orthogonal – for example, the coherent states (that we will see later) form an overcomplete basis for continuous variable quantum states. In the matrix formalism, $\langle\psi|$ is the Hermitian conjugate (conjugate transpose) of $|\psi\rangle$, and $\langle\psi_1|\psi_2\rangle = \mathbf{c}_1^\dagger\mathbf{c}_2$ (where ψ_1 [ψ_2] is represented by the column vector $\mathbf{c}_1$ [$\mathbf{c}_2$]).

Mathematically, a vector space equipped with an inner product, as we have just described, defines a Hilbert space. A larger Hilbert space is formed when considering multiparticle systems, constructed using the tensor product: $|\Psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle$. An entangled state is one that cannot be represented by a tensor product of single particle states: $|\psi\rangle \neq |\psi_1\rangle \otimes |\psi_2\rangle$.

It will also be useful for us to consider mixed states, where we introduce classical uncertainty (that is, lack of information about the system). A mixed state that is some classical mixture of states $|\psi_i\rangle$, each with probability p_i, can be expressed using the density operator:

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|. \tag{2.14}$$

Sometimes it is not immediately clear why we consider classical uncertainty, as this is not an ontological artefact (as we consider quantum uncertainty to be) but an epistemological

one. However, this is necessary to further understand important qualities of how we interact with quantum information, including no-signalling constraints, and how much information we can extract from a system. Being able to trace over other systems is vital for considering locality, and this is a valuable use of mixed states. It also helps us understand how an entangled system evolves after measurement, and it is vital in considering error.

As with classical computers, as described in Section 2.1.1, it will be more convenient if we digitise the system and use an orthonormal basis, in particular using two basis states, which are usually denoted $|0\rangle$ and $|1\rangle$. These states can be represented by vectors $\in \mathbb{C}^2$, and are called qubits (short for 'quantum bits').

We then transform states using operators, such as the time evolution operator introduced in the previous section, generally denoted by a circumflex accent, i.e. $\hat{A}$. We typically use linear operators, which, in the case of a finite-dimensional Hilbert space, can be described by matrices, in some basis $\{\psi_i\}$: $A_{ij} = \langle\psi_i|\,\hat{A}\,|\psi_j\rangle$. Typically we will use $\hat{M}$ to denote an operator in Hilbert space and M to denote its associated matrix, although these are generally interchangeable.

Recall the Hamiltonian operator, $\hat{H}$. Probability conservation implies that $\hat{H}$ should be Hermitian (further details are given in [38]; we find this by demanding that the integral of the total probability of finding the particle anywhere is constant with time), meaning that $\hat{H} = \hat{H}^\dagger$, although this is not always the case – see, e.g., [53]. This implies that $\hat{U} = \mathrm{e}^{-i\hat{H}t/\hbar}$ is unitary [51], that is, $\hat{U}\hat{U}^\dagger = \hat{\mathbb{1}}$, and therefore we find that the evolution of quantum states is described by unitary operators (generated by the system Hamiltonian). This implies the reversibility of quantum transformations, and importantly, of quantum computation. It also preserves the orthonormality of the basis. A universal quantum computer is able to approximate an arbitrary unitary (in the appropriate Hilbert space) with arbitrary precision (albeit at a possible exponential cost in the number of gates).

Describing time evolution by considering the action of fixed operators, that evolve the state through time, is known as the Schrödinger picture. It can be shown that this is equivalent to the Heisenberg picture, with time dependent operators acting on time-independent states. Thus, instead of applying the operator $\hat{A}$ to the state $|\psi(t)\rangle = \hat{U}(t,t_0)\,|\psi(t_0)\rangle$, we apply the operator $\hat{A}(t,t_0) = \hat{U}^\dagger(t,t_0)\hat{A}\hat{U}(t,t_0)$ to the fixed state $|\psi(t_0)\rangle$.

We now consider the measurement process in this formalism. A more complete description is given in [52], but for the following work it is sufficient to use the following principles:

- Measurements correspond to linear operators which can be represented by matrices, M, which must be Hermitian in order to have real eigenvalues (i.e. to correspond to real observables).
- We write M in its spectral decomposition:

$$M = \sum_i \lambda_i P_i, \tag{2.15}$$

where $\{\lambda_i\}$ are the eigenvalues of M, associated with eigenvector $\mathbf{v}_i$ defining eigenstate $|v_i\rangle$, and P_i are the matrices associated with the projection operators $\hat{P}_i = |v_i\rangle\langle v_i|$.

- The possible measurement outcomes on state $|\psi\rangle$ are then λ_i, which occur with probability $\langle\psi|\hat{P}_i|\psi\rangle$.
- After measurement, the remaining quantum state is the normalised state:

$$\frac{\hat{P}_i|\psi\rangle}{\sqrt{\langle\psi|\hat{P}_i|\psi\rangle}}. \tag{2.16}$$

This process is generally described as collapse of the wavefunction, although the exact mechanism (and the appropriate ontological description) vary according to interpretations of quantum mechanics.

This behaviour is a postulate of quantum mechanics, although it can be generalised further than we have described here. In particular, we can notice the Born rule: $P(\lambda_i) = \langle v_i|\psi\rangle^2$, which is the square of the coefficient multiplying the component of $|\psi\rangle$ along that eigenstate. As the eigenvalues are real, these correspond to observables, or measurable quantities which take real values. We may also find it useful to use the closely related formalism of POVMs (positive-operator valued measurements). In this case, the measurement operators corresponding to outcome m are given by $\hat{M}_m$, and the probability of outcome m is given by $\langle\psi|\hat{M}_m^\dagger\hat{M}_m|\psi\rangle$. We therefore require $\sum_m \hat{M}_m^\dagger\hat{M}_m = \mathbb{1}$, however, $\hat{M}_m$ may not necessarily be a projector – that is, $\mathrm{tr}\left(\hat{M}_m^\dagger\hat{M}_m\right) \leq 1$.

When measuring a classical system, the precision with which you are able to describe the system depends on your measurement apparatus and method, and in theory can be arbitrarily high. However, the possible precision of measurement is more complicated in our description of particles in quantum theory.

We introduce the commutator of two matrices, or operators,

$$[A, B] = AB - BA, \tag{2.17}$$

and, in classical mechanics, the Poisson bracket,

$$\{A(\mathbf{x},\mathbf{p}), B(\mathbf{x},\mathbf{p})\} = \sum_i \left(\frac{\partial A}{\partial x_i}\frac{\partial B}{\partial p_i} - \frac{\partial A}{\partial x_i}\frac{\partial B}{\partial p_i}\right), \tag{2.18}$$

where x_i and p_i are the position and momentum co-ordinates of the bodies present in the system, respectively[4]. Dirac proposed the canonical commutation relations, meaning that the Poisson bracket in classical mechanics can be replaced in quantum mechanics:

$$\{A, B\} \to \frac{[A, B]}{i\hbar}. \tag{2.19}$$

[4]Note that this notation is also sometimes used for the anticommutator in quantum mechanics, $\{A, B\} = AB + BA$.

This relationship cannot necessarily be derived, as it is again a postulate of quantum mechanics, however we can justify it by considering the Schrödinger equation, and in particular this arises from comparing time evolution in the Heisenberg picture to what we expect to see in the classical limit.

In particular, we consider canonically conjugate pairs of observables, which satisfy $[\hat{A}, \hat{B}] = i\hbar$, generally linked by the Fourier transform, such as position and momentum. These satisfy the Heisenberg uncertainty theorem:

$$\Delta\hat{A}\Delta\hat{B} \geq \hbar/2, \tag{2.20}$$

where $\Delta\hat{A}$ represents the uncertainty (or standard deviation) in the measurement of A. Given that the expected value of $\hat{A}$ (when measuring on state $|\psi\rangle$ is $\langle\hat{A}\rangle = \langle\psi|\,\hat{A}\,|\psi\rangle$), this is defined as:

$$\Delta\hat{A} = \sqrt{\langle\hat{A}^2\rangle - \langle\hat{A}\rangle^2}. \tag{2.21}$$

Therefore, there is an upper limit to the precision with which it is possible to know about the position and momentum (or other conjugate observables) of a particle simultaneously. In the case of commutating observables, they can be simultaneously diagonalised, so there is an eigenbasis in which you can measure both concurrently.

This can be derived from the commutation relations, in fact we find that $\Delta\hat{A}\Delta\hat{B} \geq \frac{1}{2}|\langle[\hat{A}, \hat{B}]\rangle|$ (see [38] for further details).

It can also be justified by considering Gaussian wave packets. Briefly: consider a wave propagating in free space, described by $\psi(\mathbf{x}, t) = A\exp(i(\mathbf{k}\cdot\mathbf{x} - \omega(k)t))$. We find the energy and momentum using $E = i\hbar\partial_t$ and $\mathbf{p} = -i\hbar\nabla$. In order to allow for normalisation, we must modulate the wavepacket by a Gaussian envelope. By considering the Fourier transform, we see that the product of widths is $\Delta x\Delta p = \hbar/2$. Thus, the limit on how well-defined these physical quantities can be is linked to its complementary variable. Full details of this are given in [38].

We are frequently interested in multi-particle systems[5]. In a system of identical particles, whose wavefunctions overlap, the individual particles cannot be distinguished and we describe the behaviour of the ensemble with a single state vector. If we exchange two identical particles (say with an operator $\hat{P}$), this does not change the physical state of the system, and so the multi-particle state vector must be an eigenstate, and the only action of $\hat{P}$ is to impart a global phase. Given that $\hat{P}^2$ returns the system to its initial state, we expect the eigenvalues to be ± 1, that is the wavefunctions are either unchanged (symmetric under interchange) or multiplied by -1 (antisymmetric under interchange). Thus far, these classes have been sufficient to describe all observed systems of (three-dimensional) identical particles, and

[5] Here, we present a brief overview of an argument for the spin-statistics theorem, although we note this does not have the required level of rigour for a proof of the theorem, nor does it include important details such as spin, or relativistic descriptions of particles.

this in fact depends on spin; particles are either bosons, which have completely symmetric wavefunctions, or fermions, which have completely antisymmetric wavefunctions. More exotic quantum statistics, such as anyons (which have properties different from bosons and fermions – for example, exchanging two particles twice changes their wavefunction) have been considered, but not yet observed in three dimensions [54]. This thesis focuses on the behaviour of photons, which are bosons. The relationship between spin and exchange statistics arises as a result of relativistic quantum mechanics, which we have not described here.

The ability to store and process information with quantum states is useful in many areas, including secure communications [55, 56] and metrology (the science of measurement) [57], as well as being interesting in its own right. In this thesis we will focus on quantum computing, using the formalism we have just described, with a brief introduction given in the following section.

2.2 Quantum computing

Quantum computers are computers built from systems that behave according to quantum mechanics. Therefore, the mathematics used to describe and program them is fundamentally different to any classical computer. Quantum computers allow us to access a greater variety of operations, and the ability to design new algorithms utilising these. Quantum states, in general, are not necessarily able to store or send more information than classical states; due to the collapse mechanism, they are only able to output classical information. Despite this, we expect quantum algorithms to be more powerful than classical algorithms. We access an exponentially larger Hilbert space, and can use the interference of complex-valued amplitudes to introduce new algorithms, composed of a different set of elementary gates. However, the interference effect is an important aspect of quantum algorithms - we cannot just think of a quantum computer as 'trying all of the solutions at once'. In this way we can compare it to an NTM, which only requires that one of the paths succeed, whereas, although a quantum computer may not be deterministic, we require a certain probability of success.

Much like classical computation can be decomposed into electrical circuits, the circuit model of quantum computation decomposes unitary transformations into one- or two-qubit gates (which are themselves smaller unitary matrices). In fact, constructing an arbitrary unitary, to an arbitrary level of precision, can be done efficiently using gates from a finite set [58], such as the Clifford group + the T gate[6] [59] – this is known as a universal gate set. An interesting result is that we find that any circuit consisting of only Clifford gates can be efficiently simulated by a classical computer, a result known as the Gottesman-Knill theorem [60]. Therefore, the number of T gates required by a circuit is of particular

[6]The definitions of quantum gates are omitted here, as they are not used in the rest of the thesis, but can be found in [52].

importance, especially when we note that these are typically more difficult to implement in error correction.

The logical operators that make up a circuit in classical computing aren't necessarily unitary e.g. the ADD operation, as they are not reversible. However, these can be made to be unitary by adding an ancilla qubit, which stores the output of the gate. It can be shown that quantum computers are at least as powerful as classical computers (in computational complexity terms), that is, $\mathbf{BPP} \subseteq \mathbf{BQP}$.

Recall the ECT thesis: that any reasonable, physically realisable model of computation (including quantum computing) can be efficiently simulated by a probabilistic Turing machine. In the case that **BQP** is not contained in **P** or **BPP**, then this is violated. There is an extension to Turing machines proposed, including a 'quantum Turing machine' [61], but this is outside of the intended meaning of the ECT thesis. The complexity of quantum computation can be evaluated using circuit complexity (by counting the gates or otherwise determining the size of the circuit required), by counting the number of qubits required (e.g. when considering communication complexity), or by using query complexity, the amount of calls required to an oracle/black-box function. The latter is particularly useful as it is generally easier to calculate, but can be an oversimplification when considering how to realistically solve a problem. Early evidence for this came from the Deutsch-Josza algorithm [62], which solves a problem with a single oracle call, whereas $2^{n-1}+1$ are required for a classical solution. Nonetheless, this is for a very esoteric problem and is not expected to have useful applications, and a probabilistic classical algorithm is almost as effective. Simon's algorithm [63], however, shows that $\mathbf{BQP} \not\subseteq \mathbf{BPP}$ with access to an oracle.

A significantly more exciting, well-known quantum algorithm – based on a similar principle to Simon's algorithm, period-finding – is Shor's algorithm for factorisation, which also provides an exponential speedup [64]. This gives a polynomial-time quantum algorithm for a problem which is not thought to be in **BPP**, and has real-world relevance, particularly in cryptography. This provided more definitive evidence against the ECT thesis, and further developments in quantum algorithms have followed.

Another flagship quantum algorithm is Grover's search algorithm. This provides a quadratic speedup against the best possible classical algorithms, however it is particularly useful as a subroutine for many other algorithms. There is also the HHL algorithm, for inverting matrices (although recent work on 'dequantisation' has called into question the advantage of this [65]) and many others which will not be covered here.

Simulation of real-world quantum phenomena, such as the behaviour of molecules or in materials science, is generally beyond the scope of classical computers, due to the complexity of the systems, but it is a particularly relevant use-case for quantum computers, and it was the basis of their original proposal [66]. Much like early classical computers (as described in Section 2.1.1) this can be analogue simulation, where the Hamiltonian of a physical system is

engineered to approximate the system of interest [67], or implemented on universal quantum computers using digital simulation algorithms [68].

Nonetheless, this can still cause some issues, as implementing the time evolution of a particular Hamiltonian in a digital system requires Trotterisation, which can be inefficient to realise [69].

Despite this progress, a few skeptics contend that a convincing experimental demonstration is needed to definitively refute the ECT thesis, which is still lacking.

Another important aspect of quantum computing is quantum error correction. Redundancy is needed in the encoding in order to preserve the delicate coherence of quantum systems against errors. However, the no-cloning theorem of quantum information states that it is impossible to create a copy of an arbitrary quantum state. Therefore clever encoding schemes must be used that allow computations to be carried out without the error growing to catastrophic levels. This makes use of the threshold theorem, which asserts that as long as the initial qubit error is sufficiently low, the error in the calculation can be suppressed arbitrarily, for arbitrarily long computations, so the computation is still scalable [70]. This introduces significant overheads, which is a major hurdle in implementing quantum algorithms at a meaningful scale: the need for many qubits, with low error rates.

Current state-of-the-art devices, which cannot run error-corrected algorithms beyond what is capable for classical computers, are generally seen as belonging to the NISQ (noisy intermediate-scale quantum) era [71]. These frequently make use of different algorithms, which combine classical and quantum computing power [72].

In understanding the usefulness of near-term quantum technologies, we need to compare them to classical computation. *Quantum advantage* (originally known as quantum (computational) supremacy, a phrase coined by Preskill in 2012 [73]) refers to the achievement of quantum computers carrying out tasks that would be infeasible for classical computers, generally because it would require computational power or time beyond what is possible or available, e.g. computation times in the thousands of years. This is difficult to verify with certainty, and quantum advantage claims are often later debunked (e.g. the quantum advantage result of [74], later debunked by [75]), however we are currently in an era of increasingly convincing demonstrations of quantum computational power beyond classical limits.

Although quantum advantage refers to the exact runtime of a specific task, it is widely understood that, due to the high overheads of quantum error correction, the difficulties of constructing the hardware, and the relatively slow speed or high resource requirements per qubit, this is only likely to be possible for algorithms (or processes) which offer a better-than-quadratic, or ideally exponential, speedup in comparison to the classical case [76]. These are the examples which are driving the majority of the interest in quantum computing. There is still an open, but important, question on the extent to which quantum computers can present more energy efficient calculation methods, a main focus of the Quantum Energy

Initiative [77].

Demonstrations of quantum advantage are frequently based on sampling problems [78, 79]. Generally, a quantum algorithm will produce a particular outcome representing a solution to a problem, and running the algorithm multiple times is useful to reduce the likelihood of error. The circuit acts to produce a quantum state that, upon measurement, gives the desired outcome with high probability. This process is equivalent to sampling from the probability distribution that corresponds to the probability of measuring all possible outcomes.

In the case of random sampling problems, the goal is not to produce a specific output upon measurement. Every measurement instead produces an outcome with probability defined by the underlying distribution. Various sampling problems based on randomised inputs are used as schemes for quantum advantage attempts, as it has been shown that it is intractable for a classical computer to sample from the resulting distributions. These often involve non-universal models of computation, that cannot produce arbitrary unitary transforms and therefore cannot run most quantum algorithms, but may be easier to implement experimentally, and they usually do not use error correction. Examples include sampling from the outcomes of random quantum circuits, either from a universal gate set (we note that in this case the power of the scheme may be limited by the available depth of the circuit), or a restricted gate set (in the case of IQP, instantaneous quantum polynomial-time circuits [80]). A further pedagogical introduction to sampling problems is given in [81].

This thesis is interested primarily in variations of *BosonSampling* (or boson sampling, BS), a sampling scheme for quantum linear optics (although it can be carried out by other platforms [82, 83]) [84]. As we will see, it may be more appropriate to compare this model to analogue simulators than to computers. Due to the complexity of classically modelling the dynamics describing this system, it is of significant interest in quantum information science, and in particular in its potential for quantum advantage demonstrations.

There are many different proposed platforms (i.e. the systems on which to base qubits) in which quantum algorithms can be realised, such as trapped ions [85], or superconducting circuits [86]. The requirements for a potential qubit implementation are summarised in the DiVincenzo criteria [87]. In this thesis we will be most interested in linear optics, in which single photons are used as quantum information carriers.

2.3 Quantum optics

In order to understand how light can be used to program quantum information, we now consider the quantum treatment of light.

By light, we are referring to electromagnetic radiation, which, in a vacuum, is described by Maxwell's equations, the culmination of many years' theoretical and experimental work

describing light, electricity and magnetism [88]:

$$\begin{aligned} \nabla \times \mathbf{E} &= -\frac{\partial \mathbf{B}}{\partial t} \\ \nabla \times \mathbf{B} &= \mu_0 \varepsilon_0 \frac{\partial \mathbf{E}}{\partial t} + \mu_0 \mathbf{J} \\ \nabla \cdot \mathbf{E} &= \rho/\varepsilon_0 \\ \nabla \cdot \mathbf{B} &= 0. \end{aligned} \tag{2.22}$$

$\mathbf{E}$ and $\mathbf{B}$ are the electric and magnetic vector fields respectively (with implied dependence on position and time). These depend on electrical sources which provide charge density ρ and current density $\mathbf{J}$ (which we take to be zero in the following description), and universal constants: $\varepsilon_0 = 8.854 \times 10^{-12}\,\mathrm{C\,V^{-1}\,m^{-1}}$ and $\mu_0 = 1.257 \times 10^{-6}\,\mathrm{N\,A^{-2}}$, the permittivity and permeability of free space, which satisfy $(\mu_0\varepsilon_0)^{-1/2} = c$. Note that this is not the same ε_0 as was introduced in Section 2.1.3. These equations are also the foundation for relativistic electrodynamics. The effects of special relativity can be safely ignored for the purposes of this thesis, and there is ongoing research to find a theory that effectively unifies general relativity and quantum mechanics.

We can consider the solution to these equations as a basis for *quantising* the electromagnetic field. Full details of this are omitted here, but we follow the treatment of [89, 90], as well as the more pedagogical description in [91].

When considering differential equations, it is necessary to introduce boundary conditions to arrive at a unique solution. When introducing the quantisation of the electromagnetic field, this is generally done by considering a cavity (or space) of finite volume. This is in accordance with the cavity description of a black-body emitter which was first considered by Planck. This is an approximation that will generally work for our purposes, but breaks down for certain systems, such as cavity quantum electrodynamics. However it applies well to many optical elements [92]. This allows us to find solutions in terms of a vector potential $\mathbf{A}$, where $\mathbf{B} = \nabla \times \mathbf{A}$ and $\mathbf{E} = -\frac{\partial \mathbf{A}}{\partial t}$. We then find that[7]:

$$\mathbf{A}(\mathbf{x}, t) = \sum_{\mathbf{k}} \sum_{i=1,2} \mathbf{e}_{\mathbf{k}i} (A_{\mathbf{k},i} \exp(-i\omega_k t + i\mathbf{k}\cdot\mathbf{x}) + A^*_{\mathbf{k},i} \exp(i\omega_k t - i\mathbf{k}\cdot\mathbf{x})). \tag{2.23}$$

The index i indicates the polarisation, where the unit vectors $\mathbf{e}$ are mutually perpendicular, and to $\mathbf{k}$, which indicates the direction of propagation of the wave. The components of $\mathbf{k}$ are:

$$k_{x,y,z} = 2\pi n_{x,y,z}/L, \; n \in \mathbb{Z}. \tag{2.24}$$

Here the quantisation cavity considered is a cubic space of length L, which leads to solutions expressed in terms of standing waves, and $\omega_k = ck$, the mode angular frequency. The different labels n indicate the different optical modes of the system; a different basis of

[7] Here, we use the Coulomb gauge, which imposes that $\nabla \cdot \mathbf{A} = 0$.

modes can instead be chosen [93]. The modes represent an orthonormal basis over which we express the different states. By rearranging, we can remove the dependence on L and instead consider the form of the mode vectors in terms of the wavelength of the light, although further techniques are needed to find this wavelength, which are beyond the scope of this work.

We use this solution to find expressions for the electric and magnetic fields, $\mathbf{E} = -\frac{\partial \mathbf{A}}{\partial t}$ and $\mathbf{B} = \nabla \times \mathbf{A}$, which can be used to find the Hamiltonian of the electromagnetic radiation, representing the total energy:

$$\begin{aligned} H_{\text{EM}} &= \frac{1}{2}\int_V \left(\varepsilon_0 |\mathbf{E}|^2 + \mu_0^{-1}|\mathbf{B}|^2\right) \mathrm{d}\mathbf{x} \\ &= \sum_{\mathbf{k},i} \varepsilon_0 V \omega_k^2 (A_{\mathbf{k},i} A^*_{\mathbf{k},i} + A^*_{\mathbf{k},i} A_{\mathbf{k},i}) \end{aligned} \tag{2.25}$$

where V is the volume of the cavity. This is the sum of the energy density of the electric and magnetic components of the field, and comes directly from the way that the fields are defined.

As discussed in Section 2.1.3, the quantum treatment for photon sources derives from considering the analogy to harmonic oscillators of some mass m (this can be found from the de Broglie relation using the wavelength of light) and angular frequency ω, whose quantum mechanical Hamiltonian is given by:

$$\hat{H}_{\text{SHO}} = \frac{\hat{p}^2}{2m} + \frac{1}{2} m\omega^2 \hat{x}^2. \tag{2.26}$$

This derives from the application of Dirac's canonical commutation relations to the classical Hamiltonian. Instead of the position and momentum operators $\hat{x}$ and $\hat{p}$, we can express the Hamiltonian in terms of the 'ladder' operator, defined as $\hat{a} = (2m\hbar\omega)^{-1/2}(m\omega\hat{x} + i\hat{p})$. Where did this come from, and in particular why did we introduce $\hbar$? This allows us to have the correct value of the energy of each photon, of $E = hf$, following Planck's law. It may seem a bit circular, but bear in mind that this is not a strict mathematical derivation, and must agree with empirical evidence and previously known results. We can then write the Hamiltonian as:

$$\begin{aligned} \hat{H}_{\text{SHO}} &= \frac{1}{2}\hbar\omega(\hat{a}\hat{a}^\dagger + \hat{a}^\dagger\hat{a}) \\ &= \hbar\omega\left(\hat{a}^\dagger\hat{a} + \frac{1}{2}\right), \end{aligned} \tag{2.27}$$

noting that $[\hat{a}, \hat{a}^\dagger] = 1$.

We now compare this to Eq. 2.25. Rewriting the Hamiltonian (Eq. 2.26) in terms of the dimensionless amplitudes $a_{\mathbf{k},i} = (2\omega_k\varepsilon_0 V)^{1/2}\hbar^{-1/2}A_{\mathbf{k},i}$, and applying the canonical quantisation described (so that $a_{\mathbf{k},i} \to \hat{a}_{\mathbf{k},i}$), we now have:

$$\hat{H} = \sum_{\mathbf{k},i} \hbar\omega\left(\hat{a}^\dagger_{\mathbf{k},i}\hat{a}_{\mathbf{k},i} + \frac{1}{2}\right). \tag{2.28}$$

That is, we consider the electromagnetic system to be a set of modes, each of which is described by a harmonic oscillator, which is associated with an angular frequency ω, and a mass m, which determines the wavelength of the light emitted by that oscillator. We note that each oscillator has a zero-point energy. The effect of introducing the boundary conditions gives us discrete modes, and ensures we don't have infinite energy of the vacuum. In theories that require free space representations, we must instead use renormalisation.

As previously described, the eigenvalues of $\hat{H}$ give the possible energies of the system, which we see are the eigenstates of $\hat{a}^\dagger\hat{a}$. Let us label one such eigenstate $|n\rangle$, so $\hat{H}|n\rangle = E_n|n\rangle$. We can also show that $\hat{a}^\dagger|n\rangle$ is also an eigenstate of $\hat{H}$ with eigenvalue $E_n + \hbar\omega$, and $\hat{a}|n\rangle$ is an eigenstate of $\hat{H}$ with eigenvalue $E_n - \hbar\omega$. I've omitted these derivations because they are quite familiar from textbooks – see, for example, section 4.3 of [89].

We define the ground state, or vacuum, as the lowest energy level, $|0\rangle$, so $\hat{a}|0\rangle = 0$. We thus find the energy eigenstates are equally spaced with $E_n = (n+\frac{1}{2})\hbar\omega$, $n \in \mathbb{Z}_+ \cup \{0\}$. These eigenstates represent the Fock states, which represent different numbers of excitations of the fields (i.e. photons). These are eigenstates of the number operator: $\hat{a}^\dagger\hat{a}|n\rangle = n|n\rangle$. We can then convert between these using:

$$\hat{a}|n\rangle = \sqrt{n}\,|n-1\rangle \tag{2.29}$$

$$\hat{a}^\dagger|n\rangle = \sqrt{n+1}\,|n+1\rangle \tag{2.30}$$

to increase (using the creation operator, $\hat{a}^\dagger$) or decrease (using the annihilation operator, $\hat{a}$) the photon number, where we also note the eigenstates have been introduced to satisfy normalisation. Therefore:

$$|n\rangle = \frac{(\hat{a}^\dagger)^n}{\sqrt{n!}}|0\rangle\,. \tag{2.31}$$

We note that, due to the uncertainty principle, states cannot have an exactly defined photon number but there can be minimum uncertainty states. Fock states may seem well defined within this framework but, when considered in the realistic infinite-dimensional Hilbert space, they cannot be normalised. Nonetheless, for our purposes we can continue to consider them, although we note they cannot physically be produced exactly.

Much like the wave picture of light, photons have a certain frequency, the conjugate variable to time. It is difficult to compare the propagation of a photon to the oscillation of a field, but it is interesting to note that photons have phase, which changes as photons propagate.

I recommend the theses [94, 95] for the following section. We will be considering light propagating in materials where the Hamiltonians describing quantum optical systems are at most quadratic in the creation and annihilation operators, and take the form [89, 96, 97]:

$$\hat{H} = \sum_k \left(\alpha_k\hat{a}_k^\dagger + \alpha_k^*\hat{a}_k\right) + \sum_{k,l}\left(\xi_{kl}\hat{a}_k^\dagger\hat{a}_l^\dagger + \xi_{kl}^*\hat{a}_l\hat{a}_k\right) + \sum_{k,l}\left(\mu_{kl}\hat{a}_k^\dagger\hat{a}_l + \mu_{kl}^*\hat{a}_l^\dagger\hat{a}_k\right). \tag{2.32}$$

This is due to the nature of the materials in which photons are generated and propagated, and is beyond the scope of this thesis, although it is discussed in many of the theses that have been referenced. Higher order interactions are possible, but they are rare and difficult to engineer. (From this point onwards we will use k to label distinct modes.) These generate a particular class of unitaries, known as Bogoliubov transformations. The first two sums – containing terms which are linear, and quadratic, in the ladder operators – represent displacement and squeezing, respectively, which we will return to in Section 2.7; these are used in photon generation. The third term conserves the total photon number(as every annihilation operator is matched with a creation operator), representing operations which are known as passive transformations. To describe these it will be useful to consider state transformation in the Heisenberg picture.

A Fock state over m modes, $|\mathbf{n}\rangle = |n_1, ..., n_m\rangle$ can be expressed as:

$$|\mathbf{n}\rangle = \prod_{i=1}^{m} \frac{(\hat{a_i}^\dagger)^{n_i}}{\sqrt{n_i!}} |0\rangle \,, \tag{2.33}$$

and therefore it is useful to consider transformations between Fock states in terms of their action on the creation operators.

Let us consider a vector of mode operators $\hat{\mathbf{a}} = (\hat{a}_1, ..., \hat{a}_m)^T$, $\hat{\mathbf{a}}^\dagger = (\hat{a}_1^\dagger, ..., \hat{a}_m^\dagger)$. Then the vector $\hat{\mathbf{a}}^\dagger$ represents the creation operators of photons emitted into arbitrary modes (represented by the subscripts). The Bogoliubov transformations act as:

$$(\hat{\mathbf{a}}^\dagger, \hat{\mathbf{a}})^T \mapsto (\hat{\mathbf{b}}^\dagger, \hat{\mathbf{b}})^T = U_B^T (\hat{\mathbf{a}}^\dagger, \hat{\mathbf{a}})^T. \tag{2.34}$$

The superscript T, indicating the transpose, is used by convention.

We also note that in the Heisenberg transformation, each individual $\hat{a}_k^\dagger$ is mapped to $\hat{U}_H \hat{a}_k^\dagger \hat{U}_H^\dagger = \sum_l c_l \hat{a}_l^\dagger$, however rather than writing each of these individual transforms, as this acts linearly we can express it as a matrix on the basis of creation operators, U_B. We use the transpose here as we are acting on the Hermitian conjugate.

The Hamiltonian in Eq. 2.32 generates the more general Bogoliubov transformations:

$$\hat{a}_k^\dagger \mapsto \sum_l \alpha_{kl} \hat{a}_l^\dagger + \beta_{kl} \hat{a}_l + \gamma_k. \tag{2.35}$$

Unlike $\hat{U}_H$, the matrix U_B is not necessarily required to be unitary (by the same reasoning). More generally, we are interested in the group of symplectic matrices that describe the transformations acting on the vector of ladder operators. This will be discussed further in Section 2.7, but in this case, U_B is indeed unitary (although the full derivation of this is omitted here – see, e.g. [96]). It is also interesting to note that this arises from the commutation relations, and is boson-specific – the case would be different for fermions.

It is important to note the linearity of this transformation, which in this case means that the quantum evolution does not depend on the photon number. Photons are therefore often

described as 'non-interacting'. This means the evolution of an arbitrary Fock state proceeds as:

$$
\begin{aligned}
|\mathbf{n}\rangle &= \prod_{i=1}^{m} \frac{(\hat{a}_i^\dagger)^{n_i}}{\sqrt{n_i!}} |0\rangle \\
&\mapsto \prod_{i=1}^{m} \frac{\left(\sum_{j=1}^{m} (U_B)_{ij} \hat{a}_j^\dagger\right)^{n_i}}{\sqrt{n_i!}} |0\rangle \,.
\end{aligned} \tag{2.36}
$$

Hence, if we consider the Fock basis $|\mathbf{n}\rangle$, we cannot produce an arbitrary transformation in this basis, as we will discuss further in the following section.

We note that this does not preclude photon interference, such as the HOM (Hong-Ou-Mandel) dip [98]. The combination of terms after doing this product may lead to cancellations (destructive interference), due to the commutation of creation operators.

Any quadratic Hamiltonian, as given by Eq. 2.32, may be diagonalised to give Bogoliubov operators which implement a linear transformation of the mode operators (due to Bogoliubov diagonalisation), and therefore the complete system may sometimes be referred to as linear optics. However, we may also use the term 'linear optics' to refer to only the passive transformations that preserve photon number (the third term of Eq. 2.32). In particular, displacement and squeezing, which are represented by the other terms (and will be discussed further later on) are generated by the non-linear electro-optic response of materials (such as silicon or silicon nitride), which are usually pumped by a source of light (which is omitted in the terms of the Hamiltonian) [89].

2.4 Linear optical quantum computing

Let us now consider how we would use photons to construct a quantum computer. Photons have several differences to other popular platforms. Measuring any quantum state destroys quantum coherence and collapses to an eigenstate, but photon measurements absorb them and therefore also consume the physical qubit. Photons are sometimes referred to as 'flying qubits', as they can encode information and travel long distances, with transformations carried out when they are acted on by successive optical elements. The lifetime of photons is limited by the path length available in the device, as well as their susceptibility to loss, so quantum computing implementations usually involve procedures to pass information between qubits (i.e. variations on teleportation).

Typical evaluations of qubit technologies focus on qubit coherence times (coherence will be (briefly) discussed further in Section 2.7), however a major bottleneck for constructing photonic quantum computers is loss, and therefore the lifetime of the photon should be minimised to reduce the probability that it does not survive its journey. People also often ask 'how many qubits does this quantum computer have?' – however, in measurement based

quantum computing, there are several ways to define a qubit, so this question requires some caution.

There are several different ways of encoding quantum information into photons, i.e. to represent logical states (qubits) in terms of physical states. One potential encoding is using Fock states (i.e. that the vacuum should represent the logical state $|0\rangle_L$, and that one photon in a mode represents the state $|1\rangle_L$), however using this, it is difficult to tell when a photon has been lost. The dual-rail encoding instead uses orthonormal modes, such as polarisation or spatial modes (that is, photons occupying e.g. distinct waveguides), with the encoding $|0\rangle_L = \hat{a}_1^\dagger |\text{vac}\rangle = |1,0\rangle$, $|1\rangle_L = \hat{a}_2^\dagger |\text{vac}\rangle = |0,1\rangle$ [96].

In general, the Hilbert space occupied by n photons in m modes has dimension $\binom{m+n-1}{n}$. Where does this come from? It would be tempting to use m^n, that is, the set of subsets of length n from a list of length m with replacement, however, this would suggest that the photons can be distinguished. This is actually the 'multichoose' function, found using the 'stars and bars' counting argument [99]. This follows the following argument: we have a set of n photons, and we want to partition them into m sets (although we note that $m \geq n$ so some sets are empty). Hence, we use $m-1$ 'bars' as separators denoting the dividing point between the sets. As this produces a list of $n+m-1$ objects, there are $n+m-1$ different places to put the bars, so the total number of different partitions is $\binom{m+n-1}{n}$. There are good illustrations on Wikipedia! If we use the dual-rail encoding, we put n photons in $2n$ modes, giving a total size of the Hilbert space 2^n. We note that $2^n \leq \binom{3n-1}{n}$, however, some Fock states are outside of our computational space, as we require exactly one photon in each pair of modes.

It could be tempting to instead use the encoding where each mode is labelled by a different bit string, with one photon occupying all 2^n modes. This is not a scalable implementation, as it requires an exponential number of modes, however we can do universal unitaries in this space. Qudit encodings (one photon to 3 or more modes) can be useful in some cases – particularly if it is more practical to increase the number of modes than increase the number of photons [100].

A universal *passive* unitary transformation can be expressed as $\hat{\mathbf{a}}^\dagger \mapsto U^T \hat{\mathbf{a}}^\dagger$, (i.e. not mixing creation and annihilation operators). These can be implemented in a linear optical network using an interferometer, which implements specific transformations between modes. The components used to practically implement this are beam splitters and phase shifters, and an arbitrary unitary can be decomposed into the appropriate interferometer using a Reck scheme [101], or Clements scheme [102]. Further details on experimental implementations of unitary transformations are omitted as they are not relevant to this work.

Let's think a little more about this unitary. Each operator $\hat{a}$ is itself a matrix inside an infinite dimensional Hilbert space, and we know that Heisenberg-picture transformations on these operators should be unitary. However, the matrix (note that this doesn't have a 'hat'

because it is not itself an operator, but matrix of operator transformations, or superoperator) that takes a vector of these operators/matrices to a different vector itself happens to be unitary. This is because of the form of the Bogoliubov operators, and not because it is generated by Hamiltonians in the same way as time evolution. In fact, we will see the symplectic generalisation later on. The fact that we can construct universal unitaries in linear optics is detailed in [95], but, to emphasise one of many times: this is not a universal quantum computer!

We are now using notation in a few different ways to refer to different quantum states, and it is important to keep track of what is being represented by bras and kets. In particular, describing Fock states (as we have been doing so far) is an example of second quantisation, where the formalism focuses on the occupation of different modes, e.g. the state $|a,b,c\rangle$ represents a photons in the first mode, b photons in the second mode, and c photons in the third mode. Alternatively, we can use the first quantisation, which focuses on the state of individual particles, so $|a,b,c\rangle$ represents that the first photon is in mode a, the second photon is in mode b, and the third photon is in mode c.

Due to the exchange statistics of bosons, care is needed when converting between these formalisms [103, 104]. For example, the second quantised state $|1,1\rangle = \hat{a}_0^\dagger \hat{a}_1^\dagger |\text{vac}\rangle$ can correspond to $|0,1\rangle$ or $|1,0\rangle$ in the first quantisation. However, it's not necessarily a case of the second quantised state being equivalent to either of these – these states cannot be distinguished, and it is necessary to construct the perfectly symmetric state $\frac{1}{\sqrt{2}}(|0,1\rangle + |1,0\rangle)$ as the first quantised equivalent of this state. This reflects the fact that the first quantisation formalism is redundant, and hence states which are given different labels are physically identical; it may therefore seem that the Fock space is smaller than for other encodings.

We also note an alternative description, in which states are described instead using multivariate polynomials [84], similarly to Eq. 2.33.

However, this construction is not yet sufficient to build a universal quantum computer – that is, to implement an arbitrary unitary on n qubits (i.e. an arbitrary unitary on the logical states $|0\rangle_L$ and $|1\rangle_L$) in the dual-rail encoding. As we have seen, passive linear interferometers act on Fock states according to a product of unitary transformations on the creation operators. However, this is a subgroup of the full space of unitaries that we would like to access. Let us consider the example from [97], the transformation on the logical qubits given by $|00\rangle_L \to \frac{1}{\sqrt{2}}(|00\rangle_L + |11\rangle_L)$. Using the dual-rail encoding, the transformation on creation operators is $\hat{a}_1^\dagger \hat{a}_3^\dagger \to \frac{1}{\sqrt{2}}(\hat{a}_1^\dagger \hat{a}_3^\dagger + \hat{a}_2^\dagger \hat{a}_4^\dagger)$. We can see that this cannot be decomposed into a product of single-photon transformations, and therefore cannot be implemented with the passive transformations we have considered so far.

Single photons input to a passive linear interferometer is not sufficient for scalable universal quantum computation, in particular due to the inability to deterministically produce entanglement (at least in the dual-rail encoding, as the previous example shows), hence the

need to introduce nonlinearities. One method of doing this is to introduce measurement and feedforward – the process by which results of measurements at one stage of the computation are used to decide the operations that are implemented at a later stage.

We can see an example of how this is implemented by considering, e.g. the fusion gate [105] (or see [91, 106]). Type-II fusion can be seen as a Bell state measurement, which has some probability of failure, as certain outcomes correspond to states outside of the computational space. However, on success, some photons are projected into a Bell state, which effectively entangles them. These photons are destroys, but if they are entangled to other photons in small initial resource states, the remaining photons can be built up into larger cluster states.

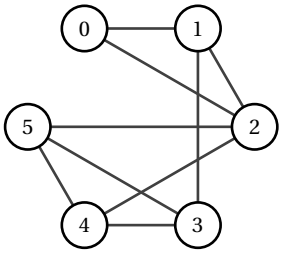
Historically, it was thought that the only way to introduce the necessary nonlinearities would be to use non-linear materials [52]. The 'KLM' (Knill, Laflamme and Milburn) scheme proposed in 2001 showed that it is possible to scalably implement a quantum computer using single-photon sources and detection, passive interferometers, and measurement and feedforward [107]. Scalable here means that the required components scale at worst polynomially with the number of qubits or gates, although KLM requires linear resources, similarly to our definition of 'efficient'.

Nonetheless, KLM still required large overheads. It was the proposal of measurement-based quantum computing (MBQC) that provided a realistic picture for how to build photonic quantum computers [108]. This is a scheme that begins with the formation of a large entangled 'cluster' state, and then requires no further two-qubit operations, just single-qubit measurements and feedforward. Producing the cluster state is itself complicated, but suggested schemes involve percolation, which uses feedforward on a much larger state to deal with the probabilistic failure of fusion gates [109]. KLM and MQBC are not entirely unrelated [110].

Another promising scheme that has recently proposed is fusion-based quantum computation [111]. This provides a particularly elegant solution to the issue of building up large cluster states by combining the entangling and operational measurements into one stage, and dealing with fusion failure through the error correction scheme.

Errors in photonic quantum computing can come from several sources. The primary concern is photon loss, which in integrated optics in caused by photons being scattered out or absorbed by the waveguides. Distinguishability between photons (caused, e.g. by slightly different spectra or polarisations) also decreases quantum interference [103], as does spectral impurity, which will be considered in more detail later on.

Throughout this work, we will not be focused on operations in the qubit picture, but instead will focus on passive unitary transformations that act on modes. In order to do this, it will be useful to consider the graphical representation of states.



$$\begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

FIGURE 2.1. An example 6-vertex graph and its associated adjacency matrix, where a 1 in the i, j-th element indicates an edge connecting vertices i and j.

2.5 Graphs

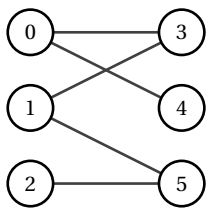
A graph is a mathematical structure which, in its most basic form, is made up of a set of vertices V (with $|V|$ members) and a set of edges E (with $|E|$ members) connecting them [112]. Graph theory is an important field of mathematics, and is used to model the relationships within groups of objects in many different subjects. We introduce them here as they will become useful in subsequent sections for describing certain quantum states.

Graphs can be *directed*, in which edges have a direction, beginning at one vertex and terminating at another. They can also be *weighted*, in which edges (or vertices) can have a value assigned to them. For now, we will focus on undirected, unweighted graphs.

A graph of n vertices can be described by an $n \times n$ adjacency matrix, $\mathcal{A}$. For an undirected, unweighted graph, this is a symmetric matrix in which element $\mathcal{A}_{ij}$ is 1 if vertices i and j are connected by an edge, and 0 otherwise, such as in Fig. 2.1.

Let us consider a particular class of graphs, bipartite graphs, with an example shown in Fig. 2.2. In this case, the vertices of the graph can be separated into two sets, V_1 and V_2. (This could, of course, be generalised further to multipartite graphs, but that will not be necessary in this work.) All edges of the graph connect a vertex in V_1 to a vertex in V_2, but there are no internal edges connecting vertices in the same set. In this case, the $|V|\times|V|$ adjacency matrix representing the graph will be block anti-diagonal. Due to the structure of bipartite graphs, we can instead use an alternative representation for the adjacency matrix, where the rows represent vertices in V_1 and the columns represent vertices in V_2 (which is the top right submatrix of the first representation).

A subgraph of G is any graph G_S such that $E_S \subseteq E$ and $V_S \subseteq V$. An *induced subgraph* is some subset V_S of the vertices V, where the edges E_S in the subgraph are all of the edges in E that connect members of V_S. In this work, we will use 'subgraph' to mean an induced subgraph, unless specified otherwise. We could also consider spanning subgraphs, which consist of all of the vertices of V, but only some subset of E (that is, they are formed from G by deleting some edges) [113].



$$\begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \end{pmatrix} \qquad \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

FIGURE 2.2. An example bipartite 6-vertex graph and its associated adjacency matrices – firstly, where rows and columns represent vertices, and secondly where rows are the left-hand vertices and columns are the right-hand vertices (note that this is the same as the upper right, or transpose of the lower left, block of the first matrix).

One important characteristic of a graph G is its density, $d(G)$. This is defined as[8] [114]:

$$d(G) = \frac{2|E|}{|V|(|V|-1)}, \tag{2.37}$$

where $|E|$ and $|V|$ indicate the number of edges and vertices respectively. As a fully connected graph has $|V|(|V|-1)/2$ edges, $d(G)$ is between 0 and 1.

A matching is a set of edges $E_M \subseteq E$ such that no edges in E_M are incident on the same vertex. If every vertex of G is connected to an edge in E_M, this is a perfect matching (note this is only possible if the number of vertices of G is even).

Given an $n \times n$ square matrix M, let us introduce the permanent [115]:

$$\text{per}(M) = \sum_{\sigma \in \mathcal{S}_n} \prod_{i=0}^{n-1} M_{i,\sigma(i)} \tag{2.38}$$

where $\mathcal{S}_n$ indicates the symmetric group, that is the set of permutations of $\{0,..,n-1\}$. For example, $\mathcal{S}_3 = \{012, 021, 102, 120, 201, 210\}$. Hence, the permanent is a sum of $n!$ terms, in which each term is the product of n matrix elements, where, across the first indices, each number 0 to $n-1$ appears only once, and similarly for the second indices. We start with 0, instead of 1, as this makes it easier to convert back and forth with the code used for the simulations - nonetheless, this is a matter of convention and does not make any difference to the definitions. This is a matrix operation closely related to the determinant, but it is typically much more difficult to compute.This is because the negative terms in the determinant lead to cancellations (e.g. if two rows are the same), which can be used in simplifications such as Gaussian elimination.

We do not require M to be symmetric, but using the second representation described previously, we can choose M to be the adjacency matrix of a bipartite graph, with $|V_1|=$

[8]This convention differs in some sources - for example, to be the ratio between the number of edges and number of vertices, or the average degree of a vertex. Here it has been chosen to be normalised to between 0 and 1.

$|V_2| = n$, so the total number of vertices is $2n$. The permanent counts the number of perfect matchings of the graph – that is, every way that the vertices can be paired together, or every subset of edges such that each edge is connected to exactly one vertex. The graph in Fig. 2.2 contains one such perfect matching, shown in Fig. 2.3, so the permanent of its adjacency matrix is 1. For a weighted graph, each term in the sum is weighted by the product of the weights of the edges in the matching. In the matrix definition, you can see that every potential paring of vertices is considered, but some of them have weight zero because edges are missing. Hence a fully connected, unweighted graph has permanent $n!$.

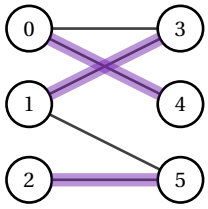


FIGURE 2.3. The perfect matching of a bipartite graph.

We would also like to consider the number of perfect matchings of a general graph (with an even number of vertices) – this is shown, using the example graph from Fig. 2.1, in Fig. 2.4. To do this, we introduce the hafnian function [115, 116]:

$$\text{haf}(M) = \sum_{\mu \in \mathcal{M}} \left(\prod_{k=0}^{|V|/2-1} M_{\mu_{2k},\mu_{2k+1}} \right). \tag{2.39}$$

Here, $\mathcal{M}$ is the set of perfect matching partitions of S (the different ways of 'pairing' the indices, or every permutation in which $\mu_{2k} < \mu_{2(k+1)}$ and $\mu_{2k} < \mu_{2k+1}$). There are $(n-1)!! = 1 \times 3 \times 5 \times ... \times (n-1)$ terms. The hafnian of a graph with an odd number of vertices is defined to be zero, and the hafnian of an empty graph is defined to be 1. The matrix M must be an adjacency matrix, so it must be symmetric.

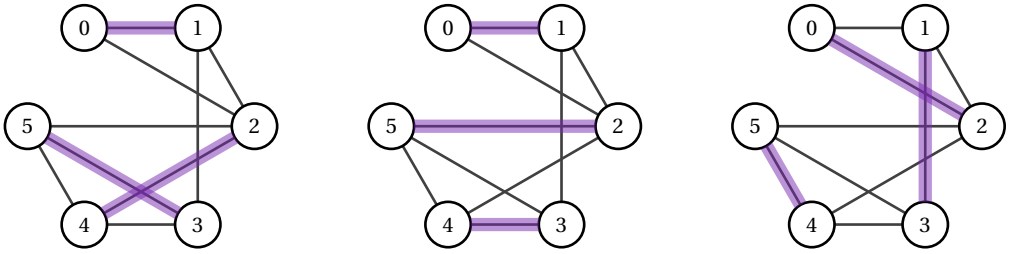


FIGURE 2.4. The perfect matchings of the example graph from Fig. 2.1 – every way to combine the vertices into pairs, so that each pair of vertices is connected by an edge.

The permanent and hafnian are both examples of a generalisation of this class of matrix functions, the immanant [117]. We note that there are several connections between the

permanent and hafnian. For example, the hafnian of a block anti-diagonal matrix [116]:

$$\text{haf}\left[\begin{pmatrix} 0 & M \\ M^T & 0 \end{pmatrix}\right] = \text{per}(M), \tag{2.40}$$

By comparison to the matrices in Fig. 2.2, we can see that these matrices are the two different adjacency matrix representations of a bipartite graph. This shows that the permanent and hafnian of a bipartite graph are equivalent (as expected), although the hafnian takes a $2n \times 2n$ size matrix as input, and the permanent takes an $n \times n$ size matrix as input.

Many problems can be expressed using the formalism of graph theory, which means that the complexity of graph problems is a rich field. Some examples include:

- The complexity of estimating the permanent, or hafnian, of a matrix whose entries are 0 or 1 is $\#\mathbf{P}$-complete [118].

- 11 of Karp's 21 $\mathbf{NP}$-complete problems involve graphs [30], including:

  whether a graph contains a clique (fully connected (sub)graph) of size k

  whether there is a subset of vertices V' (a vertex cover), with $|V'| \leq k$, such that every edge is connected to at least one vertex in V'

  whether there is a set of $\leq k$ vertices which can be removed such that the graph does not contain a cycle (closed path), or the analogous problem of removing edges from a directed graph

  whether the graph contains a Hamiltonian path, a path that visits each vertex exactly once, for both directed and undirected graphs

  where each vertex of the graph can be assigned one of k colours, so that no vertices joined by an edge have the same colour

  whether the vertices can be partitioned into $\leq k$ cliques

  whether, given some subset of the vertices, a tree (a graph in which any two vertices are connected by exactly one path, known here as the Steiner tree) exists of weight $\leq k$, that connects to all of the vertices of the subset

  given a 3-dimensional, tripartite hypergraph, whether there exists a matching of $\geq k$ hyperedges

  whether the vertices can be partitioned into two distinct sets (a 'cut'), such that there are at least k edges connecting vertices in the two different sets.

- The maximum matching – the matching with the most edges– can be found in polynomial time [119].

It is interesting to note that despite the permanent simply being the determinant but without the sign factor introduces before each term, it is significantly more difficult to compute. As explained in [120], the determinant of many matrices are zero (in particular, if two rows or columns are identical). These matrices would otherwise appear as additional terms in Gaussian elimination, which means that this is a simple process for determinants that cannot otherwise be applied to permanents.

2.6 Boson sampling

2.6.1 The model

As we have seen, passive linear optics are not thought to be scalable for universal quantum computation. Despite this, given that the implementation of measurement and feedforward remains a major roadblock for practical implementations [121], it is interesting to consider what it can do.

Although passive linear optics can be useful as a platform for quantum simulation (such as in [67]), it is difficult to ascertain whether this sort of experiment could solve problems which are classically intractable, and therefore demonstrate quantum advantage. There are many experimental quantum groups that are able to measure effects that are beyond the reach of classical simulation (e.g. in studying Bose-Einstein condensates [122]), but it is difficult to definitively say that there is not an as-yet-undiscovered algorithm that would be able to do so. Due to the work of Aaronson and Arkhipov in [84], quantum optics is fortunate in being able to confidently state this, at least in the limit of very large experiments.

We consider a setup known as *BosonSampling* (or often just 'boson sampling', or BS). We have seen how the bosonic nature of photons is important to the allowed operations of passive linear optics; fermion sampling, the fermionic equivalent, is a somewhat different scheme. In this framework, we have a passive linear optical interferometer with m modes, with $n \leq m$ single photons entering in different modes, and then we use photon-number-resolving (PNR) detectors at the output[9], as in Fig. 2.5.

We are interested in the probabilities of measuring different outcome patterns, $\mathbf{n}$, which is derived in [123]. We can assume w.l.o.g. that we input photons in the first n modes:

$$|\Psi_{in}\rangle = \prod_{i}^{n} \hat{a}_i^\dagger |\text{vac}\rangle \,. \tag{2.41}$$

In this case, $\hat{a}_i^\dagger$ indicates a photon creation operator in the i-th mode. For now, we will consider a more general input state (still n photons in m modes, so the product goes up to n,

[9] PNR detectors measure the number operator, $\hat{n}$, on a single mode.

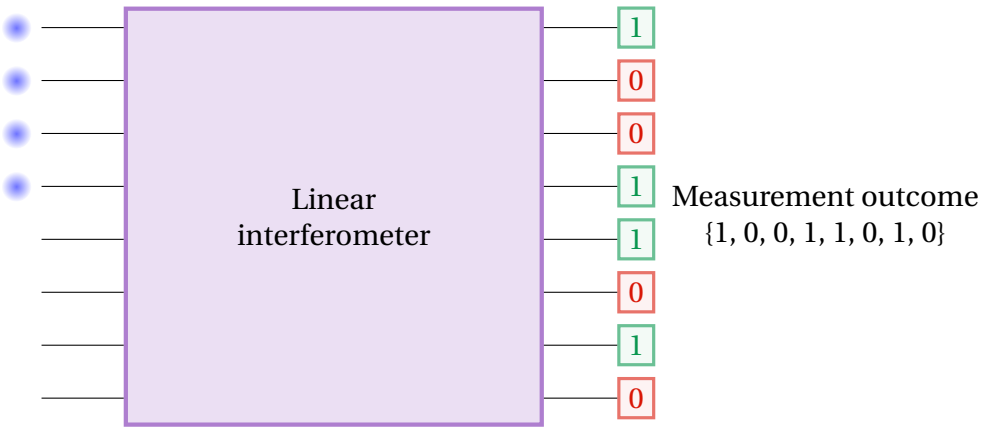


Figure 2.5: The boson sampling framework. $n = 4$ single photons enter an $m = 8$-mode linear interferometer, and are measured with photon-number-resolving detectors. The BS problem is to produce measurement outcomes randomly, but with the correct probabilities to match this physical framework.

but this time allowing each photon to be input into any mode):

$$|\Psi_{in}\rangle = \prod_{i}^{n} \frac{1}{\sqrt{\mathcal{N}}} \hat{a}^{\dagger}_{k_i} |\text{vac}\rangle \,. \tag{2.42}$$

Note that $\hat{a}^{\dagger}_{k_i}$ indicates that we are referring to the creation operator that is associated with the photon labelled by k_i. This labels a photon, not a mode – e.g. the vector $\mathbf{k} = (1, 2, 2, 4)^T$ represents 1 photon input to the first mode, 2 to the second mode, and 1 to the fourth mode. We then use $\mathcal{N}$ to indicate $\prod_i^n (n_i!)$, where n_i indicates the number of photons input to mode i.

The interferometer then implements a unitary transformation, according to Eq. 2.34. In the scheme of [84], this is a Haar-random unitary. Our output state is then:

$$\begin{aligned} |\Psi_{\text{out}}\rangle &= \prod_{i}^{n} \hat{b}^{\dagger}_{k_i} |\text{vac}\rangle \\ &= \prod_{i}^{n} \frac{1}{\sqrt{\mathcal{N}}} \sum_{j}^{m} U_{k_i,k_j} \hat{a}^{\dagger}_{k_j} |\text{vac}\rangle \,. \end{aligned} \tag{2.43}$$

Where we have used:

$$\hat{b}^{\dagger}_{k_i} = \sum_{j}^{m} U_{k_i,k_j} \hat{a}^{\dagger}_{k_j}, \tag{2.44}$$

We are interested in the probability of some output state $\mathbf{n}'$:

$$\begin{aligned} p(\mathbf{n}') &= \left|\langle \mathbf{n}' | \Psi_{out} \rangle\right|^2 \\ &= \left| \langle \text{vac}| \prod_l^n \frac{1}{\sqrt{\mathcal{N}'}} \hat{a}_{k_l} \prod_i^n \frac{1}{\sqrt{\mathcal{N}}} \sum_j^m U_{k_i,k_j} \hat{a}^\dagger_{k_j} |\text{vac}\rangle \right|^2 . \end{aligned} \tag{2.45}$$

Here, k_l indicates the indices of the output photons.

Now, we focus on one part of this:

$$\prod_i^n \sum_j^m U_{k_i,k_j} \hat{a}^\dagger_{k_j}. \tag{2.46}$$

By multinomial expansion, this creates a sum of m^n different terms, of the form

$$\prod_i^n U_{k_i,f(k_i)} \hat{a}^\dagger_{f(k_i)}.$$

The mapping $f(k_i)$ contains every choice of n elements from m, with replacement (multi-choose again). This gives:

$$p(\mathbf{n}') = \left| \sum_{f(k_i)} \langle \text{vac}| \prod_{l,i}^n \frac{1}{\sqrt{\mathcal{N}'}} \hat{a}_{k_l} \frac{1}{\sqrt{\mathcal{N}}} U_{k_i,f(k_i)} \hat{a}^\dagger_{f(k_i)} |\text{vac}\rangle \right|^2 . \tag{2.47}$$

The only surviving terms are those where $\{\sigma(k_l)\} = \{f(k_i)\}$, that is, the terms where the product of $\hat{a}^\dagger$ operators are the same as the product of $\hat{a}$ operators (in any order); $\{\sigma(k_l)\}$ denotes a reordering of $\{k_l\}$. Therefore, we arrive at:

$$\begin{aligned} p(\mathbf{n}') &= \left| \sum_{\sigma \in \mathcal{S}_n} \langle \text{vac}| \prod_{l,i}^n \frac{1}{\sqrt{\mathcal{N}'}} \hat{a}_{k_l} \frac{1}{\sqrt{\mathcal{N}}} U_{k_i,\sigma(k_l)} \hat{a}^\dagger_{\sigma(k_l)} |\text{vac}\rangle \right|^2 \\ &= \left| \sum_{\sigma \in \mathcal{S}_n} \prod_{l,i}^n \frac{1}{\sqrt{\mathcal{N}'}} \frac{1}{\sqrt{\mathcal{N}}} U_{k_i,\sigma(k_l)} \langle \text{vac}| \hat{a}_{k_l} \hat{a}^\dagger_{\sigma(k_l)} |\text{vac}\rangle \right|^2 \\ &= \left| \sum_{\sigma \in \mathcal{S}_n} \prod_{l,i}^n \frac{1}{\sqrt{\mathcal{N}'}} \frac{1}{\sqrt{\mathcal{N}}} U_{k_i,\sigma(k_l)} \right|^2 . \end{aligned} \tag{2.48}$$

Let us now introduce a modified matrix, $U_{\mathbf{n}',\mathbf{n}}$, where we choose the columns of U according to the input vector, and repeat column i, n_i times, and the rows are repeated similarly, according to the output vector $\mathbf{n}'$ – for example, the experiment in Fig. 2.5 produces the

submatrix according to the selected elements:

$$\begin{pmatrix} U_{1,1} & U_{1,2} & U_{1,3} & U_{1,4} & U_{1,5} & U_{1,6} & U_{1,7} & U_{1,8} \\ U_{2,1} & U_{2,2} & U_{2,3} & U_{2,4} & U_{2,5} & U_{2,6} & U_{2,7} & U_{2,8} \\ U_{3,1} & U_{3,2} & U_{3,3} & U_{3,4} & U_{3,5} & U_{3,6} & U_{3,7} & U_{3,8} \\ U_{4,1} & U_{4,2} & U_{4,3} & U_{4,4} & U_{4,5} & U_{4,6} & U_{4,7} & U_{4,8} \\ U_{5,1} & U_{5,2} & U_{5,3} & U_{5,4} & U_{5,5} & U_{5,6} & U_{5,7} & U_{5,8} \\ U_{6,1} & U_{6,2} & U_{6,3} & U_{6,4} & U_{6,5} & U_{6,6} & U_{6,7} & U_{6,8} \\ U_{7,1} & U_{7,2} & U_{7,3} & U_{7,4} & U_{7,5} & U_{7,6} & U_{7,7} & U_{7,8} \\ U_{8,1} & U_{8,2} & U_{8,3} & U_{8,4} & U_{8,5} & U_{8,6} & U_{8,7} & U_{8,8} \end{pmatrix} \tag{2.49}$$

This is an $n \times n$ matrix, and we can now relabel:

$$\begin{aligned} p(\mathbf{n}') &= \frac{1}{\mathbf{n}!\,\mathbf{n}'!} \left| \sum_{\sigma \in \mathcal{S}_n} \prod_i^n (U_{\mathbf{n}',\mathbf{n}})_{i,\sigma(i)} \right|^2 \\ &= \frac{1}{\mathbf{n}!\,\mathbf{n}'!} \left| \mathrm{per}(U_{\mathbf{n}',\mathbf{n}}) \right|^2 , \end{aligned} \tag{2.50}$$

where we have introduced the notation $\mathbf{n}! = \prod_i (n_i!)$. Recall the graph notation of the permanent, matching left-hand-side and right-hand-side vertices of bipartite graphs. Intuitively, we can match this to the model of boson sampling – if we say the first set of vertices represent our input modes, and the second set of vertices represent the output modes, the edges straightforwardly represent the transition amplitudes between input and output.

In that case, why is there a difference between the classical and quantum case? The answer is due to interference terms. Let's consider the case in which the photons are completely distinguishable. In this case, we would not expect the photons to interfere with each other, so we expect a different action of the interferometer.

To see this, we rewrite the input state, but giving each photon an 'internal' mode label (the one we do not measure, e.g. spectral mode), which are all distinct, and an 'external' mode label (e.g. spatial mode). The spatial modes are transformed as before, but the spectral modes are not changed:

$$|\Psi_{in}\rangle = \prod_i^n \frac{1}{\sqrt{\mathcal{N}}} \hat{a}^\dagger_{i,k_i} |\mathrm{vac}\rangle \, . \tag{2.51}$$

The output can be any arrangement of internal mode labels. Hence,

$$
\begin{aligned}
p(\mathbf{n}') &= \sum_{\tilde{\sigma}\in\mathcal{S}_n} \left| \langle \text{vac}| \prod_l^n \frac{1}{\sqrt{\mathcal{N}'}} \hat{a}_{\tilde{\sigma}(l),k_l} \prod_i^n \frac{1}{\sqrt{\mathcal{N}}} \sum_j^m U_{k_i,k_j} \hat{a}^\dagger_{i,k_j} |\text{vac}\rangle \right|^2 \\
&= \frac{1}{\mathbf{n}!\,\mathbf{n}'!} \sum_{\tilde{\sigma}\in\mathcal{S}_n} \left| \prod_l^n \prod_i^n \sum_j^m U_{k_i,k_j} \langle \text{vac}| \hat{a}_{\tilde{\sigma}(l),k_l} \hat{a}^\dagger_{i,k_j} |\text{vac}\rangle \right|^2 \\
&= \frac{1}{\mathbf{n}!\,\mathbf{n}'!} \sum_{\tilde{\sigma}\in\mathcal{S}_n} \left| \prod_l^n \sum_j^m U_{k_{\tilde{\sigma}(l)},k_j} \langle \text{vac}| \hat{a}_{\tilde{\sigma}(l),k_l} \hat{a}^\dagger_{\tilde{\sigma}(l),k_j} |\text{vac}\rangle \right|^2 \\
&= \frac{1}{\mathbf{n}!\,\mathbf{n}'!} \sum_{\tilde{\sigma}\in\mathcal{S}_n} \left| \prod_l^n U_{k_{\tilde{\sigma}(l)},k_l} \langle \text{vac}| \hat{a}_{\tilde{\sigma}(l),k_l} \hat{a}^\dagger_{\tilde{\sigma}(l),k_l} |\text{vac}\rangle \right|^2 \\
&= \frac{1}{\mathbf{n}!\,\mathbf{n}'!} \sum_{\tilde{\sigma}\in\mathcal{S}_n} \prod_l^n \left| U_{k_{\tilde{\sigma}(l)},k_l} \right|^2 \\
&= \frac{1}{\mathbf{n}!\,\mathbf{n}'!} \text{per}\left(|U_{\mathbf{n}',\mathbf{n}}|^2\right) .
\end{aligned}
\tag{2.52}
$$

In a slight abuse of notation, we are using $|U|^2$ to indicate the matrix where we take the absolute value squared of every element of U.

This matrix has only real, non-negative elements. Indeed, it is the presence of negative terms which suggests the occurence of quantum interference. As will be discussed later on, this means it is efficient to estimate the permanent classically – thus, we have lost the quantum advantage.

2.6.2 Computational complexity

This framework for quantum devices is very different to the previous algorithms we have seen, where the goal is to produce a state of interest and then measure that to find the answer to the problem. On the other hand, boson sampling may seem more like simulation, in which we are interested in the evolution of a certain quantum system.

Ultimately, we are interested in whether this framework displays quantum advantage. Even though permanents, which describe our output probabilities, are in general $\#\mathbf{P}$-hard to calculate and therefore inefficient to compute classically, this class is also larger than $\mathbf{BQP}$, and therefore we do not expect that we should be able to calculate a permanent directly using a quantum computer. Indeed, this would not be possible using a BS device, as we would have to take an exponential number of samples in order to estimate (with reasonable precision) the individual probability that we are interested in. Despite this, the main result of [84] is that it is not possible for a classical algorithm to exist that is able to efficiently simulate a boson sampler. Here we give a brief summary of their complexity proof. Certain details will become relevant in Chapter 5, and therefore more detail is given in Appendix C.

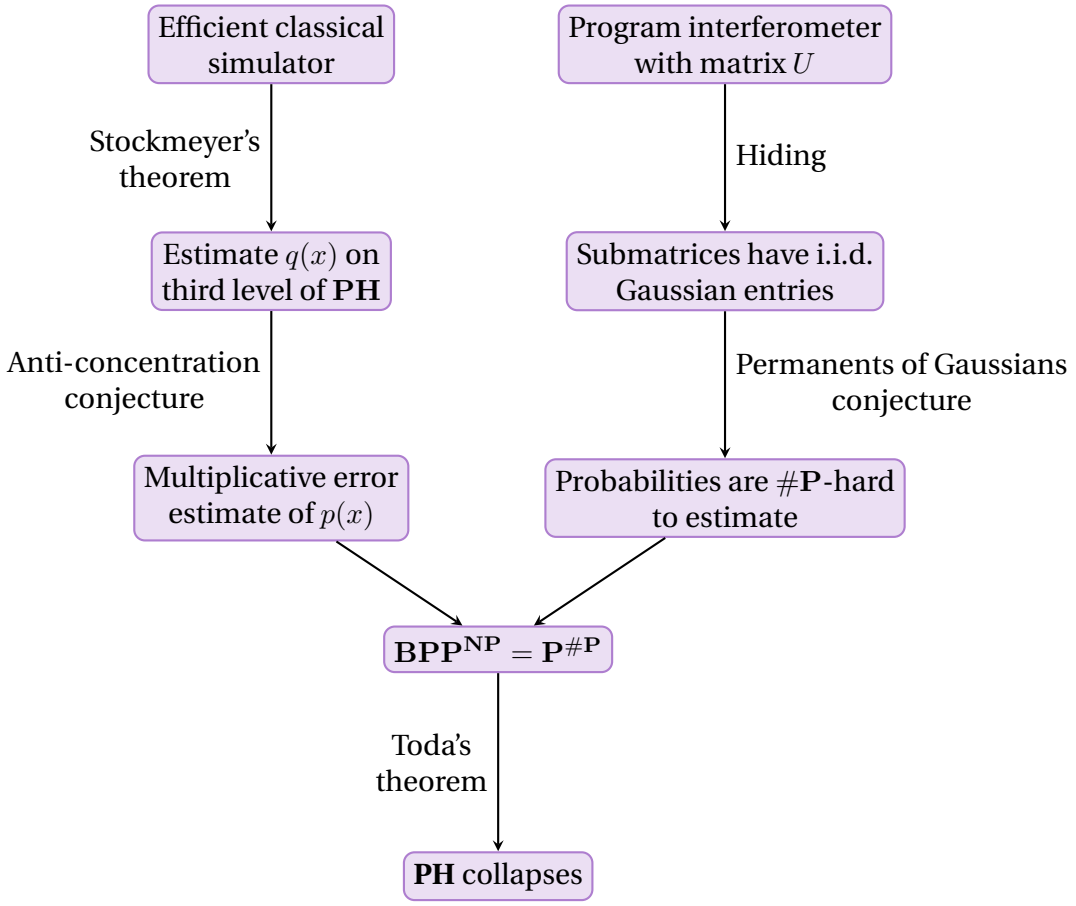


Figure 2.6: An outline of the complexity proof structure from [84]. On the left-hand side are elements of the proof that concern the computational power of an efficient classical BS simulator (particularly regarding the complexity class in which it could estimate outcome probabilities), and on the right-hand side are elements of the proof that concern the difficulty of the task of estimating outcome probabilities. These come to a contradiction, showing that an efficient algorithm with the power to simulate BS cannot exist.

Any quantum algorithm must result in the output of classical information by measuring the resulting quantum state. Due to the inherent randomness of measurement, this samples from an underlying distribution – it returns some outcome x with probability $p(x)$. With many algorithms, a distribution is produced so that there are few possible outcomes, which represent the answer to the problem.

As we have just seen, in BS, each sample has a probability proportional to the modulus square of the permanent of the submatrix (of the unitary describing the interferometer) related to that sample. This (ideal) distribution is referred to as the ground truth. Of course, implementing this exactly – that is, producing samples with the same probability as the ground truth – is not realistic for the quantum machine. Instead, we sample from a different, approximate, distribution, where outcome x has probability $q(x)$. The accuracy of

approximate sampling is given by the total variation distance (TVD) [78]:

$$\|p-q\|_{TV} = \frac{1}{2}\sum_x |p(x)-q(x)|. \tag{2.53}$$

Why the factor of $1/2$? This definition of TVD is equal to $\sup_x |p(x)-q(x)|$, the largest possible difference in probabilities that the distributions could assign to a single event. For example, say that for an event x, $p(x)-q(x)=\alpha$. Then, the other probabilities of p must sum to $1-p(x)$, and the other probabilities of q must sum to $1-p(x)+\alpha$. Therefore (although we've skipped a few steps), the TVD must be at least 2α.

In the framework of BS, we choose a constant ϵ, and try to sample from a distribution with TVD $\leq \epsilon$. To show that this is infeasible for classical machines to do efficiently (in time polynomial in n and $1/\epsilon$), we imagine the existence of some classical oracle $\mathcal{C}$ that is capable of doing this, and show how we could use this oracle to estimate permanents. We find that this would cause a contradiction due to the computational complexity of doing so. We ask: given a BS simulator, what sort of problems would we be able to solve, and how hard would it be to do so? We find that we would be able to estimate the permanents of a certain class of matrices, and although we wouldn't be able to do so in polynomial time, we could do that more efficiently than is allowed by complexity theory.

When considering approximate sampling, we will link this to the estimation of individual permanents. There are two types of error we can consider: multiplicative error, which means that for estimation E of quantity Q, $|E-Q| \leq \epsilon|Q|$, or additive error, where $|E-Q| \leq \epsilon$. Multiplicative error is generally a stronger condition.

The method for making this connection comes from a theorem by Stockmeyer [124]. This theorem considers some Boolean function; in our case, this is the function implemented by $\mathcal{C}$, which receives a description of the boson sampler, an error bound (as above), and a random seed to imitate the randomness of sampling, and we also choose a particular measurement outcome which we are interested in. Then, the output to the function is 1 if the pattern given by the simulator is equal to our chosen outcome. Stockmeyer's theorem states that the probability that the output to this Boolean function is 1 (this is equivalent to the probability of the classical simulator sampling a certain outcome) can be estimated within a multiplicative error using an algorithm in class $\mathbf{FBPP}^{\mathbf{NP}^{\mathcal{C}}}$.

The use of Stockmeyer's theorem is particularly interesting. The (classical) algorithm inputs a random string to imitate sampling, however it has some control over the 'randomness' of the procedure. This is not the case with the inherent randomness of quantum computers. Therefore, this proof is about the power of a hypothetical classical computer, and is beyond the power that we expect from a quantum computer. In fact, the exact complexity class of BS is not known.

Now, we must consider the problem we are trying to solve using our BS simulator. To do so, we impose that the unitary U describing the interferometer is Haar-random (this is

the unique translation-invariant probability measure on the unitary group – that is, if U is Haar-randomly distributed, so are $U'U$ and UU' for a fixed unitary U' [125]). We are also interested in the collision-free regime, which means that there is at most one photon in each output mode. This is likely to be true if[10] $m = o(n^2)$, due to the 'bosonic birthday paradox', considered in [84] (in particular, this states that the expected probability for collisions is upper bounded by $\frac{2n^2}{m}$). We then find that the submatrices describing our output distribution are close[11] in variation distance to matrices with Gaussian i.i.d. (independent and identically distributed) elements – these matrices are indicated by X. That is, the submatrices we are interested in 'look like' matrices with randomly generated, independent entries, and not unitary matrices. This is useful to us because these sorts of matrices have useful properties we can use later on. This is known as the hiding property, and is proven for $m = o(n^6)$, but there is also considerable evidence for it at $m = o(n^2)$. Be careful here, about the difference between 'Gaussian matrices' (matrices with i.i.d. Gaussian entries) and 'Gaussian states' (which we will discuss later, but are not based on Gaussian matrices, but instead are related to a multivariate Gaussian distribution). Finally, we draw from distributions weighted by $|\text{per}(X)|^2$ and not $\text{per}(X)$, and so [84] also finds a reduction between these.

Taking all these factors into account, we consider the following problem: we have some Gaussian i.i.d. matrix of which we would like to estimate the permanent. How hard is this? Using the permanent-of-Gaussians conjecture (as named in [23]), it is $\#\mathbf{P}$-hard to estimate $\text{per}(X)$ with multiplicative error. This conjecture focuses on the average-case hardness (instead of worst-case, which is easier to prove). Without this, it would be possible to design a classical BS simulator that was successful the majority of the time, but simply failed on the hard cases, and therefore, for any given experiment, it would be able to simulate the majority of outcomes accurately, and only fail for a few probabilities – keeping the total TVD low. This conjecture remains an open problem in the field, but it is generally considered to be plausible (recent studies include [126–128]).

The result using Stockmeyer's algorithm shows the complexity of the algorithm with which the simulator $\mathcal{C}$ would be able to estimate the probabilities $q(x)$ (that is, the probabilities of an approximate scheme, similar to the experiment that we are hoping to simulate). As these probabilities are defined according to the TVD, this only allows us (with high probability of success) to estimate permanents with *additive* error, due to the possible differences between the distributions p and q. A (relatively involved) reduction between additive and multiplicative error is given in [84], which requires a conjecture known as the anti-concentration condition to hold on the distributions described by permanents of Gaussians. This imposes a condition onto the distribution of probablities, ensuring that there are not too many sig-

[10] The notation $o(f(n))$, as opposed to $O(f(n))$, indicates that the function is lower bounded by $f(n)$, instead of upper bounded.

[11] The exact relationship depends on the relationship between m and n, however for variation distance $O(\delta)$, m is polynomial in n and $1/\delta$.

nificantly high, or low, probabilities – that is, the distribution appears relatively uniform. A full description is omitted from this section, but we will return to the anti-concentration condition in Chapters 4 and 5.

If anti-concentration did not hold, then the distributions would be too 'spiky', with some very probable outcomes and many outcomes with probability close to zero. For these very improbable outcomes, we would struggle to get a multiplicative-error estimation, and in a pathological example, these would contain the majority of the difficult-to-estimate probabilities, so our BS simulator would be failing to do anything difficult.

Indeed, we generally see that very concentrated distributions are easier to 'spoof' [129], as we will examine further in Section 5. These distributions only contain a few 'important' outcomes, which are often easy to find, and hence we can sample mostly from these and assume that the probability of the other outcomes is zero.

With all of these conditions, our efficient classical simulator would be able to solve $\#\mathbf{P}$-hard problems (i.e., the estimation of permanents with multiplicative error) in $\mathbf{FBPP}^{\mathbf{NP}^{\mathcal{C}}}$ (due to Stockmeyer's theorem). The latter is in the third order of the polynomial hierarchy, but due to Toda's theorem (Thm. 2.1), this is a contradiction, and hence such a simulator cannot exist. There is a more complete survey of this paper ([84]) in Appendix Section C. We show the structure of the proof in Fig. 2.6 (based on a similar figure in [84]).

There are many difficulties in experimentally implementing boson sampling, particularly with a sufficiently low level of error. However, a major roadblock has proven to be producing single photons. This is an active area of research, but at the time of writing, it is usually done with non-deterministic sources, and therefore producing many high-quality single photons on demand has vanishingly small probability [130]. Several potential solutions to this have been proposed. The main focus of this thesis is Gaussian boson sampling (GBS), a variation of BS which requires the use of the continuous variable formalism to understand.

2.7 Continuous variable quantum information

2.7.1 Phase space representation

As we have discussed, photons are quanta of the electromagnetic field that represent different energy eigenstates of the harmonic oscillator. So far, we have considered Fock states, eigenstates of $\hat{n}$, which correspond to challenging single-photon preparation and photon-number-resolving (PNR) measurements in our state preparation and detection steps. As we have discussed, these states have a discrete energy spectrum.

This is not the only meaningful way to represent electromagnetic radiation. We can consider instead the *quadratures* of the field, meaning the position and momentum observables. Recall from the definition of the harmonic oscillator that these are related to the ladder

operators by:

$$\hat{x} = \sqrt{\frac{\hbar}{2m\omega}}(\hat{a}^\dagger + \hat{a}) \tag{2.54}$$

$$\hat{p} = i\sqrt{\frac{\hbar m\omega}{2}}(\hat{a}^\dagger - \hat{a}). \tag{2.55}$$

Unlike the ladder operators, these correspond to real observables and are hence Hermitian. Note that $[\hat{x}_i, \hat{p}_j] = i\hbar\delta_{ij}$. The eigenstates of these operators have a continuous spectrum (in general, although under specific conditions they may be discrete), and so form the basis for an infinite-dimensional Hilbert space.

We will follow [131] and [132], with considerable inspiration taken from [133], and [134]. Consider a system of m distinct bosonic modes. Each of these will be a distinct harmonic oscillator, associated with distinct quadrature operators. We can express these in the vector $\hat{\mathbf{r}} = (\hat{x}_1, ..., \hat{x}_m, \hat{p}_1, ..., \hat{p}_m)^T$. We then have:

$$[\hat{r}_i, \hat{r}_j] = i\hbar\Omega_{ij}, \tag{2.56}$$

where:

$$\Omega = \begin{pmatrix} 0 & \mathbb{1} \\ -\mathbb{1} & 0 \end{pmatrix}. \tag{2.57}$$

Thus far we have discussed transformations in the Hilbert space, $\hat{U}|\psi\rangle$, and transformations of the creation/annihilation operators, $U\hat{\mathbf{a}}$, and therefore it may seem natural to now talk about transformations of the quadrature operators, $M\hat{\mathbf{r}}$. These transformations must preserve Eq. 2.56, so:

$$M^T\Omega M = \Omega, \tag{2.58}$$

which is the condition that the matrix M is *symplectic*.

In more detail:

$$\begin{aligned} [(M\hat{r})_i, (M\hat{r})_j] &= \left[\left(\sum_k M_{ik}\hat{r}_k\right), \left(\sum_l M_{jl}\hat{r}_l\right)\right] \\ i\hbar\Omega_{ij} &= \sum_{k,l} M_{ik}M_{jl}[\hat{r}_k, \hat{r}_l] \\ &= i\hbar\sum_{k,l} M_{ik}\Omega_{kl}M_{jl} \\ \Omega &= M\Omega M^T, \end{aligned} \tag{2.59}$$

which (as can be shown with a little rearranging and noting that $\Omega^{-1} = \Omega^T = -\Omega$) is an equivalent condition.

Recall from the discussion of Bogoliubov operators in Section 2.3 that a transformation on the vector of operators is not required to be unitary – in fact, this symplectic condition is what leads to the unitary condition in DV. However, thanks to linearity, we must consider affine transformations, i.e. that is why we can consider matrix multiplication.

Given that we are no longer in the discrete variable (DV) picture, we will need to define a more convenient approach to express states than infinite-dimensional matrices and vectors. We will therefore consider quasi-probability distributions. In classical mechanics, if we wanted to describe a state in terms of certain properties (e.g. position or momentum), each described by probabilities, it would be useful to introduce a probability distribution $p(\mathbf{r})$ in terms of these variables.

For the remainder of this work, instead of the definition of the quadrature operators in Eqs. 2.54, 2.55, we will use the 'dimensionless' definition of these operators:

$$\hat{x'} = \frac{1}{\sqrt{2}}(\hat{a}^\dagger + \hat{a}) \tag{2.60}$$

$$\hat{p'} = i\frac{1}{\sqrt{2}}(\hat{a}^\dagger - \hat{a}). \tag{2.61}$$

These have the same commutation relations, and we simply have to rescale the eigenvalues. For notational simplicity, we will write $\hat{x}$ and $\hat{p}$ instead of $\hat{x}'$ and $\hat{p}'$, but any future use of the quadrature operators refers to the above equations. We will also use the 'natural units' of $\hbar = c = 1$. Let's consider what this means. Typically, a symbol such as t represents a physical quantity that has both a value, e.g. 2, and a unit, e.g. s. In this case, $\hbar$ represents 1.055×10^{-34} J s in S.I. units.

We could define a new system of units, including the 'quantum unit' ($\mathcal{Q}$), which is defined so that $1\mathcal{Q} = 1.055 \times 10^{-34}$ J s. We could then redefine all other units to be in terms of $\mathcal{Q}$, so $\hbar = 1\mathcal{Q}$, and $2\,\mathrm{s} = 2.109 \times 10^{-34}\,\mathrm{J}\,\mathcal{Q}^{-1}$, etc. There are several ways of redefining our system of units, so this needs to be done carefully, but we can worry about this later on when we go to replacing variables with numbers. We can also go a step further and ignore the quantum unit entirely, assuming that they usually cancel out in our formulae, and just stop writing $\hbar$ when it comes up, because it is equal to unity anyway.

Alternatively (and in a more mathematically sound manner), we can consider redefining variables[12]. Therefore, instead of using $\hbar = 1.055 \times 10^{-34}$ J s, we can define a new variable, $\tilde{\hbar} = 1.055 \times 10^{-34}\,\mathrm{J\,s}/\hbar = 1$, which is inherently dimensionless. We then redefine other variables in response, e.g. $E = \tilde{\hbar}\tilde{\omega}$, where $\tilde{\omega} = \hbar\omega$. Once again, we have a choice of redefining E or ω, but this is something we can deal with later on. Of course, because $\tilde{\hbar} = 1$, we can simply not write it when it multiplies other variables. Also, because I've now told you that we are using natural units, this redefinition is implied and we won't bother writing $\tilde{\omega}$ instead of ω.

[12] It may therefore be more fitting if this were called 'natural variables' instead of natural units.

This process is known as nondimensionalisation and is common practice. In particular, most of the quantities that we will be interested in in this thesis are probabilities, and hence are dimensionless, and therefore we expect that the redefinition of these variables will cancel out before we arrive at our final outcomes. Hence, keeping track of where the $\hbar$'s should go is something only the experimentalists need to concern themselves with.

We can also do a similar process with c. However, despite us considering light, this comes up less often in this work.

To represent quantum states in terms of these *continuous* variables, we will first consider the Wigner function, which has the property that:

$$\int_{\mathbb{R}^{2m-1}} W_\rho(x_1,...,x_m,p_1,...,p_m)\mathrm{d}p_1...\mathrm{d}p_m\mathrm{d}x_1...\mathrm{d}x_{m-1} = \langle x_m|\,\rho\,|x_m\rangle\,. \tag{2.62}$$

where W_ρ indicates that we are considering the Wigner function associated with the state ρ, and $|x_m\rangle$ is the position eigenstate in mode m. That is, the probability distributions of the state over a particular co-ordinate (e.g. x_m), the marginals of the function, can be found in a similar way to the treatment of a classical joint probability distribution (by integrating over the irrelevant variables).

If we express this in terms of $\hat{\mathbf{r}}$ operators:

$$W_\rho(\mathbf{x},\mathbf{p}) = \frac{1}{(2\pi)^m}\int_{\mathbb{R}^m}\left\langle \mathbf{x}+\frac{\mathbf{q}}{2}\right|\rho\left|\mathbf{x}-\frac{\mathbf{q}}{2}\right\rangle \mathrm{e}^{i\mathbf{q}\cdot\mathbf{p}}\mathrm{d}^m\mathbf{q}. \tag{2.63}$$

If not using natural units, we make the replacements $\frac{1}{(2\pi\hbar)^m}$ and $\mathrm{e}^{i\mathbf{q}\cdot\mathbf{p}/\hbar}$.

To see where this comes from, we first have to consider the relationship between the $\hat{x}$ and $\hat{p}$ eigenstates. For simplicity, the functions will for now be presented w.l.o.g. in terms of single-mode operators. We follow [134].

First note that:

$$\begin{aligned}\langle x|\,[\hat{x},\hat{p}]\,|y\rangle &= i\,\langle x|y\rangle\\ x\,\langle x|\,\hat{p}\,|y\rangle - y\,\langle x|\,\hat{p}\,|y\rangle &= i\delta(x-y)\\ \langle x|\,\hat{p}\,|y\rangle &= \frac{i\delta(x-y)}{x-y},\end{aligned} \tag{2.64}$$

from which an arbitrary state $|\psi\rangle$ satisfies:

$$\langle x|\,\hat{p}\,|\psi\rangle = \int \mathrm{d}x' i\frac{\delta(x-x')}{x-x'}\psi(x'), \tag{2.65}$$

in which $\psi(x) = \langle x|\psi\rangle$ is the state $|\psi\rangle$ in the position basis.

Expanding $\psi(x') = \sum_{n=0}^{\infty}\frac{(x'-x)^n}{n!}\frac{d^n\psi(x)}{dx^n}$, we have:

$$\langle x|\,\hat{p}\,|\psi\rangle = \sum_{n=0}^{\infty}\frac{i}{n!}\int \mathrm{d}x'\frac{\delta(x-x')}{x-x'}(x'-x)^n\frac{\mathrm{d}^n\psi(x)}{\mathrm{d}x'^n}. \tag{2.66}$$

The integration then gives 0 for all but the $n = 1$ term (note that $\int \frac{\delta x}{x} = 0$). We therefore have:

$$\langle x|\,\hat{p}\,|\psi\rangle = -i\frac{\mathrm{d}\psi(x)}{\mathrm{d}x} = -i\frac{\mathrm{d}\,\langle x|\psi\rangle}{\mathrm{d}x}. \tag{2.67}$$

We now set $|\psi\rangle = |p\rangle$:

$$\langle x|\,\hat{p}\,|p\rangle = p\,\langle x|p\rangle = -i\frac{\mathrm{d}\,\langle x|p\rangle}{dx}, \tag{2.68}$$

which is a differential equation with the solution:

$$\langle x|p\rangle = \frac{1}{\sqrt{2\pi}}\mathrm{e}^{ipx}, \tag{2.69}$$

with the normalisation condition to ensure $\langle p|p'\rangle = \delta(p-p')$. (Useful delta function identity: $\frac{1}{2\pi}\int_{-\infty}^{\infty}\mathrm{e}^{-ik(x-y)}dk = \delta(x-y)$. Show this by inserting $\int dx\,|x\rangle\langle x|$.)

We then follow the description in [135] for the intuition of the Wigner function. Consider:

$$\begin{aligned}\int \left\langle\psi\middle|x+\frac{q}{2}\right\rangle\left\langle x+\frac{q}{2}\middle|p\right\rangle\left\langle p\middle|x-\frac{q}{2}\right\rangle\left\langle x-\frac{q}{2}\middle|\psi\right\rangle dq &= \frac{1}{2\pi}\int \left\langle\psi\middle|x+\frac{q}{2}\right\rangle\left\langle x-\frac{q}{2}\middle|\psi\right\rangle \mathrm{e}^{ipq}dq \\ &= \frac{1}{2\pi}\int \psi^*\left(x+\frac{q}{2}\right)\psi\left(x-\frac{q}{2}\right)\mathrm{e}^{ipq}dq.\end{aligned} \tag{2.70}$$

We can understand this to be all of the possible trajectories (from position $x-\frac{q}{2}$ to $x+\frac{q}{2}$) of a particle through point x with momentum p.

It is also straightforward to check that it is symmetric to write in terms of x or p, and that it satisfies the marginals property Eq. 2.62, using $\frac{1}{2\pi}\int_{-\infty}^{\infty}\mathrm{e}^{-ik(x-y)}dk = \delta(x-y)$.

We note that there exist several different conventions for the definition of important functions and variables in phase space quantum optics; in particular, different conventions are used in the papers in which we report the results described in this thesis. Therefore a table is given in Appendix Section A for comparison.

If the Wigner function is fully positive, this is analogous to a classical probability distribution, particularly considering the expectation values:

$$\begin{aligned}\langle x\rangle &= \iint xW(x,p)\mathrm{d}x\mathrm{d}p \\ \langle p\rangle &= \iint pW(x,p)\mathrm{d}x\mathrm{d}p.\end{aligned} \tag{2.71}$$

This description does not fully capture quantum interference – in fact, the Wigner function can take negative values, meaning it is better described as a quasiprobability distribution. Interestingly, with sufficiently coarse graining, these negative regions disappear, which aligns with regaining classical behaviour in the limit of large distances.

We can compare the Wigner function with the Liouville density, a probability distribution which represents the probability of a classical particle to have a certain position or momentum. Due to the uncertainty principle, it is not possible to construct a probability distribution

for a quantum particle that has the same interpretation. The Wigner function is similar, but the negative values show it cannot be used as a probability distribution. However, this is not always the case – in particular, a fully-positive Wigner function corresponds to a Gaussian state, which we will consider further in the next section.

In this statement of the Wigner function, despite there not being a clear physical meaning, the input parameters $\mathbf{x}$, $\mathbf{p}$ correspond to intuitive physical parameters of the $\hat{x}$ and $\hat{p}$ expectation values. We now use an alternative set of parameters $\boldsymbol{\xi} = (\xi_{p,1}, ..., \xi_{p,m}, \xi_{x,1}, ..., \xi_{x_m})^T$, which do not have the same physical meaning but, as we will see, will lead to a useful description of photonic states in terms of more natural operators.

We can write:

$$W_\rho(\mathbf{x}, \mathbf{p}) = \frac{1}{(2\pi)^{2m}} \int \mathrm{d}^{2m}\boldsymbol{\xi}\, \mathcal{X}_\rho(\boldsymbol{\xi}) \exp\left(i\boldsymbol{\xi}^T \Omega(\mathbf{x}, \mathbf{p})^T\right), \tag{2.72}$$

where we have introduced the parameters $\boldsymbol{\xi} = (\xi_{p,1}, ..., \xi_{p,m}, \xi_{x,1}, ..., \xi_{x,m})^T = (\boldsymbol{\xi}_p^T, \boldsymbol{\xi}_x^T)$, and the characteristic equation:

$$\begin{aligned} \mathcal{X}_\rho(\boldsymbol{\xi}) &= \int \mathrm{d}^m\mathbf{q} \left\langle \mathbf{q} - \frac{\boldsymbol{\xi}_p}{2} \right| \rho \left| \mathbf{q} + \frac{\boldsymbol{\xi}_p}{2} \right\rangle \mathrm{e}^{i\boldsymbol{\xi}_p \cdot \mathbf{q}} \\ &= \mathrm{Tr}\left[\rho \hat{D}(\boldsymbol{\xi})\right], \end{aligned} \tag{2.73}$$

where $\hat{D}$ is the displacement operator:

$$\hat{D}(\boldsymbol{\xi}) = \exp\left(i\hat{\mathbf{r}}^T \Omega \boldsymbol{\xi}\right). \tag{2.74}$$

This may seem to have come from nowhere, but the individual steps can be proven without too much difficulty. We will present it here in the single-mode case for simplicity. As before, this description mostly follows the notes of [134], but there is similar information in, e.g. [132].

Firstly, use the Baker-Campbell-Hausdorff formula to write $\hat{D}(\boldsymbol{\xi}) = e^{-i\xi_x\xi_p/2} e^{i\xi_x\hat{x}} e^{-i\xi_p\hat{p}}$. We then start with the second line of Eq. 2.73, $\mathrm{Tr}\left(\rho\hat{D}(\boldsymbol{\xi})\right) = \int dx \langle x| \rho |x\rangle$, and we show that it is equal to the first line by using the identity $e^{-iy\hat{p}} |x\rangle = |x + y\rangle$ (which can be proven also using the BCH formulas on $e^{iy\hat{p}}\hat{x}e^{-iy\hat{p}}$).

We then substitute this into Eq. 2.72 and use the delta function identity from before, as well as a change of basis, to get back to our previous statement of the Wigner function.

The bigger question is why introduce this – however, we now have a nice representation of states in terms of the expectations value of the displacement operator, which as we will see, allows us to write things in terms of the ladder operators, as we are used to doing. This is nice because it contains the simplicity of the Gaussian representation but also gives us the chance to work out important functions when translating to discrete variables, e.g. for PNR measurements.

Alternatively, we can make a change of variables and consider the complex vector $\boldsymbol{\alpha} = \frac{1}{\sqrt{2}}(\boldsymbol{\xi}_p + i\boldsymbol{\xi}_x)$, so:

$$\hat{D}(\boldsymbol{\alpha}) = \exp\Big(\boldsymbol{\alpha}\cdot\hat{\mathbf{a}}^\dagger - \boldsymbol{\alpha}^*\cdot\hat{\mathbf{a}}\Big). \tag{2.75}$$

The displacement operator acts on the vacuum to give a coherent state, $\hat{D}(\alpha)\,|\text{vac}\rangle = |\alpha\rangle$, which we can once again show using BCH formulae. Thus far we have been discussing the behaviour of individual photons, which are of primary interest to quantum opticians. These are Fock states, which have a well-defined photon number at the expense of maximum uncertainty in phase. A coherent state is the opposite. It is the eigenstate of $\hat{a}$: $\hat{a}\,|\alpha\rangle = \alpha\,|\alpha\rangle$, and the eigenvalue $\alpha = |\alpha|e^{i\theta}$ defines a specific phase, θ. It can be written as a superposition of Fock states:

$$|\alpha\rangle = e^{-|\alpha|^2/2}\sum_{n=0}^{\infty}\frac{\alpha^n}{\sqrt{n!}}\,|n\rangle\,. \tag{2.76}$$

The coherent states form an overcomplete (normalised but not orthogonal) basis for the Hilbert space, and will be particularly useful if we want to write operators as a function of the ladder operators.

The name coherent state follows from the quality of minimum uncertainty in phase, from the idea of it being emitted by many sources in phase, although this is not necessarily the best description.

The characteristic function can be generalised to any operator:

$$\mathcal{X}_{\hat{O}}(\boldsymbol{\alpha}) = \text{Tr}\Big[\hat{D}(\boldsymbol{\alpha})\hat{O}\Big]. \tag{2.77}$$

We can write any operator as a polynomial function in terms of creation and annihilation operators: $\hat{O} = f(\hat{a},\hat{a}^\dagger)$. This can be written in several different ways:

- Normal ordering: $\hat{O}^{(N)}$ is expressed as a sum of terms of the form $(\hat{a}^\dagger)^j(\hat{a})^k$.
- Anti-normal ordering: $\hat{O}^{(A)}$ is expressed as a sum of terms of the form $(\hat{a})^j(\hat{a}^\dagger)^k$.
- Symmetric ordering: $\hat{O}^{(S)}$ is expressed as the normalised sum of all possible orderings of j $\hat{a}$ and k $\hat{a}^\dagger$, and is therefore symmetric under the exchange of $\hat{a}$ and $\hat{a}^\dagger$.

These are equivalent ways of writing the same operator, and it is possible to convert between them using the commutation relations. Operators do not have to be written in these forms, but it is always possible to convert an operator into these forms.

By using the Baker-Campbell-Hausdorff formulas, we can write $\hat{D}(\alpha) = e^{\alpha\hat{a}^\dagger}e^{-\alpha^*\hat{a}}e^{-|\alpha|^2/2} = e^{-\alpha^*\hat{a}}e^{\alpha\hat{a}^\dagger}e^{+|\alpha|^2/2}$, giving the normal-ordered and anti-normal-ordered forms, respectively. We define the s-ordered characteristic function:

$$\chi_{\rho,s}(\boldsymbol{\alpha}) = \text{Tr}\Big[\hat{D}(\boldsymbol{\alpha})\rho\Big]e^{s|\boldsymbol{\alpha}|^2/2}. \tag{2.78}$$

For a function $\hat{O} = \hat{a}^{\dagger m}\hat{a}^n$, these lead to the relationships:

$$\langle \hat{O}^{(N)} \rangle = \left(\frac{\partial}{\partial\alpha}\right)^m \left(-\frac{\partial}{\partial\alpha^*}\right)^n \chi_1(\alpha)\Big|_{\alpha=0} \tag{2.79}$$

$$\langle \hat{O}^{(S)} \rangle = \left(\frac{\partial}{\partial\alpha}\right)^m \left(-\frac{\partial}{\partial\alpha^*}\right)^n \chi_0(\alpha)\Big|_{\alpha=0} \tag{2.80}$$

$$\langle \hat{O}^{(A)} \rangle = \left(\frac{\partial}{\partial\alpha}\right)^m \left(-\frac{\partial}{\partial\alpha^*}\right)^n \chi_{-1}(\alpha)\Big|_{\alpha=0}. \tag{2.81}$$

Therefore, we introduce the more general class of s-ordered quasiprobability distributions:

$$W_s(\boldsymbol{\alpha}) = \frac{1}{\pi^{2m}} \int \mathrm{d}^{2m}\boldsymbol{\beta} \mathrm{e}^{(\boldsymbol{\alpha}\cdot\boldsymbol{\beta}^* - \boldsymbol{\alpha}^*\cdot\boldsymbol{\beta})} \chi_s(\boldsymbol{\beta}), \tag{2.82}$$

noting that the Wigner function in the complex basis corresponds to W_0.

We can now introduce some related quasiprobability distributions. First, let us consider:

$$\begin{aligned} \langle \hat{O} \rangle &= \mathrm{Tr}\left[\hat{O}\rho\right] \\ &= \frac{1}{(\pi)^m} \int \mathrm{d}^{2m}\boldsymbol{\alpha} \left\langle \boldsymbol{\alpha} \right| \rho \left| \boldsymbol{\alpha} \right\rangle f^{(A)}(\boldsymbol{\alpha}) \\ &= \int \mathrm{d}^{2m}\boldsymbol{\alpha} Q_\rho(\boldsymbol{\alpha}) f^{(A)}(\boldsymbol{\alpha}), \end{aligned} \tag{2.83}$$

where $f^{(A)}(\boldsymbol{\alpha})$ is found from the anti-normally-ordered polynomial $f(\hat{\mathbf{a}}, \hat{\mathbf{a}}^\dagger)$ by replacing $\hat{a}_i$ with α_i and $\hat{a}_i^\dagger$ with α_i^*. The introduction of this function isn't arbitrary – it relates to the optical equivalence theorem, which relates expectation values in the Hilbert space with expectation values in a particular quasi-probability distribution. The coefficient of $1/\pi$ is necessary to account for the overcompleteness of coherent states as a basis.

We have introduced the Q-function:

$$\begin{aligned} Q_\rho(\boldsymbol{\alpha}) &= \frac{1}{\pi^m} \left\langle \boldsymbol{\alpha} \right| \rho \left| \boldsymbol{\alpha} \right\rangle \\ &= W_{-1}(\boldsymbol{\alpha}) \end{aligned} \tag{2.84}$$

using the relationships in Eq. 2.78. The (Husimi) Q-function is not negative, nonetheless it is still a quasiprobability distribution and not a valid probability distribution, as different coherent states are non-orthogonal.

We can also introduce the P-function:

$$P_\rho(\boldsymbol{\alpha}) = W_1(\boldsymbol{\alpha}), \tag{2.85}$$

where:

$$\rho = \int \mathrm{d}^{2m}\boldsymbol{\alpha} P(\boldsymbol{\alpha}) \left|\boldsymbol{\alpha}\right\rangle\!\left\langle\boldsymbol{\alpha}\right|. \tag{2.86}$$

Proving these forms of the P- and Q-function is more involved, and is described in e.g. [132]. We use some textbook results about the overlap of coherent states, but more important, the Fourier-Weyl transform, which is yet another way of expressing states in the coherent basis.

For projector $\hat{P}$ of a POVM, we have:

$$\begin{aligned} \mathrm{Tr}\left[\rho\hat{P}\right] &= \mathrm{Tr}\left[\hat{\rho}\int \mathrm{d}^{2m}\boldsymbol{\alpha} P_{\hat{P}}(\boldsymbol{\alpha})\,|\boldsymbol{\alpha}\rangle\langle\boldsymbol{\alpha}|\right] \\ &= \int \mathrm{d}^{2m}\boldsymbol{\alpha} P_{\hat{P}}(\boldsymbol{\alpha})\,\langle\boldsymbol{\alpha}|\,\rho\,|\boldsymbol{\alpha}\rangle \\ &= \pi^m \int \mathrm{d}^{2m}\boldsymbol{\alpha} P_{\hat{P}}(\boldsymbol{\alpha}) Q_\rho(\boldsymbol{\alpha}), \end{aligned} \tag{2.87}$$

which we can use to find the probabilities of certain outcomes.

We can understand the P-function and Q-function as playing different roles in calculating measurement probabilities, based their relationship to different orderings. Normal ordering is useful for describing the operator that we want to measure, and we write the state in anti-normal ordering to 'balance' this (resulting from the optical equivalence theorem). This motivates all this rewriting – we can utilise the properties of different functions to simplify our calculations, even though the output may not seem as intuitive as when using the DV formalism.

2.7.2 Gaussian states

We are particularly interested in the class of states involved in linear optical transformations. We focus on the class of Gaussian states, which are generated by Hamiltonians that are at most quadratic in the creation and annihilation operators, as in Eq. 2.32.

A Gaussian function of multiple variables takes the following form:

$$G(\mathbf{x}) = C\exp\left(-\frac{1}{2}\mathbf{x}^T A\mathbf{x} + \mathbf{b}\cdot\mathbf{x}\right). \tag{2.88}$$

This is the moment generating function; differentiating n times with respect to $\mathbf{x}$ and setting $\mathbf{x} = 0$ gives the n-th moment. The first moment is the expected value and the second moment is the variance, and a Gaussian distribution is defined entirely by these. It is defined entirely by the covariance matrix A, showing the variances and covariances between variables, and the vector of means $\mathbf{b}$. Similarly, we will be considering characteristic functions of the form:

$$\chi(\boldsymbol{\xi}) = \exp\left(-\frac{1}{2}(\Omega^T\boldsymbol{\xi})^T V(\Omega^T\boldsymbol{\xi}) - i(\Omega\mathbf{d})^T\boldsymbol{\xi}\right). \tag{2.89}$$

Conventions occasionally differ by a factor of 2, as noted in Appendix A.

These can be fully defined by the first and second moments, the displacement vector $\mathbf{d} = \langle\hat{\mathbf{r}}\rangle$ and the covariance matrix $V_{ij} = \frac{1}{2}\langle\{\hat{r}_i - d_i, \hat{r}_j - d_j,\}\rangle$ respectively (where $\{\cdot,\cdot\}$ represents the anti-commutator). Alternatively, we can use the complex basis, which is

defined by the operations on the vectors $\hat{\mathbf{a}}, \hat{\mathbf{a}}^\dagger$ instead of $\hat{\mathbf{x}}, \hat{\mathbf{p}}$, with covariance matrix σ and displacement vector $\boldsymbol{\delta}$:

$$\boldsymbol{\delta} = F\mathbf{d} \tag{2.90}$$

$$\sigma = FVF^\dagger \tag{2.91}$$

$$F = \frac{1}{\sqrt{2}}\begin{pmatrix} \mathbb{1} & i\mathbb{1} \\ \mathbb{1} & -i\mathbb{1} \end{pmatrix}. \tag{2.92}$$

For an m-mode Gaussian state, the covariance matrix is dimension $2m \times 2m$, and the displacement vector is dimension $2m$. The uncertainty relation in Eq. 2.20 requires $\sigma + \frac{i}{2}\Omega \geq 0$, which also means σ must be positive definite. See [132] for further details.

The s-ordered, complex-basis characteristic equation for Gaussian states is:

$$\begin{aligned} \chi_s(\boldsymbol{\alpha}) &= \exp\left(-\frac{1}{2}(\Omega^T\boldsymbol{\alpha})^\dagger F\sigma F^\dagger(\Omega^T\boldsymbol{\alpha}) - i(\Omega F^\dagger\boldsymbol{\delta})^\dagger\boldsymbol{\alpha}\right)\mathrm{e}^{s|\boldsymbol{\alpha}|^2/2} \\ &= \exp\left(-\frac{1}{2}\boldsymbol{\alpha}^\dagger(\Omega F)\sigma(\Omega F)^\dagger\boldsymbol{\alpha} + \boldsymbol{\delta}(\Omega F)\boldsymbol{\alpha}\right)\mathrm{e}^{s|\boldsymbol{\alpha}|^2/2} \\ &= \exp\left(-\frac{1}{2}\boldsymbol{\alpha}^\dagger(\Omega F)(\sigma - \frac{s}{2}\mathbb{1})(\Omega F)^\dagger\boldsymbol{\alpha} + \boldsymbol{\delta}(\Omega F)\boldsymbol{\alpha}\right). \end{aligned} \tag{2.93}$$

Therefore, we can consider the quasiprobability distributions:

$$\begin{aligned} W_s(\boldsymbol{\alpha}) &= \frac{1}{\pi^{2m}}\int \mathrm{d}^{2m}\boldsymbol{\beta}\mathrm{e}^{(\boldsymbol{\alpha}\cdot\boldsymbol{\beta}^* - \boldsymbol{\alpha}^*\cdot\boldsymbol{\beta})}\exp\left(-\frac{1}{2}\boldsymbol{\beta}^\dagger(\Omega F)(\sigma - \frac{s}{2}\mathbb{1})(\Omega F)^\dagger\boldsymbol{\beta} + \boldsymbol{\delta}(\Omega F)\boldsymbol{\beta}\right) \\ &= \frac{\exp\left(-\frac{1}{2}(\boldsymbol{\delta}-\boldsymbol{\alpha})^T(\sigma - \frac{s}{2}\mathbb{1})^{-1}(\boldsymbol{\delta}-\boldsymbol{\alpha})\right)}{\sqrt{\det\left(\pi(\sigma - \frac{s}{2}\mathbb{1})\right)}} \end{aligned} \tag{2.94}$$

which is found using the identity:

$$\int_{\mathbb{R}^{2n}} \mathrm{d}\mathbf{r}\mathrm{e}^{-\mathbf{r}^T A\mathbf{r} + \mathbf{r}^T\mathbf{b}} = \frac{\pi^n}{\sqrt{\det(A)}}\mathrm{e}^{\frac{1}{4}\mathbf{b}^T A^{-1}\mathbf{b}}. \tag{2.95}$$

Symplectic transformations, as defined in Eq. 2.58, have the following effect:

$$\sigma \mapsto M^T\sigma M,\ \boldsymbol{\delta} \mapsto M\boldsymbol{\delta} \tag{2.96}$$

and therefore take Gaussian states to Gaussian states.

Gaussian states, and Gaussian operations (including measurements that project onto Gaussian states) have positive Wigner functions. Therefore, these behave as classical probability distributions. They are also classically efficient to simulate [136, 137], and in order to produce a quantum advantage, non-Gaussian resources (such as non-vacuum Fock states), either in state preparation or in the measurements performed, must be present (although this is not sufficient). This can be considered analogous to the Gottesman-Knill theorem, that Clifford operations alone are efficiently classically simulable. Resource theories are a field of study that discuss these requirements further.

2.7.3 Implementation

As shown in Eq. 2.32, we will only be focused on operations in linear optics which are described by Hamiltonians at most quadratic in the creation and annihilation operators. So far we have seen displacement, which is generated by the Hamiltonian term $(\alpha\hat{a}^\dagger + \alpha^*\hat{a})$. We distinguish between passive transformations, which preserve photon number, and active transformations, like displacement.

Another active transformation that generates photons that we will consider is squeezing. This is implemented through the operator:

$$\hat{S}(\zeta) = \exp\left(\frac{1}{2}(\zeta\hat{a}^{\dagger 2} - \zeta^*\hat{a}^2)\right). \tag{2.97}$$

When ζ is real, we can see that this is equivalently generated by $\hat{x}\hat{p}$ (up to a phase).

In comparison to the coherent or Fock states, which minimise the phase or photon number uncertainty, squeezed states $|\zeta\rangle = \hat{S}(\zeta)\,|\text{vac}\rangle$ minimise the uncertainty in the quadratures $\hat{x}$ or $\hat{p}$, or some linear combination, depending on the squeezing angle (given by θ in $\zeta = r\text{e}^{i\theta}$). Any Gaussian state can then be created by a combination of the active transformations, squeezing and displacement, acting on the vacuum, followed by passive transformations (a linear interferometer).

We are now in a situation, similar to that discussed in Section 2.1.1, where we may be tempted to store information in terms of continuous variables (like analogue computers). However, as before, the precision, accuracy and – particularly important in quantum computing – error correction possibilities of the computer are improved by using discrete variables.

On the other hand, if that is the case, it may seem pointless to go beyond the use of Fock states, which we have already considered. As we will see, there are many advantages to using the continuous variable (CV) framework – the first of which we will discuss further in the next chapter, which is that this framework more closely matches the currently used experimental procedure for generating photons in the lab. We also find that qubit encodings used in CV quantum computing are more resilient to different types of errors. Error correction is introduced by adding redundancy into the system; this can be done by using a code that has each logical qubit represented by multiple physical qubits, but it can also be done by using a qubit encoding that has some inherent redundancy, such as bosonic codes, which make use of the CV encoding [138].

For example, qubits can be encoded in the GKP (Gottesman-Kitaev-Preskill) state [139, 140], which are useful in correcting small, continuous displacement errors. There are several different (equivalent) ways to define GKP states. Given the real symplectic group in 2 dimensions $\mathbf{S} = \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix}$, GKP states are the eigenstates of

$$\hat{S}_x = \text{e}^{i(2\sqrt{\pi})(S_{11}\hat{x} + S_{21}\hat{p})}, \tag{2.98}$$

$$\hat{S}_p = \mathrm{e}^{i(2\sqrt{\pi})(S_{12}\hat{x}+S_{22}\hat{p})}. \tag{2.99}$$

Alternatively they can be described as the simultaneous +1 eigenstates of $\hat{D}(2\alpha)$ and $\hat{D}(2\beta)$, where

$$\beta\alpha^* - \beta^*\alpha = i\pi. \tag{2.100}$$

These represent idealised GKP states, which are not physically realisable as they have infinite energy. Instead, some approximation is required, which imparts some inherent error to the system. Additionally, the other difficulty of preparing GKP states is that their preparation requires non-Gaussian resources.

In this thesis, we will not be using a qubit encoding of CV states, but instead directly dealing with the covariance matrices and displacement vectors, where the only non-Gaussian resource is the final measurement. As we will see, this is an experimentally accessible framework, and we are interested in what can be achieved using this.

2.8 Gaussian boson sampling

Let's return to boson sampling, our problem of interest. Realistically, the generation of quantum states of light can be done in many different ways, but a useful and easily accessible way of doing this uses spontaneous parametric down conversion (SPDC) [141], e.g. using the nonlinear optical properties of silicon. This produces photon pairs non-deterministically, so the presence of a single photon is heralded by the detection of its partner. On the other hand, if each source has some probability p of producing a single photon, producing n single photons in this way scales as p^n and is therefore exponentially difficult, and is a major barrier to the implementation of boson sampling.

Therefore, a variation known as scattershot boson sampling was proposed [142, 143]. Several different spontaneous sources are simultaneously pumped, with the photons in each pair entering separate modes. One set of modes are measured to herald the presence of photons in the other set of modes, which then enter an interferometer, implementing boson sampling with a random input state.

What happens if we instead input a Gaussian state into a boson sampling setup (i.e. a passive linear interferometer, followed by PNR measurements)? This was proposed in [144] and further explored in [145]. We also note helpful descriptions in [81, 132, 146–148].

We first use state generation with squeezing and displacement operations on the vacuum, followed by a passive linear interferometer. Until the measurement stage, this uses only Gaussian operations, so is entirely simulable – we can consider this formalism to be equivalent to preparing a Gaussian state, described by σ and $\boldsymbol{\delta}$, followed by the measurement stage, which is where there is the potential for quantum advantage from the non-classically-simulable step, as shown in Fig. 2.7. The original proposal suggests that displacement is zero, but for

full generality, let us consider the outcome statistics of making PNR measurements on an arbitrary Gaussian state.

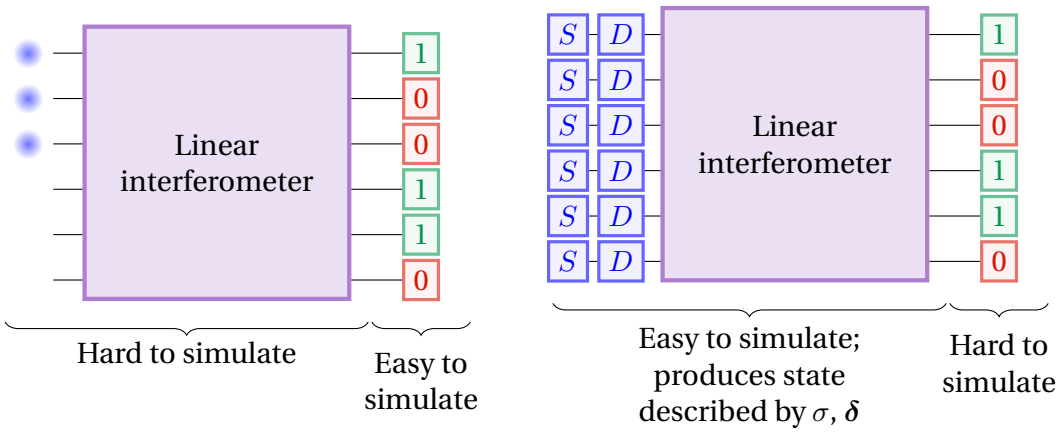


Figure 2.7: Comparison between the boson sampling (left) framework, in which single photons are input to the interferometer, and the Gaussian boson sampling (right) framework, in which states are created by squeezing and displacement (in full generality, we can assume this applies to all states, although in the framework of [144] this need only be some subset $K \leq m$). In BS, it is difficult to simulate the unitary evolution of Fock states (although outcome probabilities can then be calculated using the Born rule), whereas in GBS, the difficulty of simulation comes from applying non-Gaussian PNR measurement.

Once again, we are interested in the probability of measurement patterns $\mathbf{n}$, $p(\mathbf{n})$. We can use Eq. 2.87, so we need to find the P function of the projector $|\mathbf{n}\rangle\langle\mathbf{n}|$, using Eq. 2.82:

$$P_{|\mathbf{n}\rangle\langle\mathbf{n}|}(\boldsymbol{\alpha}) = \prod_{i=1}^{m} \mathrm{e}^{|\alpha_i|^2} \partial_{\alpha_i}^{n_i} \partial_{\alpha_i^*}^{n_i} \delta^2(\alpha_i). \tag{2.101}$$

Therefore we need to do the integral:

$$\begin{aligned} p(\mathbf{n}) =& \pi^m \int d\boldsymbol{\alpha} Q_{\hat{\rho}}(\boldsymbol{\alpha}) P_{|\mathbf{n}\rangle\langle\mathbf{n}|}(\boldsymbol{\alpha}) \\ =& \frac{1}{\mathbf{n}!\sqrt{|\sigma_Q|}} \int \mathrm{d}\boldsymbol{\alpha} \exp\left(-\frac{1}{2}(\boldsymbol{\alpha}^\dagger - \boldsymbol{\delta})\sigma_Q^{-1}(\boldsymbol{\alpha} - \boldsymbol{\delta})\right) \prod_{i=1}^{m} \mathrm{e}^{|\alpha_i|^2} \partial_{\alpha_i}^{n_i} \partial_{\alpha_i^*}^{n_i} \delta^2(\alpha_i) \\ =& \frac{\exp\left(-\frac{1}{2}\boldsymbol{\delta}^\dagger \sigma_Q^{-1} \boldsymbol{\delta}\right)}{\mathbf{n}!\sqrt{|\sigma_Q|}} \\ & \times \prod_{i=1}^{m} \partial_{\alpha_i}^{n_i} \partial_{\alpha_i^*}^{n_i} \exp\left(-\frac{1}{2}\boldsymbol{\alpha}^\dagger(\sigma_Q^{-1} - \mathbb{1})\boldsymbol{\alpha}\right) \exp\left(\frac{1}{2}\left(\boldsymbol{\alpha}\sigma_Q^{-1}\boldsymbol{\delta} + \boldsymbol{\delta}^\dagger \sigma_Q^{-1}\boldsymbol{\alpha}\right)\right)\Big|_{\boldsymbol{\alpha}=\mathbf{0}}, \end{aligned} \tag{2.102}$$

where we have introduced $\sigma_Q = \sigma + \mathbb{1}/2$.

At this point it will be convenient to introduce the A matrix:

$$A = \begin{pmatrix} 0 & \mathbb{1}_n \\ \mathbb{1}_n & 0 \end{pmatrix} (\mathbb{1}_{2n} - \sigma_Q^{-1}). \tag{2.103}$$

Hence we can rewrite Eq. 2.102 as:

$$p(\mathbf{n}) = \frac{\exp\left(-\frac{1}{2}\boldsymbol{\delta}^\dagger \sigma_Q^{-1}\boldsymbol{\delta}\right)}{\mathbf{n}!\sqrt{|\sigma_Q|}} \prod_{i=1}^{m} \partial_{\alpha_i}^{n_i}\partial_{\alpha_i^*}^{n_i} \exp\left(-\frac{1}{2}\boldsymbol{\alpha}^* A\boldsymbol{\alpha}\right)\exp(\boldsymbol{\gamma}\boldsymbol{\alpha})\Bigg|_{\boldsymbol{\alpha}=\mathbf{0}}, \tag{2.104}$$

where we have also introduced the (row) vector $\boldsymbol{\gamma} = \boldsymbol{\delta}^\dagger \sigma_Q^{-1}$.

We now need to perform the derivatives. Once again we will find it useful to define new matrices and vectors, where $A_{\mathbf{n}}$ indicates that the columns and rows have been repeated according to $\mathbf{n}$, and $\boldsymbol{\gamma}_{\mathbf{n}}$ indicates that the element i and $i+m$ of $\boldsymbol{\gamma}$ have been repeated according to n_i. Hence we can rewrite Eq. 2.104 as:

$$p(\mathbf{n}) = \frac{\exp\left(-\frac{1}{2}\boldsymbol{\delta}^\dagger \sigma_Q^{-1}\boldsymbol{\delta}\right)}{\mathbf{n}!\sqrt{|\sigma_Q|}} \prod_{i=1}^{2n} \partial_{\alpha_i} \exp\left(-\frac{1}{2}\boldsymbol{\alpha}_{\mathbf{n}}^* A_{\mathbf{n}}\boldsymbol{\alpha}_{\mathbf{n}} + \boldsymbol{\gamma}_{\mathbf{n}}\boldsymbol{\alpha}_{\mathbf{n}}\right)\Bigg|_{\boldsymbol{\alpha}=\mathbf{0}} \tag{2.105}$$

where i now runs over all the photons in the measurement outcome.

In order to carry out the derivatives we can apply Faà di Bruno's formula [149]:

$$\prod_{i=1}^{n} \partial_{x_i} \exp(y) = \exp(y) \sum_{\pi} \prod_{B\in\pi} \left(\prod_{j\in B} \partial_{x_j}\right) y, \tag{2.106}$$

in which π runs over all partitions of the set of $\{x_i\}$, with B being one subset of a partition. For example, $\pi \ni (12)(34)$, $B \ni (12)$. We see that the surviving terms are only partitions of the indices into pairs and single terms, where the pair terms contribute $A_{i,j}$ and single terms contribute $\boldsymbol{\gamma}_i$.

First let us consider the case in which $\boldsymbol{\gamma} = \mathbf{0}$, and hence only the pair terms contribute. In that case we are left with the hafnian function:

$$\text{haf}(A) = \sum_{\mu\in\mathcal{M}} \left(\prod_{k=1}^{|S|} A_{\mu_{2k-1},\mu_{2k}}\right). \tag{2.107}$$

If the displacement vector is non-zero, the output probabilities of GBS when the sources include displacements, as well as squeezing, are given by:

$$p(\mathbf{n}) = \frac{\mathrm{e}^{-\frac{1}{2}\boldsymbol{\delta}^\dagger \sigma_Q^{-1}\boldsymbol{\delta}}}{n_1!\ldots n_m!\sqrt{|\sigma_Q|}} \text{lhaf}\left(\text{filldiag}(A_{\mathbf{n}}, \boldsymbol{\gamma}_{\mathbf{n}})\right), \tag{2.108}$$

in which filldiag$(X, \mathbf{v})$ replaces the diagonal elements of the matrix X with the elements of the vector $\mathbf{v}$. We have also introduced the loop hafnian:

$$\begin{aligned}\text{lhaf}\left(\text{filldiag}(A,\gamma)\right) =& \text{haf}(A) + \sum_{i_1,i_2,i_1\neq i_2} \gamma_{i_1}\gamma_{i_2}\text{haf}(A_{S-\{i_1,i_2\}}) \\ &+ \sum_{i_1,i_2,i_3,i_4,i_1\neq i_2\neq i_3\neq i_4} \gamma_{i_1}\gamma_{i_2}\gamma_{i_3}\gamma_{i_4}\text{haf}(A_{S-\{i_1,i_2,i_3,i_4\}}) + \ldots + \prod_{i}^{M}\gamma_i.\end{aligned} \tag{2.109}$$

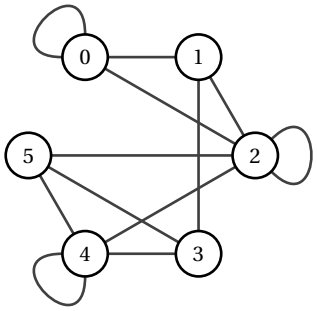


FIGURE 2.8. An example 6-vertex graph with loops.

Note that $S - \{i_1, i_2\}$ is the set of modes in which photons were measured except j_1 and j_2.

For a graph G that contains loops – edges that connect a vertex to itself, which occupy the diagonal elements of the adjacency matrix – this is:

$$\text{lhaf}(G) = \sum_{\mathcal{M} \in \text{SPM}(G)} \prod_{(i,j)\in\mathcal{M}} G_{(i,j)}, \tag{2.110}$$

in which SPM(G) is the set of perfect matchings that permit matching a vertex to itself - equivalently, a set of edges in the graph such that each vertex is connected to exactly one edge (assuming that a loop connects to a vertex once). Fig. 2.8 shows a graph with loops, and the perfect matchings are shown in Fig. 2.9.

If the state is pure, then $A = B^* \oplus B$. We can then describe the probabilities in Eq. 2.108 in terms of the B matrix, noting that $\text{haf}(A) = |\text{haf}(B)|^2$. It is also useful to note that $A = URU^T$, where U describes the interferometer and R described the initial squeezing. This provides further intuition behind the use of the hafnian: roughly speaking, the permanent is used in BS to find the different ways that input patterns can lead to output patterns, whereas the hafnian in GBS is used to connect pairs of photons that are created by the same sources (as squeezing generates photons in pairs). This 'retrodictive' intuition is explored further in [81], and is the basis for the algorithm in [150].

Let's consider a bit more where the A matrix comes from. We can see where the relationship $A = URU^T = X(\mathbb{1} - \sigma_Q^{-1})$ comes from (largely based on the description in [81]). We'll ignore displacement, so the covariance matrix will look like:

$$M_U M_S \sigma_{\text{vac}} M_S^\dagger M_U. \tag{2.111}$$

The inverse of this (see [81] for further details) is:

$$\sigma_Q^{-1} = \begin{pmatrix} \mathbb{1} & -URU^T \\ -U^* R^* U^\dagger & \mathbb{1} \end{pmatrix}. \tag{2.112}$$

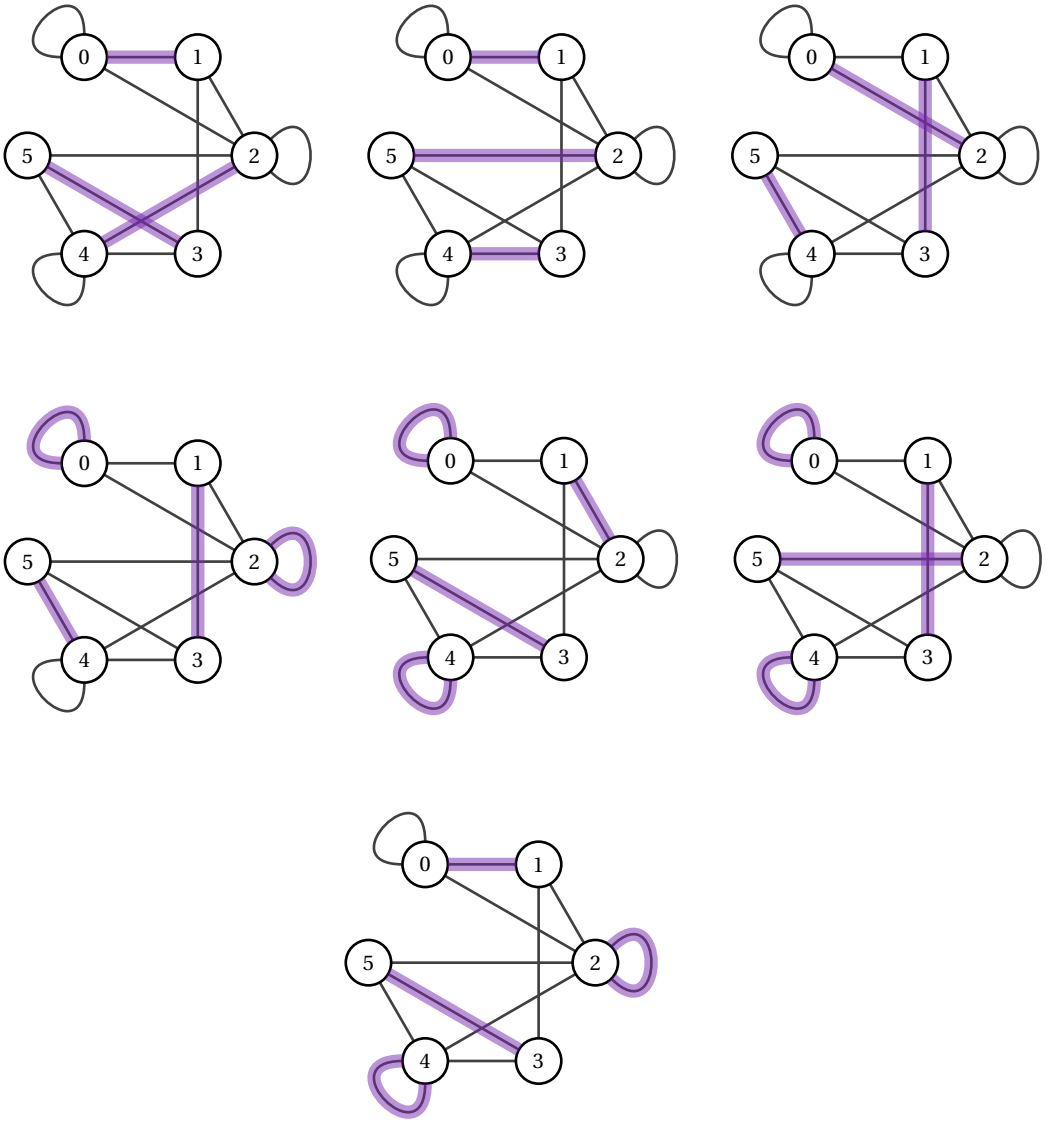


FIGURE 2.9. The perfect matchings of a graph with loops.

This form may still be somewhat unintuitive. Firstly, we would like to use normally-ordered expressions of the creation and annihilation operators. By the optical equivalence form, this means we want to consider the expression of the state in terms of the anti-normally ordered quasiprobability distribution, so we work with σ_Q. The form $\mathbb{1} - \sigma_Q^{-1}$ comes from the relationship between the characteristic function and the quasiprobability distribution (using the inverse is the same case when considering a multivariate normal distribution, although the identity comes from commutation relations). Finally, the permutation matrix X is useful because it allows us to take the complex conjugate of one side.

Instead of using PNR detectors, we can use threshold detectors, which do not count photons in a particular mode but only indicate whether there are any present (which is indicated by a 'click'). One method of doing this is using the Torontonian [151], an alternative matrix function similar to the hafnian. We will consider an alternative method in Section 3.4.2, which requires us to calculate the probability of measuring the vacuum in a subset of modes, S. The P function for $|\text{vac}\rangle\langle\text{vac}|$ is simply 1, so $\text{Tr}[\rho\,|\text{vac}\rangle\langle\text{vac}|] = \pi^{|S|}Q_\rho(\mathbf{0})$, and

$$p_{\text{vac}}(S) = \frac{\exp\left(-\frac{1}{2}\boldsymbol{\delta}^T(\sigma_Q^{-1})_S\boldsymbol{\delta}\right)}{\sqrt{\det((\sigma_Q)_S)}}, \tag{2.113}$$

where the subscript S indicates that we only keep the columns or rows associated with the elements of S.

The form of Eq. 2.108 motivates the use of the A matrix (and $\boldsymbol{\gamma}$ vector) in graphical representations of Gaussian states, and their manipulations [152]. The theory of matchings on a graph is closely related to the monomer-dimer model [153]. 'Dimers' can be placed on the edges of a weighted graph, so that no vertex has more than one dimer, and any uncovered vertices remaining are known as monomers. This represents, for example, simple two-atom molecules being placed on a surface so that the atoms of each molecule are aligned with the atoms on the surface.

This relationship with graphs leads to several applications. One example is dense subgraph finding (and the special case of maximum clique identification), which will be explored further on in this thesis. Additionally, there are applications in graph similarity and point processes [154]. A review of GBS applications is given in [155].

A further example of a potential use of GBS is for the simulation of molecular vibronic spectra [67]. This connection was made, as a proposal for quantum analogue simulation, due to the appearance of the permanent in the function describing these transitions. However, since then it has been identified that this is not an example of a use case with a potential for quantum advantage from GBS, as the way that outcome probabilities are grouped together can be exploited for efficient classical algorithms [156]. The original work highlights the possibility of extending the scheme using non-linearities (which would not be accessible using only Gaussian operations). As well as these applications, where GBS is used to do a calculation, or as a subroutine to a classical computation, a GBS-type framework can be used as a method to produce GKP states, which as a CV qubit encoding, form a fundamental building block of several proposed architectures [157].

Alongside the original proposal of scattershot boson sampling, there have been several other variations described. Often, it is required to adjust the complexity proof to compensate for the new scheme. For example, bipartite GBS is a scheme similar to scattershot boson sampling, but where both the signal and idler branches of the photon pairs created (via two-mode squeezing) pass through separate interferometers [158]. There is also higher

dimensional GBS [159], which uses loops of fibre to construct a scheme whereby photon packets can interfere at regular intervals.

The latter scheme was used in a proposed quantum advantage experiment by Xanadu [160], with a maximum photon count of 219 photons across 216 modes. This follows the Jiǔzhāng series of experiments at USTC, beginning with the exciting announcement in 2020 of the first claimed observation of quantum advantage with GBS (or indeed with photonics) [161], with 76 'clicks' (measurements of one or more photons) across 100 modes, and most recently has included experiments aimed at solving graph problems [162].

In each of the aforementioned experimental examples, the difficulty of classical simulation (and hence the basis for any claim of quantum advantage) comes from the amount of time it would take to return the sample with the highest photon count. Comparison to classical capabilities requires an accurate estimation of how long it would take to solve this task, which is more complicated than simply estimating how long it would take the best supercomputer to calculate a single, very large hafnian.

2.9 Simulation, emulation, and approximation

As we have seen previously, there are complexity-theoretic arguments for why (G)BS cannot be exactly simulated, or approximately simulated, efficiently by a classical computer. Nonetheless, this deals with asymptotic scaling in an idealistic case, and the interpretation of 'quantum advantage' generally has a more practical view. Therefore, we should consider how the real-world competition between classical and quantum computers plays out, given the advantages and disadvantages of either method; this view will be the main focus of this thesis.

For an arbitrary complex matrix, the state-of-the-art exact hafnian calculation method is from [163], which runs in time $O(n^3 2^{n/2})$, and can be generalised to loop hafnians. In the case of the permanent, the fastest known algorithm is due to Ryser, and runs in time $O(n2^{n-1})$ [164, 165].

On the other hand, there are special cases in which calculation, or approximation, is more efficient, by utilising the structure of the underlying graph or matrix [166]. In the case of a matrix that only has real, nonnegative entries, there is an efficient method for approximating the permanent [167]. As introduced in [168] and discussed further in [169], it is possible to use an approximation of the permanent to estimate the Hafnian of the same matrix, however the error of this scales exponentially, and thus it is still an open question whether the Hafnian of a nonnegative valued matrix can be approximated efficiently using a *fully* polynomial-time approximation scheme (FPTAS), where the complexity is polynomial in both n and $1/\varepsilon$, although this is widely thought to be possible. A further useful summary, and more information on these algorithms, is given at [170].

When the underlying matrix describing the (G)BS experiment is nonnegative valued, this is generally understood to correspond to lack of quantum interference effects, related to the positivity/negativity of the Wigner function, and so we would expect the experiment to be efficiently classically simulable. This is utilised by the algorithm in [150], which can efficiently sample from the distribution weighted by the Hafnian of submatrices (although not the Hafnian squared, which would be the case for a GBS device).

Nonetheless, as discussed previously, a (G)BS device does not itself efficiently calculate permanents (Hafnians), as an exponential number of samples would be required to estimate a particular outcome probability. Therefore, we instead consider the challenge of drawing a sample from the ground truth.

We define additive error when considering the approximation with error ε, i.e. $\|p-q\|_{TV} \leq \varepsilon$, where ε is fixed (although there may be some dependence on ε in the complexity of the algorithm). We will also consider multiplicative error, where for each individual probability p_x, $cp(x) \leq q(x) \leq p(x)/c$ (we assume $c \leq 1$), and additive error, $|p(x) - q(x)| \leq \varepsilon$.

It is not always clear how to classically draw a single sample from a target distribution. A naive method is to calculate the probability of all possible outcomes, and then roll a weighted die, however this generally involves the calculation of an exponential number of probabilities, all of which may themselves be difficult to compute. Therefore, we require sampling methods that do not need knowledge of the entire distribution. A more complete (and pedagogically valuable) introduction to sampling schemes for GBS can be found in [81].

The standard method for simulating GBS comes from the chain rule method of Clifford and Clifford [171], originally proposed for simulating standard boson sampling, which can be adapted for GBS [148, 172]. This method considers mode by mode, and requires the calculation of marginal probabilities - that is, the probability of measuring n_i photons in mode i, given a fixed measurement pattern in the preceding $i-1$ modes, and regardless of the possible measurement outcomes in the remaining modes. As the number of photons 'measured' in previous modes increases, finding the probability needed to weight later modes involves finding larger and larger hafnians, which is why samples with more photons take longer to be returned by a classical computer. Overall, the complexity of sampling n photons in m modes is $O(mn^3 2^{n/2})$, whereas the complexity of calculating an individual $n \times n$ hafnian scales as $O(n^3 2^{n/2})$ [163].

Alongside the methods that can utilise the underlying mathematical structure of the problem, we must consider the adaptations made to the experimental set up that may deviate from the ideal situation described. For example, threshold detectors are often used instead of PNR detectors, which can identify whether or not photons are present at the output of a mode, but cannot distinguish how many. For GBS, the Torontonian should be used in place of the hafnian, and for Fock state boson sampling with threshold detectors, it should be the Bristolian [133, 173]. A table comparing the algorithm efficiencies for different

matrix functions is given in [81]. Furthermore, recent work suggests that quantum advantage can still be achieved using a BS scheme where the number of modes scales linearly with the number of photons [126], where the hiding and collisionless properties do not apply, but providing evidence that average-case hardness of permanents for these matrices holds anyway.

Quantum advantage is not likely to be a single demonstration by a quantum computer of a task that is impossible for classical computers. By definition, a finite size experiment will not be able to prove the asymptotic advantage of quantum devices, and therefore we are now in an exciting era where the best quantum advantage experiments and the state-of-the-art classical algorithms are competing to demonstrate computational power. By exploiting errors or unwanted structure in the experiment, classical simulation methods that scale exponentially can still 'spoof' (give outcomes that are recognised as having better accuracy to the ground truth by some verification scheme) the outcomes of large experiments within a feasible amount of time, such as in [172]. Nonetheless, these methods would quickly become infeasible for larger system sizes.

Alternatively, it may be the case that efficiently simulable models are sufficiently accurate explanations of the results produced by the experiments. This is usually the case when there is a high level of error in the implementation, meaning that a classical model can spoof the data produced. There has been a great deal of work done in finding how much error can be tolerated before the experiment becomes susceptible to simulation. In particular, phase-space methods have been used to show the threshold of loss that means a thermal state, which can be sampled from efficiently [174], provides a multiplicative-error approximation [175]. Similarly, squashed states (squeezed thermal states) are particularly useful when there is a high incidence of collisions [176]. In these cases, scaling up the experiment would not be particularly helpful, as the spoofing method scales polynomially. However, there is a limit to the precision with which they can be used to model the quantum behaviour of the experiment. Therefore, reducing the error of the experiment is required to circumvent these classical simulation methods.

A recent paper using tensor network simulation methods to simulate large-scale GBS experiments addresses these shortcomings of previous spoofing methods [177]. This technique is able to separate the state into the 'classical part', which can be simulated efficiently, and the 'quantum part' (using the Williamson decomposition described in Section 5.6.1). Due to errors (chiefly losses) in the experiment, the ratio of photons produced from the simulation of the quantum system to the photons from the classical system is sufficiently small that the simulation is feasible for a classical computer, and it grows slowly as the number of modes increases (assuming the loss rate stays high). Furthermore, the precision of the inefficient quantum part can be increased with increased run time, so decreasing error rates does not immediately invalidate the simulation method. At the time of writing, this result represents

the latest, and most intimidating, opponent to quantum advantage using photonics.

Much of this thesis deals with the ways in which modifications to, or errors in, implementations of GBS affect their usefulness. GBS may not be the end point of photonic quantum computing, but it is an important stepping stone, and it is interesting to understand how to make best use of the quantum tools currently at our disposal.

CHAPTER 3

Effect of photonic errors on quantum-enhanced dense subgraph finding

It was full of ants. They scuttled along the tubing and through
complex little spirals in their thousands.
... Ponder sighed, 'we think it might be able to do other things.'
'Like what?' Ridcully demanded.
'Er, that's what we're trying to find out ...'
... 'Well, we think it might be able to do quite complicated maths. If
we can get enough bugs in it.'

Terry Pratchett, *Soul Music*

3.1 Preface

The emulator used to produce the data presented in this chapter is available at `https://gitlab.com/dualityqp/qgot_public/` (further details and links are given throughout the chapter).

The results of this chapter are published in Physical Review Applied: N. R. Solomons, O. F. Thomas and D. P. S. McCutcheon, *Effect of photonic errors on quantum enhanced dense-subgraph finding* Phys. Rev. Applied 20, 054043 — Published 21 November 2023, [1]. A poster describing this work is available on the Paraty 2023 website, and my talk from QCTiP 2023 is available on their YouTube channel.

Statement of work: This work was supervised by Dara P. S. McCutcheon, and used an emulator originally built by Oliver F. Thomas, which was maintained and improved by O.F.T.

and N.R.S. while being employed by Duality Quantum Photonics (including adding additional modules to create the data presented in this chapter).

N.R.S. carried out the majority of data collection and analysis presented in this chapter. O.F.T. and N.R.S. wrote the code to produce the data and D.P.S.M., O.F.T. and N.R.S. all contributed to interpreting the results. The emulator is based on methods published in [178], described in Section 3.4, and D.P.S.M. derived the expression for the optimal scaling parameter that is described in Section 3.4.3. Section 3.3 is a background section that reviews existing results and is not original work.

Acknowledgements: I am indebted to Oli for his support and contribution throughout this work, and in particular his patience in teaching me how to use the emulator. This work was carried out in collaboration with Duality Quantum Photonics, where Dara and Oli were both employees while working on this project, and where I was employed part time. Many thanks to everyone at DQP who gave their input to this work, and also for the use of their computing services, without which we would have struggled to collect all the necessary data.

I would also like to thank John Scott for useful discussions in constructing the emulator, and Ryan Mann for helpful discussions regarding computational complexity. Thank you to Patrick Yard for his comments on the paper that improved the presentation of this work, and Anthony Laing for useful discussions throughout the project.

3.2 Introduction

Gaussian boson sampling is interesting within quantum computing, as a framework for describing the power of quantum optics, and as a method for reaching the milestone (or, more accurately, the era) of quantum advantage. However, it has been more difficult to find use cases. The goal is to capture the asymptotic quantum advantage described by complexity theory, and utilise it to provide a benefit of using a quantum system instead of classical computing, prior to the added challenges of error correction, measurement and feedforward, or other non-Gaussian operations.

Given that the basic mathematical task of GBS is centered around sampling from Hafnians, graph theory problems are a major potential use case [155]. In this chapter, we focus on one of these in particular, dense subgraph finding, as described in [179].

The main goal of this chapter is to use numerical simulations to study how photonic errors impact the advantage to be gained from using GBS in a realistic implementation of these protocols, although we do not use experimental data. We consider two main sources of error. Firstly, the use of spectrally impure photon sources; this was chosen as it is less well-studied, and as we could use recent results from [178] to simulate this more efficiently, and therefore gain a new insight into the impact of this form of error. Secondly, we considered the impact of photon loss, for the opposite reason – it is much more well understood (including

some similar studies in [180]), as a primary source of error in photonic experiments, as well as how it impacts the simulability of the sampling from the resulting state, and it is therefore a useful standard of comparison.

As we will see, our results generally show that GBS is still effective for this application, even with high levels of error. This gives a promising outlook for the usefulness of earlier realisations of these devices. However, as we will discuss further, it also calls into question whether a quantum advantage beyond a polynomial speedup exists for this use case, and we consider how efficient classical algorithms with a similar performance could be constructed.

In Section 3.3.1, we discuss the graph theory problem of interest, dense subgraph finding, and in Section 3.3.2, classical algorithms to solve it. In Sections 3.3.3, 3.3.4 and 3.3.5 we consider how GBS can be used for this problem. In Section 3.3.6 we compare the computational complexity of various methods, and in Section 3.3.7 we briefly review other work on this topic.

The simulation methods which were used in this study are described in Section 3.4, and the results of our work are presented in Section 3.5 and discussed in Section 3.6.

3.3 Background

3.3.1 Dense subgraph finding

Given an input graph G, the densest k-subgraph problem (DkS) is defined as the problem of finding the subgraph of k vertices with the maximum density [181]. This is an **NP**-hard problem, as a generalisation of the **NP**-complete clique decision problem, which asks whether or not a graph contains a *clique* (fully connected graph) of a certain size. By finding the densest subgraph, and checking whether this is a clique, it is possible to solve the clique decision problem. Checking this only takes polynomial time - hence, if there is a polynomial solution to the DkS problem, then there is also a polynomial solution to the clique decision problem. There also exist several variations, such as finding the densest subgraph of arbitrary size, which admits a polynomial-time solution [182].

Dense subgraph finding has useful applications in many disparate fields, due to the wide-ranging usefulness of graphs. These include identifying emerging internet communities [183, 184], important news stories [185], and spam accounts [186] in web services; in finance, detecting booby traps in CDOs (collateralised debt obligations) [187]; and recognising interesting features of biological networks (which represent, for example, molecular interactions) [188].

One prominent application is for modelling molecular docking, an important step in the *in silico* stage of simulating the behaviour of drugs, which determines how two large molecules will fit together. This can be solved using the clique-finding problem [189]; as described in [190], the possible interactions of the two molecules are represented by a vertex-

weighted graph. The probability of sampling from the right cliques can then be improved using statistics from GBS. A similar method of searching for the maximum weighted clique can be used in predicting protein folding [191].

These example uses give an illustration of the relevant ranges of graph and subgraph size: in 1999, searching through internet pages involved looking at graphs with hundreds of millions of vertices while searching for subgraphs of 6 vertices, with similar parameters used in searching databases of tweets for emerging stories in 2012. Identifying protein complexes in interaction networks involves finding subgraphs of average size 6 from graphs of approximately 10,000 vertices [192], whereas molecular docking simulations model graphs with hundreds of vertices, with the optimal orientation represented by a subgraph consisting of tens of vertices [193]. Therefore, molecular docking stands out as one of the more eligible problems suitable for applications of GBS, within the regime of quantum advantage, but still within the realm of what is experimentally viable.

3.3.2 Classical algorithms for dense subgraph finding

As DkS is **NP**-hard, there are no polynomial time classical algorithms that are guaranteed to find the densest subgraph for any given graph (although some particular structures of graphs are efficiently solvable [194]). It has also been shown that there is no polynomial time approximation scheme (PTAS) for general graphs [195] – this means that, given some error bound ϵ, there does not exist an algorithm that can produce a subgraph with density that is within a factor of $(1-\epsilon)$ of the optimal density, that has complexity polynomial in n, where n is the number of vertices in the initial graph. Note that this definition doesn't place any restriction on the complexity of the algorithm in ϵ - that is, if the approximation scheme in polynomial in n but exponential in ϵ in order to find a solution within $(1-\epsilon)$ of the optimal solution, it's still a PTAS. We can even have $n^{1/\epsilon}$, because this is polynomial in n for a fixed ϵ. It's important to note the difference between this and an FPTAS, a *fully* polynomial time approximation scheme, that also requires the complexity to be polynomial in $1/\epsilon$. Hence, not permitting a PTAS is an even stronger hardness than not permitting an FPTAS (which is what we were interested in, in the case of (G)BS).

Also, note that we're using $(1-\epsilon)$ here because it's a maximisation problem – a minimisation problem like the travelling salesman problem would aim to have a solution within a factor of $(1+\epsilon)$ of the optimum.

Exact approaches are therefore limited in the size of graph that they can tackle, with maximum graph sizes between 100 and 200 vertices. Strategies include enumeration algorithms such as branch-and-bound algorithms[1] (see [197] and [198] for reviews).

[1] Branch-and-bound is a class of algorithms for solving optimisation problems which can admit a near-quadratic quantum speedup [196].

Heuristic approaches, which provide solutions which may not be optimal, are more manageable for larger graph sizes. For example, the algorithm in [181] runs in polynomial time, and has approximation ratio (the factor by which the final subgraph density differs from the optimal subgraph density) of $O(n^\delta)$ for some $\delta < \frac{1}{3}$. One popular heuristic strategy is simulated annealing, a technique for solving optimisation problems (named after the process of heating and then gradually cooling a material to change its properties) [199]. This is an approach that starts with a random configuration and applies iterative improvements.

Another algorithm that we will use frequently is the 'greedy' algorithm, introduced in [200]. This is a simple algorithm that proceeds by iteratively removing the lowest degree vertex. If there are several vertices with the same degree, one is chosen randomly from this set - other than these steps, the algorithm proceeds deterministically. It is explored further in [201] as a method for calculating the maximum average degree[2] across subgraphs of any size, for which it is 2-optimal, meaning it has an approximation ratio of 2 (here, this is the ratio between the true value of the highest average degree, and the calculated value). This algorithm is particularly useful as a benchmark due to its ease of implementation. Many sources cite [201] as the source of this algorithm, as it contains a more thorough analysis, however this work itself references [200].

3.3.3 Hafnians and density

One might expect that the perfect matchings and density of a graph are related, which would be useful in relating GBS to the DkS problem, which is discussed in this section. The proceeding sections discuss how to use GBS as a basis for solutions to DkS. Using GBS for DkS is introduced in [179], and the background for the following sections is mostly based on this work, alongside [202] and [203]. These works give some results describing the speedup that could be expected in using GBS for DkS (over classical methods), the question which we aim to consider further in this chapter. The content of these references is as follows: [202] links graphs and Gaussian states and uses this to sample perfect matchings, [203] introduces the stochastic algorithms based on GBS, and [179] explores the link between perfect matchings and density, to show how these sampling algorithms can be used to find dense subgraphs.

As seen in Section 2.8, the outcome probabilities of GBS are calculated using the Hafnian function:

$$\text{haf}(A) = \sum_{\mu \in \mathcal{M}} \left(\prod_{k=1}^{|S|} A_{\mu_{2k-1},\mu_{2k}} \right). \tag{3.1}$$

[2]The average degree of a graph is defined as $2|E(G)|/|V(G)|$. Note that, for the DkS problem, $|V(S)| = k$ and hence the average degree is proportional to the density for any size-k subgraph S. However, for the dense subgraph problem where k is not fixed, this is not the case, and therefore (for example) a clique of size 4 is a better solution than a clique of size 3.

Recall that $\mathcal{M}$ is the set of perfect matching partitions of the set of indices S (the different ways of 'pairing' the indices, or every permutation in which $\mu_{2k} < \mu_{2(k+1)}$ and $\mu_{2k} < \mu_{2k+1}$).

As the number of edges increases, it seems intuitively that the number of perfect matchings should increase, as there are more ways to match vertices. Despite this, they are not perfectly correlated - for example, note that if the graph from Fig. 2.1 is modified so that the $\{0,1\}$ edge is deleted, and replaced with an edge at $\{2,3\}$, the density remains the same, but there is now only 1 possible perfect matching.

In [204], it is shown that (assuming that $|E(G)| \geq |V(G)| \geq 2$, and that G has an even number of vertices) the number of perfect matchings in graph G, $|M(G)|$, is upper bounded by a function of the number of edges[3]. For $m := |E(G)|$ and $u := |V(G)|/2$:

$$|M(G)| \leq \left(\left\lfloor \frac{m}{u} \right\rfloor !\right)^{\frac{u-\alpha}{\lfloor m/u \rfloor}} \left(\left\lceil \frac{m}{u} \right\rceil !\right)^{\frac{u}{\lceil m/u \rceil}}, \quad \alpha := m - u\left\lfloor \frac{m}{u} \right\rfloor. \tag{3.2}$$

Equality is reached (i.e. graphs with this structure have the most possible perfect matchings given a certain number of vertices and edges) for graphs that are a disjoint union of complete bipartite graphs made up of either $2j$ or $2(j+1)$ vertices, for some integer j. Recall that a bipartite graph is a graph made up of 2 sets of vertices, where every edge connects a vertex from each set, and there are no edges connecting vertices within sets [112].

A consequence of Eq. 3.2 is that, given a graph with a certain number of vertices and a certain number of perfect matchings, it is possible to obtain a lower bound for the density of this graph. Fig. 3.1 shows the maximum number of perfect matchings for a given density, for graphs of different sizes (using Eq. 3.2).

This alone does not mean that measuring perfect matchings can be used to find high density graphs (for example, in the DkS problem). Any real graph will have a number of perfect matchings lying below the lines indicated in Fig. 3.1, and therefore, measuring many perfect matchings is only possible for high density graphs, although there may be high density graphs that do not have many perfect matchings which would be overlooked. If there seems to be very little correlation between perfect matchings and density (so the majority of points lie below the line), then it may be that measuring a high Hafnian does not correspond to a dense graph. In [179] there is a numerical study on graphs of size 16 that gives evidence that the number of perfect matchings and the number of edges in a graph closely follows the trend of Fig. 3.1. This suggests that by choosing subgraphs with large Hafnians, we are more likely to find dense subgraphs.

3.3.4 Using GBS to find dense subgraphs

Consider a GBS experiment in the collision-free regime; that is, when using PNR detectors, all outcomes are either 0 or 1 photon(s). We consider states with $\boldsymbol{\delta} = \mathbf{0}$ (zero displacement).

[3] It is worth noting that the results in this paper generalise to loop Hafnians.

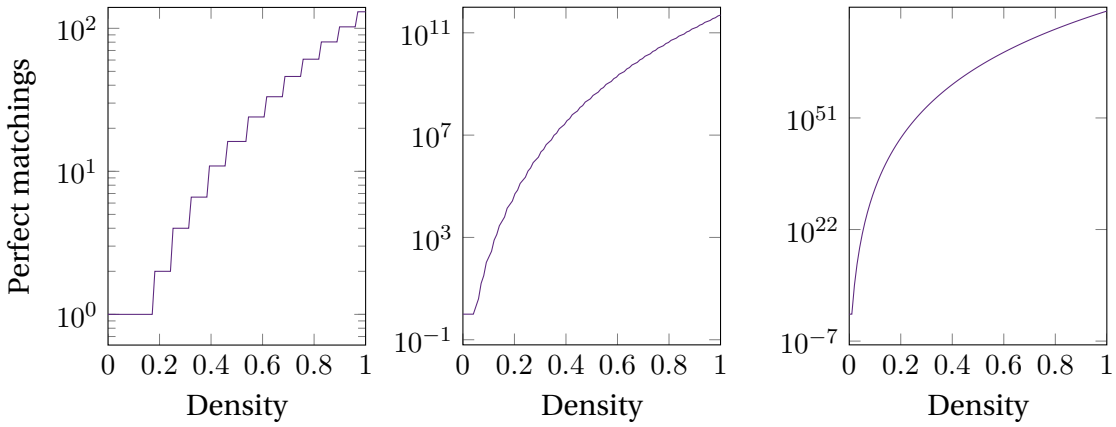


FIGURE 3.1. We consider the maximum number of perfect matchings for a given density for graphs of size 8 (left), 24 (middle) and 100 (right). The number of edges is chosen to be the even integer closest to the expected value for each density, then the number of perfect matchings is calculated according to Eq. 3.2. The 'staircase' look of this plot, for smaller graph sizes, comes from the way we plot the graph - for a particular density, we need to round to the nearest number of edges for that density. This means there are jumps in the maximum number of perfect matchings as the number of edges increases by integer values. Therefore, any graph is guaranteed to occupy the space below this line – that is, for a given number of perfect matchings, the density can be lower bounded as shown here.

As seen in Section 2.8, a measurement pattern $\mathbf{n}$ using PNR detectors has probability:

$$p(\mathbf{n}) = \frac{1}{n_1!\ldots n_m!\sqrt{|\sigma_Q|}}\text{haf}(A_{\mathbf{n}}). \tag{3.3}$$

In particular, when the input state is pure, the A matrix can be written as $A = B^* \oplus B$, in which case:

$$p(\mathbf{n}) = \frac{1}{n_1!\ldots n_m!\sqrt{|\sigma_Q|}}|\text{haf}(B_{\mathbf{n}})|^2. \tag{3.4}$$

Recall all the notation here: there are m modes, and $\mathbf{n}$ is a vector of size m, $\{n_1, n_2, \ldots n_m\}$. $\sigma_Q = \sigma + \mathbb{1}$ and $A_{\mathbf{n}}$ is A with the rows and columns repeated according to how many photons we measure in each mode.

Now consider that we construct a Gaussian state such that the B matrix is equal to the adjacency matrix $\mathcal{A}$ of an undirected, unweighted graph G (as introduced in [202]). Note that in other papers, $\mathcal{A}$ and A swap roles. However, as the A matrix is discussed widely in GBS literature, I chose to keep its meaning here, and use $\mathcal{A}$ for the adjacency matrix instead. Given that $\mathbf{n} \in \{0,1\}^m$, the submatrix of B given by $B_{\mathbf{n}}$ is equivalent to finding the adjacency matrix of a subgraph of G, in which vertices are selected based on modes in which photons have been measured.

Therefore, for each outcome in which the total photon number is $k \in \mathbb{N}$, the probability of this click pattern is proportional to the square of the Hafnian of the selected subgraph. That is, for an appropriately programmed GBS experiment, the more likely outcomes correspond to subgraphs with more perfect matchings – which are also likely to be the more dense subgraphs.

The covariance matrix of this Gaussian state can be found using $A = (\mathcal{A} \oplus \mathcal{A}) = (X \otimes \mathbb{1})(\mathbb{1} - \sigma_Q^{-1})$. We invert this to give:

$$\sigma = (\mathbb{1} - (X \otimes \mathbb{1})A)^{-1} - \mathbb{1}/2. \tag{3.5}$$

For a graph with n vertices, σ is size $2n \times 2n$, and thus the relevant GBS experiment has $m = n$ modes.

This does not necessarily produce a valid covariance matrix (this can be seen by noticing that Eq. 3.4 produces probabilities much greater than 1). We introduce a scaling parameter, $c \in \mathbb{R}$, so that $A = c(\mathcal{A} \oplus \mathcal{A})$. Note that, by Eq. 2.39, if the dimension of A is $2n \times 2n$, then $\mathrm{haf}(cA) = c^n \mathrm{haf}(A)$. Therefore, assuming that we are in the k-photon, collision-free subspace, we find that

$$p(\mathbf{n}) = \frac{c^k}{\sqrt{|\sigma_Q|}} |\mathrm{haf}(B_\mathbf{n})|^2. \tag{3.6}$$

The appropriate range of c to give a valid covariance matrix is derived in [202]. This discussion is reproduced in Appendix Section B for completeness. We find that $0 < c < 1/\lambda_{\max}$, where $\lambda_{\max}$ is the maximum eigenvalue of the adjacency matrix $\mathcal{A}$ (which is upper bounded by the number of vertices). Changing c also changes the squeezing value used, and should therefore be chosen to optimise the probability of drawing samples from the k-photon subspace. This is discussed further in Section 3.4.3.

Measurements involving collisions, in which multiple photons are present in the same mode, are typically assumed to be improbable, and often neglected (including in the original GBS proposal [144]). The probability of collisions depends on the relationship between the number of modes and the number of photons detected, hence will also depend on c, as increasing the squeezing will increase the average photon number. This is also discussed further in Section 3.4.3. If using PNR detectors, these events can be detected and therefore removed in postselection.

Importantly, the probability of measuring in the k-photon subspace, p_k, and the probability of measuring in the collision-free subspace, p_{cf}, depend on the parameters of your experiments, but then don't change between different measurement outcomes. As described in [179], we can express the outcome probabilities normalised over drawing a sample in the correct subspace ($p_{k \wedge cf}$):

$$P_{k \wedge cf}(\mathbf{n}) = \frac{c^k}{\sqrt{|\sigma_Q|}} \frac{|\mathrm{haf}(B_\mathbf{n})|^2}{p_{k \wedge cf}}. \tag{3.7}$$

Beware of what I believe is an error in [179]: Eq. 2 should say c^k and not c^2.

3.3.5 Sampling algorithms based on GBS

We now consider algorithms that use the samples taken from the GBS experiments like we have described, as considered in [179] and [203]. In particular, data from GBS are used to enhance stochastic algorithms, which use randomness as an essential part of their operation.

Firstly, we consider random sampling. Consider the completely classical DkS algorithm, Alg. 1. This contains the subroutine `Random-subgraph`, which chooses a random subset of size k of the vertices of G, and returns the induced subgraph. The algorithm takes the initial graph, G, the desired subgraph size, k, and the number of steps, a, as inputs. It then draws a random subgraphs of size k and keeps the one with the highest density. This is a very simple algorithm. After giving the inputs, we start by randomly choosing a subgraph, which we call 'Best'. This is our best subgraph so far, as it is the only one chosen. As we have completed our first step, we reduce a by 1. a will act as the counter for the number of steps remaining - it starts with the total number of steps we would like to do, and it is reduced by 1 every time we do an iteration. For each iteration, we pick a random subgraph which we call S, and if the density of S is higher than the density of Best, our best subgraph so far, we replace Best with S. Note that we could not include the first step in this while loop, because we did not start with a best subgraph to compare to. Once we finish all the steps, we return our best subgraph.

Algorithm 1 Classical random sampling for DkS.

G: input graph of size n (can be given by an adjacency matrix, $\mathcal{A}$)
$k \in \mathbb{N}, k \leq n$: required subgraph size
$a \in \mathbb{N}$: number of steps
Best = `Random-subgraph`(G,k)
$a \leftarrow a-1$
while $a \neq 0$ **do**
  S = `Random-subgraph`(G,k)
  if $D(S) > D(\text{Best})$ **then**
    Best $\leftarrow S$
  end if
  $a \leftarrow a-1$
end while
return Best

There are $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ possible subgraphs, therefore the likelihood of selecting the densest subgraph at random is low (particularly if there is only one that fulfills the maximum density, or very few). Nonetheless, this is simple to implement, and therefore it is useful to see how GBS performs in comparison to another, more naïve, sampling-based algorithm.

We can now consider an improved version that uses proportional sampling to return subgraphs with a probability that is proportional to the probability that the relevant measurement pattern would be given as an output to an appropriately programmed GBS device. We replace the `Random-subgraph` subroutine with `GBS-Explore`, based on the same subroutine as introduced in [203], which returns subgraphs according to the probability distribution defined by Eq. 3.4 (note that this subroutine takes as input c, G, and k, from which the appropriate B and σ_Q matrices can be constructed, and returns a subgraph chosen according to the set of modes in which photons are measured). This could be implemented by performing the appropriate GBS experiment, or simulating it. Depending on the number of samples drawn (a) and the size of the graph, classical simulations are likely to have lower overheads when drawing samples shot by shot, instead of calculating the entire probability distribution before sampling.

The second algorithm considered is simulated annealing, from [179]. Simulated annealing is a commonly used technique for optimisation problems. It is named after a technique that involves heating and cooling a metal to change its crystalline structure. As well as the subroutine `GBS-Explore`, simulated annealing requires `GBS-Tweak`. Note that there is a different `GBS-Tweak` subroutine given in the sister paper, [203]. This takes as input a subgraph S, as well as the parameter l (the minimum number of vertices of S to be left unchanged), (note that this has to be even so that we can sample it as a GBS outcome) and returns a different subgraph of the same size. It proceeds as follows:

1. We choose an l-vertex subgraph of S, denoted R, according to the GBS distribution. (This means drawing samples from the distribution $P(\mathbf{n}) \sim |\text{haf}(B_{\mathbf{n}})|^2$, where we normalise over the probability of drawing an l-photon, collision-free sample where all modes are in the set S, and then using these modes as the set of vertices of the subgraph. An experimental implementation would keep the same setup as was used to choose S, but postselect on drawing a sample in the correct subspace). In the classical equivalent simulated annealing algorithm, this is selected from the uniform distribution.

2. We generate a random number $m' \in \{0, 1, ..., k-l-1\}$ called m' instead of m to distinguish it from the number of edges, m. We choose m' vertices at random (both in the classical and quantum-enhanced versions of the algorithm) from $S \backslash R$ (the vertices of S that are not in R) and add these to R.

3. We draw a sample L, of size $k-l-m'$, from $G \backslash R$ according to the GBS distribution[4] (or from the uniform distribution in the classical version).

4. The final graph is the size k subgraph of G given by $R \cup L$.

[4]This is done as before, drawing a sample from the original distribution and postselecting on it having no collisions, $k-l-m'$ photons, and there being no overlap with the modes in R.

Note that our implementation of `GBS-Tweak` differs slightly from that introduced in [179]. In the originally proposed algorithm, the subgraph L is chosen by drawing a sample of size $k - l_{\text{min}}$ from the overall graph G, and then deleting m' vertices at random. If the subgraphs R and L then have some overlap, this step is repeated. When producing the following data, we found that the need to repeat this step many times made the computation time prohibitively long, hence L was drawn directly from the correct subspace. The subroutine retains the important elements of modifying the initial graph using GBS statistics (as well as a small amount of classical randomness), and adding vertices that are well connected with each other. Nonetheless, this modified subroutine may deviate slightly from the results of [179]. It also has a marginally higher reliance on GBS data instead of classical randomness.

Algorithm 2 Quantum-enhanced simulated annealing for DkS.

G: input graph of size n (can be given by an adjacency matrix, $\mathcal{A}$)
$k \in \mathbb{N},\ k \leq n$: required subgraph size
$a \in \mathbb{N}$: number of steps
$T \in \mathbb{R}_+$: initial temperature
$l \in \{0, 2, 4, ..., k-2\}$: minimum number of vertices to change in `GBS-Tweak`
$c \in (0, 1/\lambda_{\text{max}})$
$T_{\text{step}} = T/a$
$S =$ `GBS-Explore`(G, c, k) ▷ Classical: sample from uniform distribution
Best $= S$
while $a \neq 0$ **do**
  $R =$ `GBS-Tweak`(S, l, G) ▷ Classical: alternative implementation described previously
  if $D(R) > D(S)$ **then**
    $S \leftarrow R$
  else
    $S \leftarrow R$ with probability $\exp[(D(R) - D(S))/T]$
  end if

  if $D(S) > D(\text{Best})$ **then**
    Best $\leftarrow S$
  end if

  $T \leftarrow T - T_{\text{step}}$
  $a \leftarrow a - 1$
end while
return Best

Alg. 2 presents the simulated annealing algorithm. It starts by selecting a subgraph S (which may be a GBS-enhanced choice). Each successive step 'tweaks' this subgraph, producing the new subgraph R. If R has a greater density than S, it replaces S – if not, it still replaces S but with a probability dependent on the 'temperature'. This temperature is successively lowered in each step (the amount that it is lowered by is not specified in [179]; we choose this amount so that the temperature decreases to zero, as described in [205]). Like

in the random sampling algorithm, the output is the best subgraph of all of those found in any step of the algorithm, even though this might not be the subgraph the search ends up on. Therefore, the algorithm works by using random search or GBS statistics to explore different subgraphs, and using the 'hill climbing' technique to make slight improvements. It avoids getting stuck in local maxima by sometimes accepting tweaks to the subgraph that slightly reduce the density.

3.3.6 Complexity considerations

Thus far we have considered the complexity of simulating GBS, and the complexity of the dense subgraph finding problem. In this section we will consider the complexity of performing the stochastic algorithms described, and how to evaluate their success at solving the given problem.

First we consider drawing a sample of a fixed size (k). In the classical case, this amounts to choosing a subset of size k, which has an expected running time $O(k)$ [206]. In the quantum case, aside from the difficulty involved in preparing the experiment, (which, by my understanding, is essentially trivial), collecting samples at the output is constant in the system size.

We must also consider the complexity of finding the appropriate interferometer settings with which to programme the GBS device. That is, given an adjacency matrix of an initial graph, we must find the appropriate covariance matrix (using Eq. 3.5). We then use the Williamson decomposition to find the interferometer unitary and squeezing parameters, and the Bloch-Messiah decomposition to find the optical elements to carry out this unitary. These typically take $O(m^3)$ to carry out.

The complexity of deterministic algorithms is determined by the number of steps that are required to arrive at the result. Stochastic algorithms, like those used in this chapter, do not necessarily have a well-defined number of steps, and are typically analysed by looking at the infinite-time convergence, or expected values of the finite-time behaviour [205]. We will consider some run-time implications of stochastic methods in the following chapter, but for this chapter we focus on qualitative differences that can be observed with a finite number of steps.

Finally, we must consider the complexity of simulating GBS. It could be hoped that an application of GBS such as DkS could be a way of utilising the exponential difficulty of classical simulation in a useful and relevant quantum algorithm. (Although, given that the GBS framework would be a component of the larger architecture of a fault tolerant quantum computer such as that described in [157], and therefore are likely to be developed anyway, a polynomial advantage may still be worth investigating.) For that to be the case, the quantum-enhanced algorithms must rely on an appropriate implementation of GBS that cannot be efficiently simulated.

In particular, we note that we are concerned with those instances of GBS in which the A matrix has only real, positive elements (namely, 0 and 1). It is already known that the permanent of such a matrix can be efficiently approximated [169]. This reference gives an efficient estimator for the Hafnian, but within a factor that scales subexponentially and therefore cannot give an FPTAS. At the time of writing, it is not known whether the Hafnian of a positive-valued matrix can also be efficiently approximated – there is a connection between between hafnians and permanents [169], and therefore methods to efficiently estimate the permanent of these matrices can be useful, but we do not call them 'efficient' due to the exponential scaling of the error when used to estimate hafnians.

Importantly, this property is utilised in [150], which notes that, as the matrix is positive-valued, this suggests that there is no quantum interference occurring. The authors present an efficient classical algorithm which samples from the Hafnian of $A_{\mathbf{n}}$, and therefore can be used for the DkS problem. This work demonstrates that there is unlikely to be an exponential quantum advantage from using GBS over their efficient classical sampler. The authors also independently found similar results to those presented in this chapter regarding the impact of loss.

We also recall the results presented in Section 2.9. Namely, we expect there to be regimes of sufficient error in which the experimental results become efficient to approximate classically.

3.3.7 Other work

Alongside the initial proposal for using GBS for DkS, as presented in [155, 179, 203], the same principle can be applied to a quantum-enhanced algorithm for a closely related problem of finding the maximum weighted clique [190]. This can be used to predict molecular docking configurations, which is an important step of drug design.

Some work complementary to the results of this thesis is published in [207]. This includes further details of the construction of Gaussian states from graphs, and how they change under the impact of loss and spectral impurity.

There have been several experimental implementations of the protocol. This includes using the Jiŭzhāng experiment for Max-Haf and DkS [162], which, alongside the work presented in this chapter and [150], concurrently looked at the impact of loss (using theoretical studies), and the relative advantage of using GBS with increased problem size (we will be returning to this paper some more throughout this chapter). The time-bin encoded device in [208] was used for both molecular docking and protein folding prediction. A time-bin encoded device was also used for an earlier demonstration of dense subgraph finding [180]. This work also contains a theoretical analysis that considers the performance of the algorithm with loss, in particular considering the effect of changing the squeezing parameter to compensate for different loss values, noting that increasing the squeezing values leads to increased susceptibility to loss. We also note ongoing work in this group on the effect of

displacements for these problems.

Finally, recent work has also considered the possibility of using Fock-state boson sampling for graph problems [209]. Although the authors found the potential for some quantum speed-up in the case of the DkS problem, this framework suffers from the fact that in this case, the outcome probabilities are proportional to the squared permanent of a submatrix with rows chosen according to the output measurement pattern, but columns chosen according to the input measurement pattern. Therefore, the submatrix is not guaranteed to correspond to the adjacency matrix of a subgraph of the input graph.

3.4 Simulation methods

The data presented in this chapter were generated using the quantum Gaussian optics toolkit QGot, which is open source and available to use [210]. This is a C++ framework which uses the continuous variable formalism previously described to simulate quantum optics processes[5]. Further information, and a more detailed description of the development of the software, is given in [207]. At the time of writing, it is maintained by Duality Quantum Photonics. There is also a similar package available, written in Nim[6].

The sampling data used in this chapter were produced in the following way. Firstly, we chose an initial graph size ($n = 24$), a final graph size ($k = 8$), and an objective initial density (either $d = 0.2$ or $d = 0.4$). Using these parameters, we produced a random graph using the Erdős-Rényi model, which creates each potential edge between a pair of vertices with probability d. As this is probabilistic, it does not produce graphs with density exactly d, and hence the graphs which were actually used have densities 0.196 and 0.362. Using this graph, we can find the covariance matrix of the relevant quantum state using Eq. 3.5. This method is implemented in [210], in the `Graph` class, as `random(int num_vertices, double density, int seed_in)`.

3.4.1 Adding noise

In this chapter we consider the effect of two sources of noise - spectral impurity, and loss[7]. We assume uniform loss L across all modes, in which case a lossy channel with transmission $\eta = 1 - L$ being applied to the quantum state implements the transformation $\sigma \to \eta\sigma + \frac{1}{2}(1-\eta)\mathbb{1}$. There is an alternative simulation method, by which a variable beamsplitter is applied before measurements, coupling each mode to a vacuum ancilla mode, which is traced out.

We also want to consider the case in which the sources themselves are spectrally impure, although we assume that each source is identical (in spectral profile, although not necessarily

[5] Documentation is available at: `https://dualityqp.gitlab.io/qgot_public/index.html`.

[6] Documentation is available at: `https://ofthomas.gitlab.io/noptics/`.

[7] We consider these separately, although these sources of error can be combined using the software, and data related to both sources of error being applied concurrently are available upon request.

in brightness). In our model, the input to the interferometer is a single-mode squeezing operation on each spatial mode. Nonetheless, these squeezers do not emit photons perfectly at a particular frequency. The spectral profile of emitted photons is described by the squeezing Hamiltonian:

$$\hat{H} = \iint \mathrm{d}\omega_1 \mathrm{d}\omega_2 F(\omega_1, \omega_2) \hat{a}^\dagger(\omega_1) \hat{a}^\dagger(\omega_2) + h.c. \tag{3.8}$$

where $h.c.$ indicates the Hermitian conjugate term. Here, $F(\omega_1, \omega_2)$ is the joint spectral amplitude (JSA). This represents the correlation of the two emitted photons, which are usually referred to as the signal and idler. Here, we assume the sources are not too bright, so we can simply take the exponentiation of the squeezing operator to first order. If the JSA is nonseparable, then measurement of one photon collapses the other into a spectrally mixed state. Therefore, the purity of the sources represents the separability of the JSA. This is found using the Schmidt decomposition of the JSA [211]:

$$F(\omega_1, \omega_2) = \sum_l \lambda_l \alpha_l(\omega_1) \beta_l(\omega_2)^*, \tag{3.9}$$

where $\{\alpha\}$ and $\{\beta\}$ are sets of orthonormal functions (although not necessarily equal or internally orthonormal). Normalisation imposes that $\sum_l \lambda_l^2 = 1$.

The Schmidt decomposition [52] is a way of quantifying the entanglement of a multipartite system. Beware the overlap in terminology here – a 'pure' source really means 'having a separable JSA', even though the state may not have any classical uncertainty associated with a mixed state, and we generally say 'impure' instead of 'mixed' in our work to avoid further confusion.

In order to simulate spectral error, we consider adding internal modes into our state. That is, the n spatial modes of the experiment are computationally relevant, and for each spatial mode, we consider m spectral modes, meaning that the total covariance matrix is size $2nm \times 2nm$. As we are most concerned with errors in the sources, we must decompose the state into its component linear optical transformations on these sources (as described in [132, 178, 207, 212]). These are all implemented in the `decompositions` folder of [210].

We first use the Williamson decomposition[8], $\sigma = \frac{1}{2} M \hat{\mathbb{1}} M^\dagger$. This holds for a pure state, which may be considered a symplectic transformation acting on the vacuum. For a mixed state, we instead consider the symplectic transformation on a thermal state, as considered further in Section 5.3.4.

Using the Bloch-Messiah decomposition, this symplectic matrix can be expressed in terms of the single-mode squeezers that generate the state (S) and the unitary performed by the interferometer:

$$M = USV^\dagger. \tag{3.10}$$

[8] This is another point where it is important to note the difference in notation between this thesis and the paper [1].

Now we can consider the sources and interferometer separately. We rewrite the unitaries as $\mathcal{U} = U \otimes \hat{\mathbb{1}}_m$ and $\mathcal{V}^\dagger = V^\dagger \otimes \hat{\mathbb{1}}_m$ to include the internal modes. We will also replace S with $\mathcal{S}$, so that $\sigma = \frac{1}{2}\mathcal{M}\mathcal{M}^\dagger$, where $\mathcal{M} = \mathcal{U}\mathcal{S}\mathcal{V}^\dagger$. The matrix $\mathcal{S}$ represents a set of squeezers which act on a single spatial mode, but multiple spectral modes. This process, which includes the decompositions, is implemented in the function `make_spectral_version(State state, Jsa jsa)`, in the file `circuits/optical_circuits.cpp` of [210].

The sources are characterised by finding a suitable JSA, as in Eq. 3.9. We consider frequencies to be discrete. Furthermore, in the simulation, these are simply treated as internal mode labels (and so this could equally be applied to other degrees of freedom). First it is necessary to choose a set of basis functions - for integrated optics, these are typically the Hermite polynomials [93, 213]. We then need to find the Schmidt coefficients λ_l.

We impose the purity, P:

$$\sum_i \lambda_i^4 = P := \mathrm{Tr}\left[\hat{\rho}^2\right]. \tag{3.11}$$

We construct the set of l non-zero Schmidt coefficients to satisfy the normalisation condition using:

$$\{\lambda_{l,b}\} = \{\lambda_1\} \cup \{\lambda_i = \sqrt{k_{l,b}(i)(1-\lambda_1^2)} | 2 \leq i \leq l\}, \tag{3.12}$$

using the values $k_{l,b}(i)$ with $\sum_i k_{l,b}(i) = 1$. One useful form for $k_{l,b}$ is the geometric scaling:

$$k_{l,b}(i) = \frac{b^{l-i}}{\sum_{j=1}^{l} b^{j-1}}. \tag{3.13}$$

This looks a bit confusing because of the factor b^{l-i} on the top instead of b^i, but this makes sure that if there is only 1 Schmidt coefficient it is set to 1.

Using these definitions the purity condition is then:

$$\sum_i \lambda_i^4 = \lambda_1^4 + (k_{l,b}(2))^2(1-\lambda_1^2)^2 + \cdots + (k_{l,b}(l))^2(1-\lambda_1^2)^2 = P, \tag{3.14}$$

which produces a quadratic equation in λ_1^2 which is straightforward to solve. It looks like this:

$$\begin{aligned} &\left(1 + \sum_{i=2}^{l} (k_{l,b}(i))^2\right)(\lambda_1^2)^2 \\ &- 2\left(\sum_{i=2}^{l} (k_{l,b}(i))^2\right)(\lambda_1^2) \\ &+ \left(\sum_{i=2}^{l} (k_{l,b}(i))^2\right) - P = 0\,, \end{aligned} \tag{3.15}$$

We note that throughout we take the positive roots to keep all λ_i positive. We can then substitute λ_1 back into Eq. 3.12 to solve for the remaining Schmidt coefficients.

Due to the choice of the two parameters l and b, there are many distributions of Schmidt coefficients that generate a source with a particular purity P. The construction described here does not generate all Schmidt distributions but it can be used to generate a large class of distributions. Using this method, a JSA with a particular purity can be constructed in [210] using the class constructor `Jsa(int nspec, Purity purity)`, where `nspec` is the number of spectral modes, in the file `math-code/symplectic.cpp`.

Using this construction, any purity with $\frac{1}{2} \leq P \leq 1$ only needs two Schmidt coefficients. For quantum information processing applications we are typically interested in sources with purities close to 1, and as such the majority of physically relevant or interesting sources can be described with two Schmidt modes [214]. Nonetheless, as this doubles the effective number of modes, you could be worried that this would cause an exponential increase in the simulation time due to the increased size of the covariance matrix (and the scaling of the complexity of the hafnian). Including spectral modes that are traced over in the detection calculations in fact only introduces a cubic overhead, using the simulation method introduced in [178], which we outline in the following section.

3.4.2 Sampling method

In this chapter, we consider threshold detectors, as these are used most widely experimentally. These 'click' when there are photons in the associated spatial mode, but do not resolve the spectral mode, or distinguish between different (non-zero) photon numbers. The method implemented in [210], used in this work, is described in [178, 207].

Let clicks be measured in the subset of spatial modes $\mathcal{C}$ (which we label 'on'), and vacuum in the spatial modes $\mathcal{V}$ (which we label 'off'). This is equal to:

$$\begin{aligned} p_{\text{on, off}}(\mathcal{C},\mathcal{V}) &= \mathrm{Tr}\left[\hat{\rho}\prod_{i\in\mathcal{C}}(1-|\text{vac}\rangle\langle\text{vac}|_i)\prod_{j\in\mathcal{V}}|\text{vac}\rangle\langle\text{vac}|_j\right] \\ &= \sum_{\mathcal{B}\in 2^{\mathcal{C}}}(-1)^{|\mathcal{B}|}p_{\text{off}}(\mathcal{B}\cup\mathcal{V}), \end{aligned} \tag{3.16}$$

where $\mathcal{B}$ sums over the power sets of $\mathcal{C}$ (see [178] for further details). Furthermore (as in Eq. 2.113):

$$p_{\text{off}}(\mathcal{B}) = \mathrm{Tr}(\rho\,|\text{vac}\rangle\langle\text{vac}|_{\mathcal{B}}) = \det\left(\hat{\mathbb{1}}/2+\sigma_{\mathcal{B}}\right)^{-1/2}. \tag{3.17}$$

As would be expected, Eq. 3.16 has an exponential number of terms. However, each term given by Eq. 3.17 is efficient to calculate. The number of terms in Eq. 3.16 scales exponentially in the number of spatial modes in which photons are detected, but independently of the number of spectral modes we consider, whereas increasing the number of spectral modes increases the size of the matrices in Eq. 3.17, which induces a cubic overhead (from the complexity of calculating determinants). This is in comparison to other calculation methods

(for threshold detection, this would require calculation of the Torontonian [173]), in which the complexity would scale with the size of the submatrix on which the function is applied, and therefore considering additional internal modes induces an exponential overhead. We also note that the size of the matrix in Eq. 3.17 depends on the number of modes but not on the number of detected photons, which is not true for the PNR case. This advantage is also shared by the related method for PNR statistics described in [178]. Therefore, although this is not an efficient simulation scheme – we would not expect it to be given the conjectured complexity of simulating GBS – we can use it to simulate additional spectral modes more efficiently than previous methods, without incurring a prohibitive exponential runtime cost. This sampling method is implemented in [210], in the `Quantum_sampling` class in the file `sampling/samplers.hpp`.

Finally, it is worth noting that this is an example of exact simulation - that is, the difficulty of these simulations scales exponentially with photon number, but we do not expect an increase in error. Therefore, although we may be simulating experiments with levels of error that make them classically simulable, this was not done using efficient (although higher-error) methods.

3.4.3 Choosing the correct scaling parameter

In the following simulations, we look for samples of a specific size (these are mostly samples of size k), and discard the others. This introduces an additional overhead in the complexity of the algorithms, and therefore we wish to choose the optimum scaling parameter c (as introduced in Section 3.3.4) to increase the proportion of overall samples that have the correct photon number.

By considering Eq. 3.6, we see that the coefficient c^k is the same for every k-vertex subgraph (as c is chosen for the initial state and is not changed for each subgraph). Therefore changing c does not change the relative probability (when considering PNR detectors) of different subgraphs of the same size, and only the relative probability of different numbers of vertices. Hence this only changes the results of Section 3.5.3.

For threshold detectors, we can use the marginal probabilities of Gaussian states – that is, the probability of a particular outcome when just considering a single mode, while tracing over the possible outcome patterns of all other modes – which are easy to calculate. Therefore, we can use that the expected number of 'clicks' (detectors registering at least one photon during a measurement) is given by

$$\langle \hat{C} \rangle = \sum_{i=1}^{m} 1 - p_i(\text{vac}), \tag{3.18}$$

where m is the number of modes.

We consider a state after loss L, so $\sigma \to L\hat{\mathbb{1}}/2 + (1-L)\sigma$. Using Eq. 3.17, the vacuum probability in mode i, $p_{\text{vac}}(i)$ is given by:

$$\begin{aligned} p_{\text{vac}}(i) &= \det\left(\hat{\mathbb{1}}/2 + \sigma_{\{i\}}\right)^{-1/2} \\ &= \det\left(L\mathbb{1} + (1-L)\left(\mathbb{1} + (X\otimes\mathbb{1})A\right)^{-1}_{\{i\}}\right)^{-1/2} \\ &= \det\left(L\mathbb{1} + (1-L)\begin{pmatrix} \mathbb{1} + c^2\mathcal{A}D^{-1}\mathcal{A}^* & c\mathcal{A}D^{-1} \\ cD^{-1}\mathcal{A}^* & D^{-1}\end{pmatrix}_{\{i\}}\right)^{-1/2}, \end{aligned} \tag{3.19}$$

where we have introduced $D = \mathbb{1} - c^2\mathcal{A}^*\mathcal{A}$. Recall that $A = c\mathcal{A}\oplus\mathcal{A}$ (or $c\mathcal{A}^*\oplus\mathcal{A}$ for complex weighted graphs), and that $\sigma_{\{i\}}$ indicates the submatrix of σ associated with mode i: $\sigma_{\{i\}} = \begin{pmatrix} \sigma_{ii} & \sigma_{i,i+m} \\ \sigma_{i+m,i} & \sigma_{i+m,i+m}\end{pmatrix}$.

As $c\mathcal{A} = B$ where B is the B matrix described in Section 2.8, $c\mathcal{A} = URU^T$ where U describes the interferometer and R is a diagonal matrix of squeezing values ($R = \text{diag}(t_i)$). Hence,

$$\mathcal{A}^{-1} = cU^*R^{-1}U^\dagger \tag{3.20}$$

and:

$$D = U^*\left(\text{diag}\left(\frac{1}{1-|t_i|^2}\right)\right)U^T, \tag{3.21}$$

hence:

(3.22)

$$\begin{aligned} p_{\text{vac}}(i) &= \det\left(L\mathbb{1} + (1-L)\begin{pmatrix} \mathbb{1} + U\text{diag}\left(\frac{|t_i|^2}{1-|t_i|^2}\right)U^T & U\text{diag}\left(\frac{t_i}{1-|t_i|^2}\right)U^T \\ U^*\text{diag}\left(\frac{t_i^*}{1-|t_i|^2}\right)U^\dagger & U^*\left(\text{diag}\left(\frac{1}{1-|t_i|^2}\right)\right)U^\dagger\end{pmatrix}_{\{i\}}\right)^{-1/2} \\ &= \det\begin{pmatrix} 1+(1-L)\sum_k U_{ik}^2\left(\frac{|t_i|^2}{1-|t_i|^2}\right) & L+(1-L)\sum_k U_{ik}^2\left(\frac{t_i}{1-|t_i|^2}\right) \\ L+(1-L)\sum_k U_{ik}^{*2}\left(\frac{t_i^*}{1-|t_i|^2}\right) & L+(1-L)\sum_k U_{ik}^{*2}\left(\frac{1}{1-|t_i|^2}\right)\end{pmatrix}^{-1/2} \\ &= \left(\left|\sum_j |U_{ij}|^2\left(\frac{1-L|t_i|^2}{1-|t_i|^2}\right)_j\right|^2 - (1-L)^2\left|\sum_j U_{ij}^2\left(\frac{|t_i|}{1-|t_i|^2}\right)_j\right|^2\right)^{-1/2}. \end{aligned}$$

Increasing c means increasing the squeezing parameter, and hence this increases the likelihood of collision events, which makes the GBS experiment more simulable [176]. The effect of squeezing on dense subgraph finding was investigated experimentally and theoretically in [180].

Further results regarding the probabilities of different photon numbers with different values of c are presented in [207]. Choosing the ideal scaling parameter (for threshold detectors) is implemented in [210] as `set_optimal_threshold_c`.

3.5 Results

We now present results on dense subgraph finding. In Sections 3.5.1 and 3.5.2, we use 3 different example graphs of $n = 24$, searching for subgraphs of size $k = 8$. The first was generated using the Erdős-Rényi method with $\rho = 0.4$, producing a graph with initial density 0.362. The densest subgraph has density[9] 0.714. The second graph was generated using the Erdős-Rènyi method with $\rho = 0.2$, producing a graph with initial density 0.196. The densest subgraph has density 0.536. The third is a modification of the first graph so that it has the same initial density, but contains a clique.

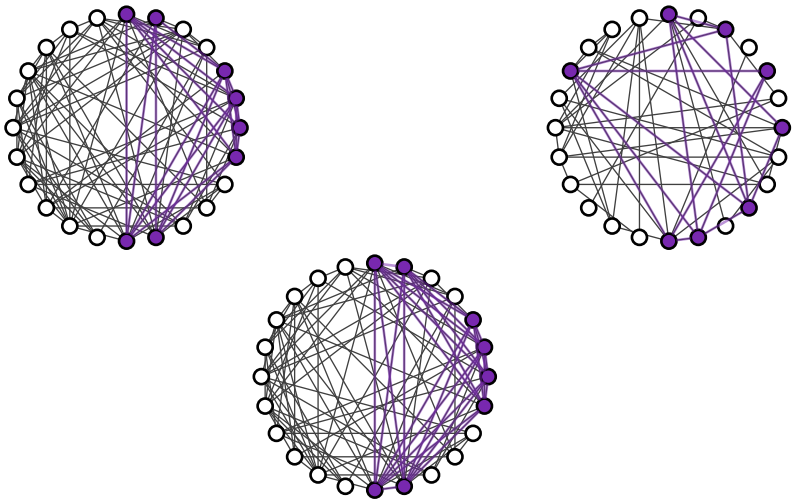


FIGURE 3.2. The 24-vertex example graphs used to generate data in this chapter. The top left is denser, the top right is more sparse, and the bottom contains a clique (a fully connected graph), but is otherwise the same as the top left. The densest subgraphs are highlighted in purple.

Examples of how to generate these data are given in the file `examples/graph_examples.cpp` of [210].

We note that, in comparison to [179], we require significantly fewer samples (by approximately an order of magnitude). This is likely to be due to the fact that we consider slightly smaller graphs.

3.5.1 Random sampling with error

We consider the results of the random sampling protocol presented in Alg. 1, with the initial graphs described above.

In Fig. 3.3, we compare the performance of the classical and quantum-enhanced algorithms. The horizontal axis shows how many subgraphs are drawn from the 8-vertex

[9] This was found using a brute force search, which is possible at the graph sizes considered here, but quickly becomes intractable for larger graphs.

subspace, and the vertical axis represents the density of the densest subgraph found. Each point is the mean outcome over 1000 repetitions of the algorithm.

In the case of the quantum-enhanced algorithms, we consider GBS in which the quantum states have suffered the application of a lossy channel, or where the sources are spectrally impure, using simulation methods described previously. The scaling parameter c was kept constant when adding error to directly compare the impact of noise.

We also compare the performance of the deterministic classical algorithm described in Section 3.3.2. Due to the randomness that can be present in this algorithm, there can be different outcomes - we ran this once, and when there were different options for selecting subgraphs, we chose the one that appears first when all subgraphs of the same size are ordered numerically.

As is the case in [179], the quantum-enhanced algorithms noticeably outperform the classical algorithm, as we would expect. The deterministic classical algorithm is able to find the clique, but fails to find the densest subgraph in either other case. Similarly to [179], the average results from the quantum-enhanced algorithm fall short of identifying the densest subgraph in the first and third case, but are more effective in the second case (where the initial subgraph has a lower density). However, the most interesting feature of these results is that the quantum-enhanced algorithm still performs very well, and is significantly more effective than the classical algorithm, with the presence of high levels of error.

3.5.2 Simulated annealing with error

We consider the results of the simulated annealing protocol presented in Alg. 2, with the same initial graphs as previously described. We kept the temperature schedule fixed (as described in Section 3.3.5), selecting the initial temperature as 0.01 (as in [179]), and used $l = \lfloor k/2 \rfloor$.

In Fig. 3.4, we compare the performance of the classical and quantum-enhanced algorithms. The horizontal axis shows the number of steps in the algorithm (i.e. the number of times that the `GBS-Tweak` subroutine is run), which is the variable a in Alg. 2. We note that this is not equivalent to the overall number of samples which must be drawn, and also that the size of the samples required varies depending on parameters of the algorithm, and can vary due to the random nature of the algorithms.

As in Section 3.5.1, the vertical axis represents the density of the densest subgraph found, each point is the mean outcome over 1000 repetitions of the algorithm (although this time considering Alg. 2), and we consider the impact of loss and spectrally impure sources on the GBS statistics used in the quantum-enhanced algorithms. We also use the same values of c for each plot.

These results show a similar pattern to Fig. 3.3, with the quantum-enhanced algorithms giving a much better result than the classical algorithm, but showing little difference with high amounts of error.

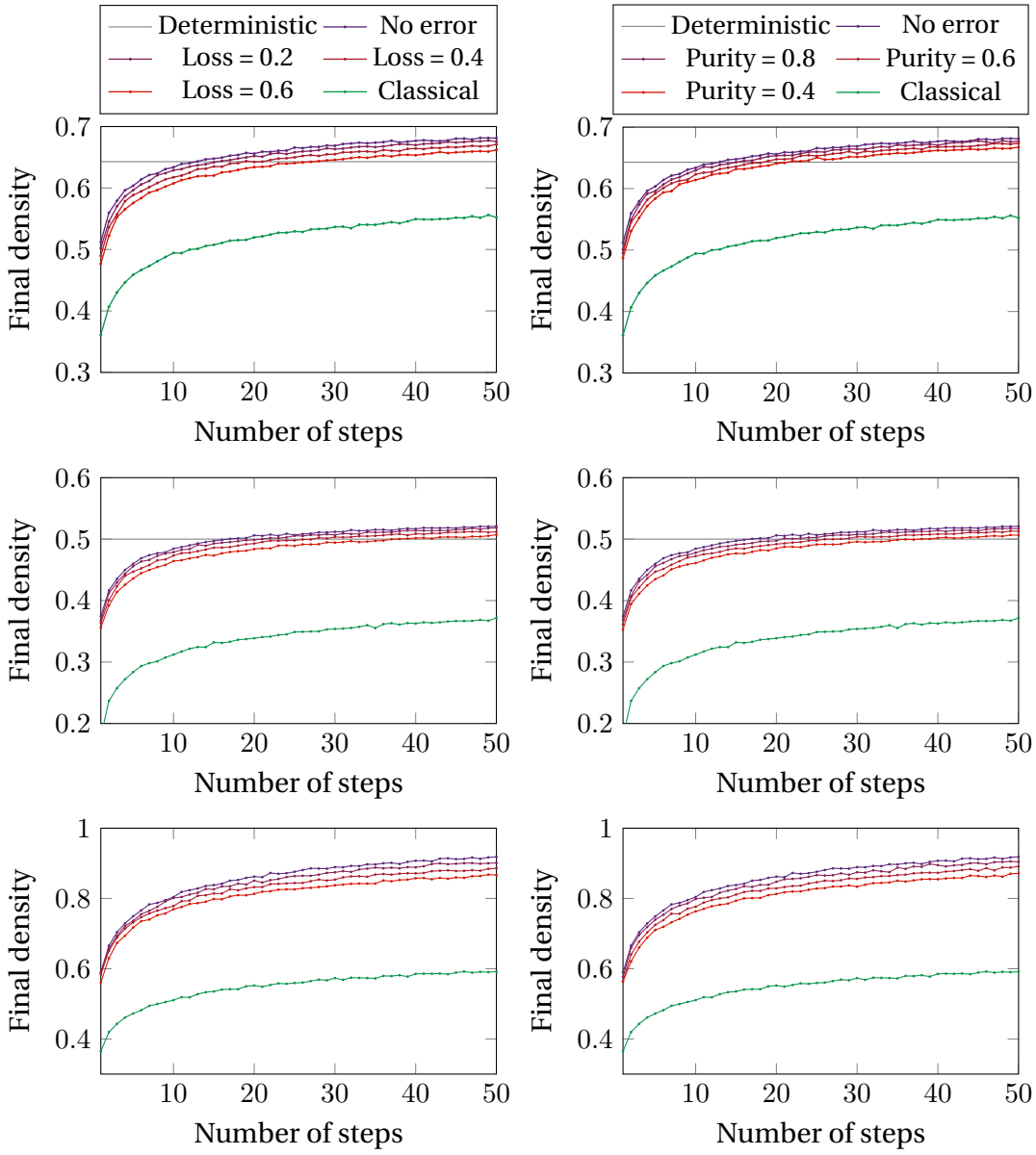


FIGURE 3.3. We show the effect of error on the simulated GBS-enhanced random sampling algorithm, in terms of the final density achieved (y axis) in the graph output after a certain number of steps of the algorithm (x axis). The left hand column shows the effect of increasing loss, and the right hand column shows the effects of using sources with lower spectral purity. The grey horizontal line shows the density of the graph found by the deterministic classical algorithm from [200], as a benchmark for comparison. From top to bottom, the rows represent different starting graphs: with initial density 0.362, initial density 0.196, and the first graph but modified to contain a clique.

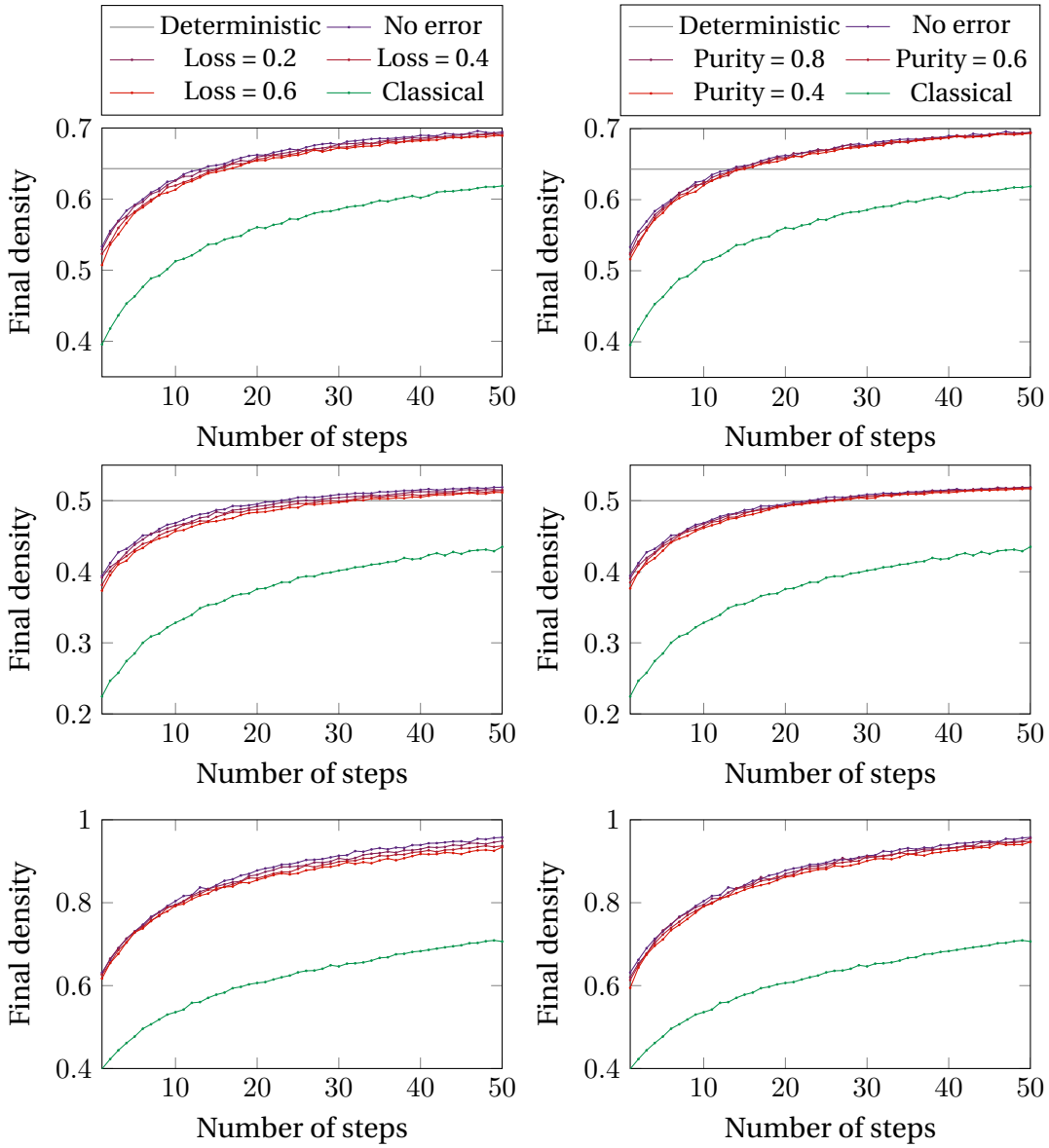


Figure 3.4: As previously, we consider the impact of error on a simulated GBS-enhanced DkS algorithm, but using simulated annealing. The left hand column shows the effect of increasing loss, and the right hand column shows the effects of using sources with lower spectral purity. The grey horizontal line shows the density of the graph found by the deterministic classical algorithm. From top to bottom, the rows represent different starting graphs: with initial density 0.362, initial density 0.196, and the first graph but modified to contain a clique.

3.5.3 Sampling without postselection

The results of the previous sections use results from GBS statistics where we have postselected to only use samples that were drawn from the correct (i.e. k-click) subspace. This is not necessarily an unrealistic reflection of how the algorithms would be used. Given a subgraph

with a number of vertices close to k, it is easy to make a small change by adding vertices or removing low-degree vertices. Choosing c well also reduces the probability of this happening.

Nonetheless, in Fig. 3.5 we show the results of the random sampling algorithm (Alg. 1), applied to the first graph (density 0.362), as presented in Fig. 3.3. The lines representing classical, uniform sampling are the same, but the quantum-enhanced plots are adjusted so that each step of the algorithm does not draw samples from the 8-vertex subspace, but instead we count every sample generated through the GBS simulation. If the number of clicks in the sample is not equal to k, this is discounted (we count this as drawing a subgraph with density 0).

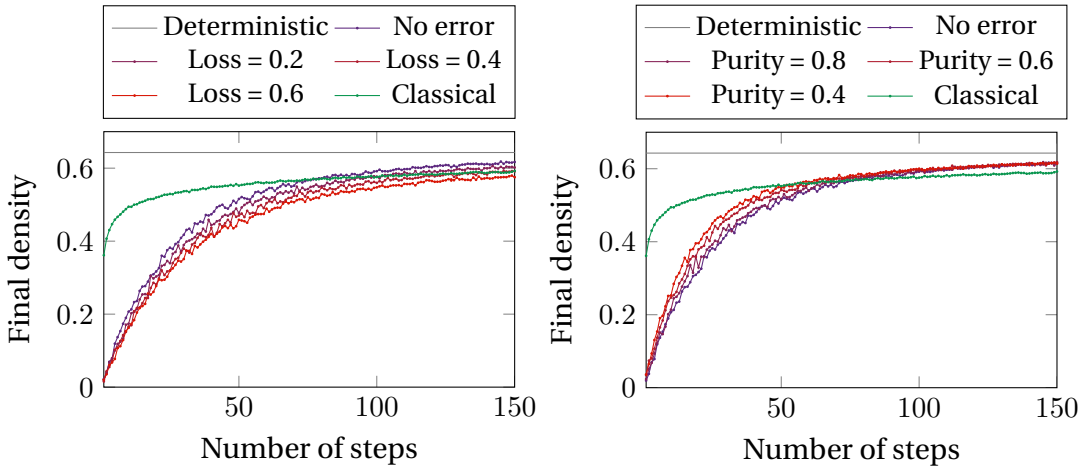


FIGURE 3.5. The effect of error on the GBS-enhanced random sampling algorithm, but adjusted so that the number of steps includes samples that fail to draw subgraphs in the 8-vertex subspace. The left hand side shows the effect of increasing loss, and the right hand side shows the effects of using sources with lower spectral purity. The grey horizontal line shows the density of the graph found by the deterministic classical algorithm. We use the initial graph with density 0.362.

We use a different c parameter for each value of loss, which optimises the expected photon number (to correspond to 8 clicks). By doing this, we assume that, in an experimental implementation of this protocol, the experiment would be characterised so that the expected loss value was known and the squeezing parameter could be adjusted accordingly.

In this case, the advantage to be gained from using the classical-enhanced algorithm is less significant, although we note that it seems to plateau at a higher value. Once again, the results are robust to the sources of error considered, even with high levels of loss, as we can compensate for this by choosing the optimal c value.

One feature in which these results differ from the postselected results comes from considering the impact of spectrally impure sources, in which the higher-error statistics seem to be more effective. Given that this pattern is not evident in the postselected case, and that every

line in this plot uses the same value of c, it suggests that the presence of spectral impurity increases the chances of drawing outcomes in the 8-click subspace.

It may be particularly relevant in this case that we use threshold detectors. Consider the 'bosonic birthday paradox', discussed in [84]. This is the statement that, as long as $m \gg n^2$, there is a low probability of collisions. We expect this to be the case for classical particles, but for identical bosons, the proof is more involved, as they are 'gregarious', or more likely to end up in the same state.

3.5.4 Average density of samples

The previous results focused on the effectiveness of the algorithm when considering individual examples with particular starting graphs. Although these results show an interesting pattern, it would now be useful to see whether this continues for a variety of different graphs, particularly at different sizes.

In Fig. 3.6, we consider the density of samples drawn from graphs of different sizes. For each initial size (shown on the horizontal axis), we use the Erdős-Rényi model to create a graph with $\rho = 0.4$ or $\rho = 0.2$. We then draw subgraphs of size k, where k is the nearest integer to $\sqrt{n}$ (chosen to align with use cases such as molecular docking), from the uniform distribution in the classical case, or by simulating GBS from the Gaussian state made from the graph, in the quantum-enhanced case. We choose c to be optimised to maximise the probability of drawing samples in the k-click subspace.

We draw 1000 samples and plot the average density over all samples. Note the difference to the random sampling algorithm - this is not the density of the best subgraph found. As before, the quantum-enhanced versions are considered in the presence of loss, or with spectrally impure sources.

We see, once again, the robustness to error of sampling to find dense subgraphs. It also seems that the improvement of GBS statistics for this task, over the uniform distribution, decreases as the graph size increases.

We also see that in these results, the GBS statistics with spectrally impure sources perform generally as well as, or even better than, without error. It may seem contradictory that these samples give subgraphs with high density on average, and yet they do not outperform the no-error results in Figs. 3.3 and 3.4. However, we note that the goal of the algorithms is to find the densest subgraph, and therefore may benefit from different qualities of the distribution. In particular, the most important aspects of DkS are the height of the peaks corresponding to the densest subgraphs, and so the probabilities of other graphs make less difference to these results, but will affect the overall average density.

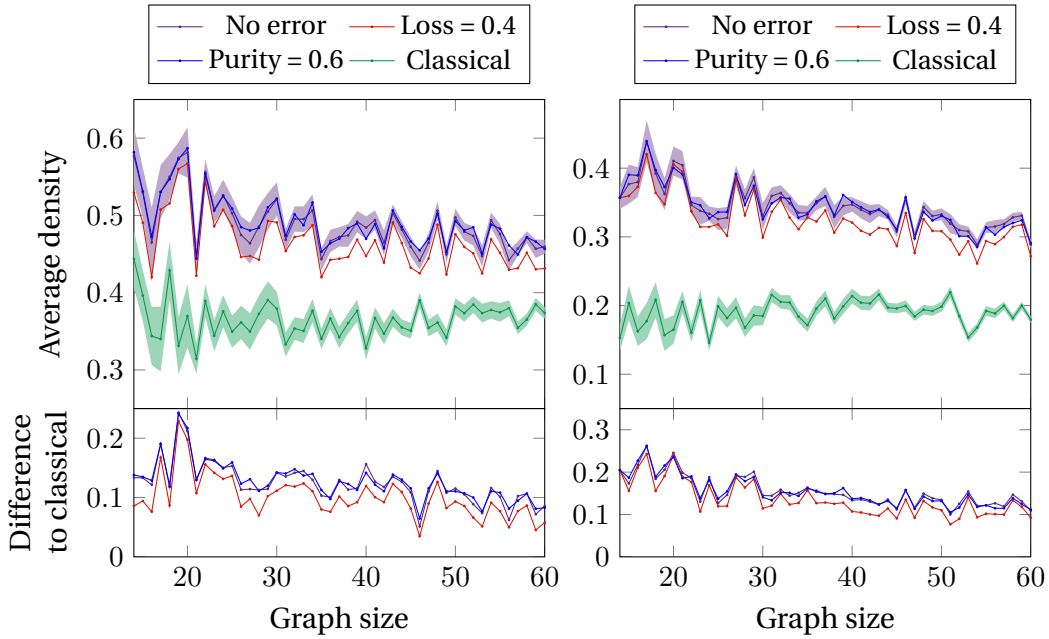


FIGURE 3.6. We plot the average density of size $\sqrt{n}$ subgraphs from 1000 samples, from initial graphs of different sizes (x-axis). The initial graphs are randomly generated, with density approximately 0.4 (left hand side) or 0.2 (right hand side), and samples are produced from simulated GBS (with error) or from the uniform distribution (the 'classical' distribution). Shaded regions show the standard deviation in sampled subgraph density for the no-error quantum-enhanced case, and the classical case. The difference between the quantum-enhanced and classical results is also shown for ease of comparison.

3.6 Discussion

The goal of this section is to investigate the impact of error on quantum-enhanced DkS algorithms, and the main feature of these results that is of interest is that the use of GBS for DkS seems to be robust to the forms of error considered, which is evidenced by little variation in the density of subgraphs output by simple algorithms, as a function of number of steps. Naïvely, this is a promising outcome for the use of near-term quantum devices. However, we further propose that these results could be used as a basis for efficient classical algorithms which would therefore bypass the need for quantum devices – which aligns with the results of [150].

As discussed in Section 2.9, high levels of error, including loss, make GBS experiments more easily classically simulable. The impact of spectrally impure sources is less well understood, although it decreases the quality of quantum interference [178], and therefore we would also expect these states to be easily simulable. It approaches the limit of simulating thermal states, which are also efficient to classically sample from [174].

Using [175], we can find the loss threshold beyond which it becomes efficient to classically simulate the experiment. As we increase c, this threshold increases, and as we chose a different value of c for each simulation with different loss levels, the threshold at which the experiment would become classically simulable changes. Following this analysis, with loss of at least 43.7%, the no-error GBS considered here is efficiently classically (approximately) simulable, which rises to a threshold of 47.3% for our simulation where loss was at 60%. As such, although the algorithms appear robust to errors, the level of errors we considered permit efficient classical algorithms to simulate the underlying sampling task with minimal error. The levels of loss considered in this work are within the region expected for current large-scale quantum advantage experiments (see, for example, the comparison table in [177]).

Insofar as our results can be generalised, they therefore suggest that, although GBS may be useful for the task of DkS, even if the experiment has a high level of error, this may challenge the claim of quantum advantage for this task. An efficient classical algorithm could be constructed which simulates sampling from the appropriate quantum state, with sufficiently high levels of error; the preceding results indicate that this would perform similarly to the GBS-based algorithms initially proposed.

Crucially, this does not contradict the central quantum advantage claim of GBS in general, nor the hardness of DkS. Efficient classical algorithms that can perform as well (or similarly) as GBS for this application cannot necessarily sample from the target distribution (i.e. when considering graph hafnians) as GBS. The algorithms considered in this chapter are stochastic, and therefore it is difficult to consider their complexity, however they are not exhaustive searches and do not guarantee to provide a solution to the DkS problem in a fixed amount of steps. Furthermore, in comparing the effect of different sources of error, we exactly simulated the high-error states, and did not run the efficient sampling schemes, which may not perform as well due to the further error they impart.

This work is based on a numerical study of particular examples, and hence is not a definitive proof against quantum advantage, although the code repository allows for further study and is available for anyone to use. Nonetheless, the results are consistent within the cases studied, and show a clear pattern of robustness to error, which we further investigate in the following chapter. We also note that this supports other research which was carried out concurrently into the use of GBS for the DkS problem; in [150], they show that, as the graphs have real, positive weights, it is possible to efficiently classically sample from the distribution described by haf(S) and therefore there is an efficient classical algorithm which can recreate the results of the quantum-enhanced DkS algorithms in most cases. Further work on nonnegative Hafnians show that efficient classical estimators are usually (although not guaranteed to be) effective [215].

Additionally, we can further analyse the impact of error on the quantum states, and measurement outcome distributions, to understand the source of the robustness. In Fig. 3.7, we

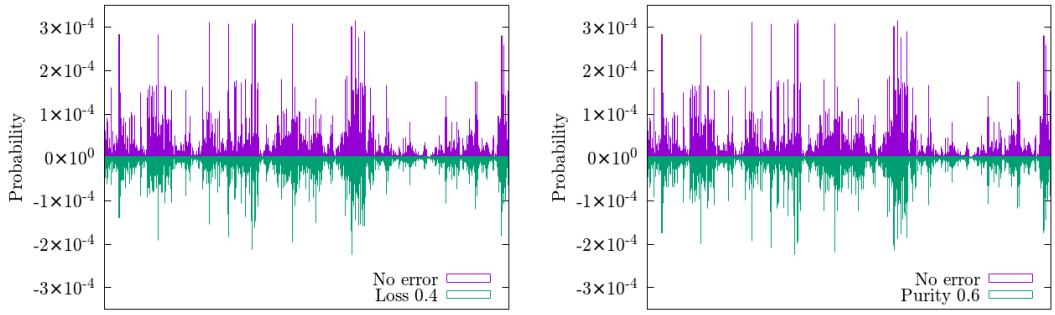


FIGURE 3.7. The probability (normalised to the 8-mode subspace) of choosing a particular 8-mode sample using threshold detectors on GBS. The x-axis is omitted, but is the ordered list of binary strings of Hamming weight 8. The same initial state is used for both, but the reflected probabilities (appearing as negative probabilities on the y-axis) correspond to the sample probabilities with different forms of error on the initial state.

show the probability distribution of different outcomes in GBS experiments (using threshold detectors), only considering the 8-click subspace, comparing the distributions of the experiment with and without error (for example, on the left-hand graph, a particular position on the x-axis corresponds to a measurement outcome, with the positive value representing the probability of that outcome in the no error simulation, and the negative weight representing the negative of the probability of that outcome in the simulation with 40% loss). It is clear that the peaks of the distributions are in the same places with and without error, and are still considerably higher than the rest of the distribution – hence, these experiments can continue to identify the dense subgraphs correctly within a reasonable number of samples. This gives an intuition as to the effectiveness of the algorithm with high levels of error, which we will investigate further in the following chapter.

It is also constructive to consider the Gaussian states used in the algorithm. We consider the example of the first 24-mode graph considered, with density 0.362. In Fig. 3.8, we plot the adjacency matrix (upper left) of our 24-mode example graph, and the covariance matrix (upper right) of the associated Gaussian state.

We then use the Williamson decomposition, followed by the Bloch-Messiah decomposition, to find the unitary matrix and matrix of single mode squeezers that produce the state, also shown in Fig. 3.8 (lower left and lower right, respectively).

In particular, it is interesting to note that the sources show one bright squeezer, and many low-power single-mode squeezers. We found this was also the case with the other example graphs considered. The similarity of these states to thermal states, and the impact of using threshold measurements on these states, and for the DkS problem in general, is considered

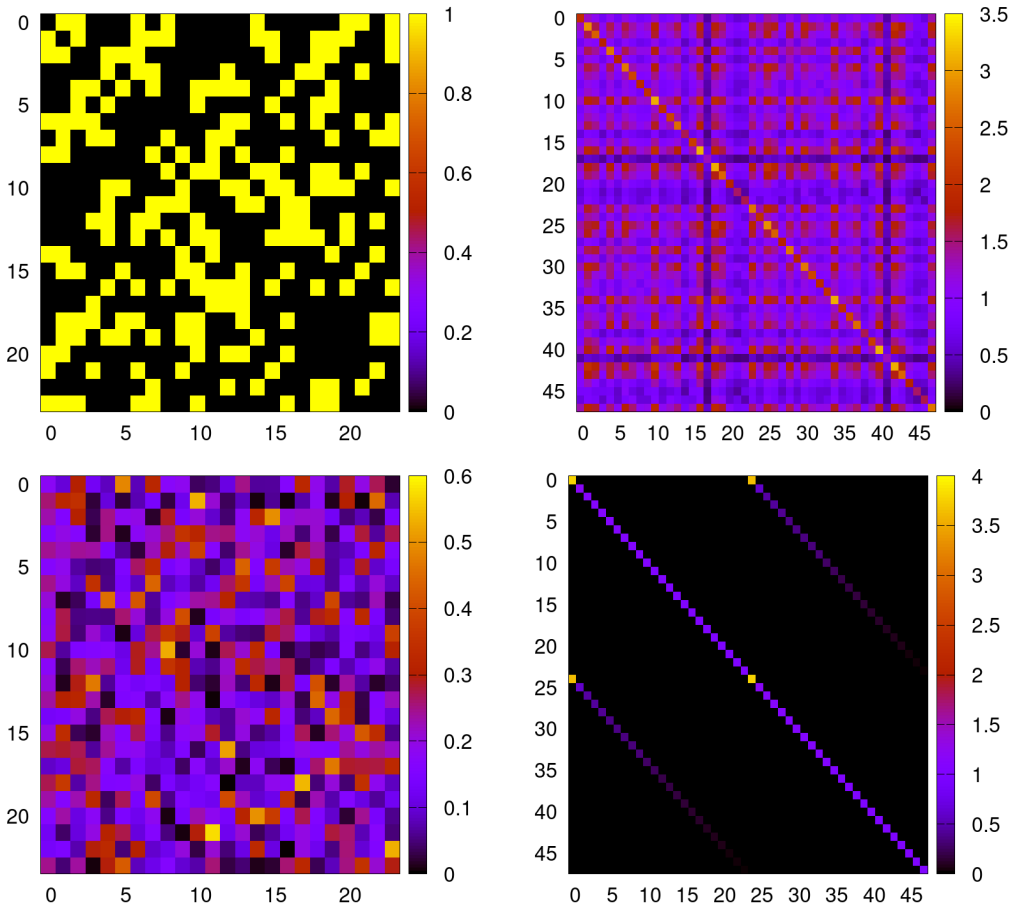


FIGURE 3.8. We plot the absolute values of the Gaussian transformation corresponding to our 24-vertex graph with a density of approximately 0.6. From left to right, top to bottom, we show: the adjacency matrix of the graph, the covariance matrix describing the associated Gaussian state, and the element-wise absolute values of the linear interferometers U to create the state and the single-mode squeezer matrix S (arising from the Bloch-Messiah decomposition, as in Eq. 3.10), which shows the brightness of the sources input to each mode.

further in [207]. This work motivates the use of a thermal state approximation, as this would capture the important features of the distribution – namely, that the maximal probabilities should be preserved.

Alongside the possibility of a quantum-inspired efficient classical algorithm based on simulable, high-error states, there is the possibility that there could be an experimental advantage using optics, but that there is not necessarily a quantum advantage (but perhaps an 'optical', or 'analogue' advantage instead). That is, interference effects by using states that may not be considered quantum, and are more easily implemented, could give a useful advantage.

3.7 Outlook and further work

There are several further directions that work on this problem could take. The program described in this chapter can be used to collect further numerical data. For example, it could be useful to look at graphs with a particular structure, beyond the Erdős-Rényi form considered here, such as those that may be more relevant to certain applications (for example, in internet networks [216]). Alternatively, we could consider weighted or directed graphs, or look specifically at clique finding, as this is particularly relevant to molecular docking [190]. We also have not considered the use of a slightly modified protocol that uses mixed Gaussian states, as described in [202].

If designing a protocol that can be done with 'classical' resources, to examine the possibility of an analogue or optical advantage, a similar study could be carried out using the appropriate states. As described in [175, 177], it is possible to use semidefinite programming to find the closest 'classical' (non-negative Wigner function) state to a given Gaussian state. We approach these states (in particular, thermal states) with increasing amounts of loss, so our work in this chapter suggests they may be a useful resource to consider for this purpose. Other possible states that could be of interest include the squashed states investigated in [176], or alternatively, encoding data in intensity values (such as in [217]). Furthermore, given that spectrally impure states seem particularly useful for this application, it may be of interest to the community to further understand the way that this impacts the complexity of simulation, and how to find classical states that are close in phase space, which is currently the focus of fewer studies than, e.g. distinguishability or loss.

As we have only seen numerical studies on a certain number of examples, it would be useful to be able to generalise these results further – in particular, to understand why GBS may demonstrate a quantum advantage in the original, proposed framework, which cannot be translated into this application. One explanation is that these protocols deal with Gaussian states constructed from real, non-negative matrices, which may be classically efficient to sample from. This is the foundation of the quantum-inspired classical algorithms in [150]. A further explanation could come from the structure of the problems considered here, and how that differs from the problem that is proposed for a classical algorithm attempting to simulate GBS. That is, the challenges of sampling from a distribution of hafnians differs to identifying the largest hafnian, and therefore sources of error (from experimental applications to classical simulation) may impact these differently. Finally, this problem uses the distribution of hafnians of the subgraphs as a proxy for the distribution of interest, which is the density of the subgraphs. Nonetheless, these distributions have different properties, so we may be able to access the distribution of densities in a way which does not have the same difficulty as sampling hafnians. Determining which explanation is most likely for the seeming lack of quantum advantage, and the possible consequences of this for other applications of GBS, will be the focus of the next chapter.

CHAPTER 4

Sampling the density of subgraphs

The sky above was blue at last,
The sky beneath me blue in blue.

Mary Elizabeth Coleridge, *L'Oiseau Bleu*

4.1 Preface

The work in this chapter was was conceived and carried out by N.R.S., under the supervision of Dr Dara P. S. McCutcheon and Prof Anthony Laing. This work was carried out using the computational facilities of the Advanced Computing Research Centre, University of Bristol (`http://www.bristol.ac.uk/acrc/`).

4.2 Introduction

The results of the previous chapter suggest that the advantage that can be gained from using GBS for the DkS problem is at most polynomial. This is confirmed by the results of [150], which show that the sampling task for which we require GBS can be performed by an efficient classical simulator (that is, sampling from the hafnians of non-negative matrices).

This classical algorithm is possible as we are only interested in the cases in which the A matrix has only real, positive elements, and so we do not expect there to be the difficult-to-simulate effects of quantum interference. Although there are no currently-known efficient classical algorithms for hafnian approximation of matrices with real positive elements, this classical solution is able to efficiently sample from the correct distribution, which is required to explain why GBS should not lead to an exponential quantum advantage.

Nonetheless, it is not clear that this is the only factor that influences whether this is a valuable application of GBS. We can also examine other aspects of the quantum-enhanced DkS algorithms. Firstly, we note that the task of sampling from a distribution and finding the maximum value of that distribution may not necessarily have equivalent difficulties. Secondly, when considering the density of graphs, it may be more effective to sample from a distribution where the different subgraphs are weighted by density and not perfect matchings. This difference is particularly relevant, as the density of a graph is efficient to calculate, whereas its hafnian is not.

In this chapter, we compare different aspects of the quantum-enhanced DkS algorithms with similar schemes, in order to better understand why the algorithm is effective, and why it may not give a quantum advantage. Firstly, in Section 4.4, we examine other sampling tasks – firstly the Max-Haf problem, and then DkS using complex-weighted graphs – in order to understand whether states formed from unweighted graphs show a higher resilience to error than other, randomly generated states, in similar sampling tasks. We then construct a distribution where the probabilities of outcomes are proportional to the relevant subgraph weights, and use verification tasks to show how effectively GBS with error can sample from this distribution, in Section 4.5. Finally, in Section 4.6 we give an example of a technique that can be used to sample from this distribution directly, which is not dependent on the graphs being positive-weighted. As in the previous chapter, we use numerical data from classical simulations, and not experimental data, although in some cases we can draw comparisons to existing experimental work.

We hope to give some insight into various reasons that an application of GBS may fall short of quantum advantage, with the intention that this analysis is not only useful when considering the effectiveness of GBS for this particular problem, but also when considering possible future applications.

4.3 Background

4.3.1 Density of weighted graphs

We will now be considering a more general version of the DkS problem when applied to weighted (although still undirected) graphs. For a graph with adjacency matrix $\mathcal{A} \in \mathbb{C}^{n\times n}$, an alternative definition of the density can be calculated as [162]:

$$\mathcal{W}(G) = \left| \sum_{i,j=1}^{n} \mathcal{A}_{i,j} \right|, \tag{4.1}$$

the sum of the weights of all edges[1]. We use the letter W to signify that this is the total weight of the graph. Note that in this case there is not a well-defined maximum possible density (unless there is a maximum possible edge weight), so it is not appropriate to scale this so that it is normalised, although some definitions consider the density to be $\mathcal{W}(G)/|V|$ [218]. This means that for graphs in which all edge weights are real and positive, adding vertices (and their associated edges) will increase the density – that is, larger subgraphs will be likely to have a higher density. Furthermore, it is possible for a graph with many edges to have a low density, if the positive and negative contributions from different edges are very close.

For real, nonnegative weighted graphs, many of the results reviewed in Section 3.3.1 regarding dense subgraph finding are still applicable, although these break down when considering negative-weighted edges. Although this problem is less common, it does appear as a subroutine in other graph problems. The problem is **NP**-hard even without restricting to a particular subgraph size k [219]. It is also shown in [219] that the greedy algorithm previously introduced (with modifications to consider edge weights) achieves an approximation value of $\mathcal{W}^*/2 - \Delta_-/2$, in which $\mathcal{W}^*$ is the highest subgraph density, and Δ_- is the absolute value of the largest negative degree (summation of negative weights for a single vertex).

Further variations include the heavy and dense subgraph problem [220], which considers the case that both edges and vertices have associated weights. Although we do not consider this variation in this chapter, we note the importance of vertex weights in modelling molecular docking, as well as its relationship to displaced GBS, which is covered in the following chapter [190].

4.3.2 The Max-Haf problem

A further problem that we will consider is the Max-Haf problem introduced in [203]. The problem is (informally) as follows: given a fixed input matrix A, find the submatrix A_S of dimension k that maximises $|\text{haf}(A_S)|^2$. We note firstly that we place two further restrictions on the problem so that it is suitable for using GBS: firstly, we consider submatrices constructed by selecting the same rows and columns of A, and we assume that k is even. We will also be particularly interested in matrices that are valid A matrices of Gaussian states (i.e. positive semi-definite).

This is an **NP**-hard problem by reduction to maximum clique, however it can be approximated using GBS by using appropriate adjustments to Algs. 1 and 2 (although we will focus in this chapter on the first method, random sampling). It is the canonical optimisation problem for GBS and we therefore expect it to provide a more faithful representation of the speedup that can be provided.

[1] In this chapter, we will assume that there are no loops in the graph, although this possibility is included in this definition.

4.3.3 Verification of sampling problems

Verification is an important, and difficult, part of the theoretical analysis surrounding quantum advantage experiments. This is the process of ensuring the correct functioning of the quantum computer, often using comparison with expected outcomes [221, 222]. I highly recommend Bill Fefferman's QIP tutorial.

Consider the case of integer factorisation, a problem which is expected to benefit from an exponential quantum speedup due to Shor's algorithm. Once a solution has been provided, it can be efficiently verified by a classical computer (by checking that the product of the provided factors is equal to the input integer). Another example, which is verifiable, although likely to still be out of reach of near-term quantum computers, is the Yamakawa-Zhandry protocol [223] (this is useful to know in case it turns out that factorising integers is in P). On the other hand, in the case of sampling experiments, it is more difficult to ascertain whether claims of quantum advantage are valid – see [224] for an informal discussion of the importance of these schemes.

In general, verification is done by comparison to the ground truth, which is the ideal distribution that the device aims to sample from (i.e. the distribution that would be sampled by an accurately implemented experiment with no error). Naïvely, the goal is to show that the TVD between the ground truth and experimental distribution is sufficiently small. However, we don't have direct access to the experimental distribution, and calculating ground truth probabilities is, by the nature of quantum advantage experiments, exponentially difficult. Therefore, several methods are generally employed when aiming to demonstrate or refute quantum advantage.

Firstly, smaller subsystems can be verified, as the classical task of simulating these, although inefficient, is not intractable. For example, in the case of (G)BS, this involves verifying experiments with a low photon number. This is useful in characterising errors in the device, and it is clear that an experiment must be able to complete these simpler tasks in order to be able to scale up successfully. However, it does not provide a reliable benchmark for demonstrating that an experiment has achieved its large-scale goal, as some of the more interesting quantum effects may originate from many-photon interference terms (although the importance of these terms in GBS is unclear, which is the basis of [225], and discussed informally in [226]). Similarly, some subspace of the possible transformations can be tested, and particular interferometer settings have signature outputs that can be identified [227].

Secondly, other measurements such as correlation functions are an efficient method to provide evidence of genuine quantum interference effects [228], or considering the output photon number (particularly using threshold detectors) [161]. There are also verification schemes that require some (although smaller) quantum resources [229].

All of these methods are not sufficient to efficiently guarantee that a GBS experiment has been carried out that samples from the correct (or a sufficiently accurate) distribution,

without some assumptions about the device, or additional (non-classical) resources, and this remains an ongoing subject of research.

Thirdly, there are several methods by which the difference between the experimental distribution and the ground truth can be measured. These methods generally take another, classically simulable model, the 'adversary' (e.g. squashed states) and compare to see whether this or the experiment (the 'trial' distribution) is a better approximation to the ground truth. The following techniques all use a similar principle, in that they take a certain number of samples from both the experiment and the simulation, and compare the likelihood of drawing these samples from the ground truth distribution. The intuition is that a better sampler will draw samples that are more likely according to the ground truth.

- Bayesian test [230]: we consider N samples drawn from the trial distribution. For each sample, we compare the ground truth probability of that outcome, $p(x)$, with the probability according to the adversary distribution, $q(x)$, and calculate:

$$r_B = \prod_{i=1}^{N} \frac{p(x_i)}{q(x_i)}. \tag{4.2}$$

  The numerator is the conditional probability of these samples occurring if drawing from the ground truth, and the denominator is the conditional probability of these samples occurring if drawing from the adversary distribution. So we can use Bayes' theorem:

$$\begin{aligned} r_B &= \frac{\Pr(\text{samples}|\text{ground truth})}{\Pr(\text{samples}|\text{adversary})} \\ &= \frac{\Pr(\text{ground truth}|\text{samples})\ \Pr(\text{samples})\ \Pr(\text{adversary})}{\Pr(\text{adversary}|\text{samples})\ \Pr(\text{samples})\ \Pr(\text{ground truth})} \\ &= \frac{\Pr(\text{ground truth}|\text{samples})\ \Pr(\text{adversary})}{\Pr(\text{adversary}|\text{samples})\ \Pr(\text{ground truth})}. \end{aligned} \tag{4.3}$$

  We're assuming there's equal probability of the underlying distribution being either the ground truth or the adversary, as what we're trying to do is measure our confidence that the trial samples came from either one. The ratio r_B then, by Bayes' theorem [231], corresponds to the degree of confidence we have in the samples coming from the ground truth distribution, over the adversary distribution (that is, the probability that this hypothesis is true given the data).

  We quantify this in the following way: the independent probability, $r_B/(r_B+1)$ is normalised and can be assigned independently to the ground truth probability (or the adversary in the reverse case). Then, different adversaries can be compared.

- Heavy output generation (HOG) [232], which proceeds in the following way:

  1. Find the median probability in the ground truth distribution.

2. Find the probabilities $p(x)$ of N trial samples.
3. The HOG test passes if $\geq \frac{2}{3}N$ trials have a probability greater than the median probability.

We note that, unlike the above test, this is not in comparison to an adversary distribution. This relies on the intuition that an efficient classical algorithm is not effective at determining whether a certain outcome is 'heavy' (probable with regards to the ground truth) – evidence for this (with a focus on IQP) is given in [232].

The hardness of spoofing the HOG test is based on the quantum threshold assumption (QUATH). The advantage of QUATH (the full statement is omitted here) is that it's based on the possibility of classical algorithms to estimate amplitudes, and decide whether they are larger than the median, which is separate from sampling, or relation problems. However, this is better studied for IQP than for BS, and there is currently a lack of strong complexity-theoretic evidence that it is true.

- Chen-HOG (CHOG) [161]: this is another adversarial test, where instead of comparing the probabilities from the ground truth and an adversary distribution, we instead draw N samples from both the trial and the adversary. We then calculate the ratio:

$$\begin{aligned} r_{\text{CHOG}} &= \frac{\Pr(\text{samples}_{\text{trial}})}{\Pr(\text{samples}_{\text{trial}}) + \Pr(\text{samples}_{\text{adv}})} \\ &= \left(1 + \prod_i \frac{\Pr(\text{sample}_{\text{adv}}(i))}{\Pr(\text{sample}_{\text{trial}}(i))}\right)^{-1}, \end{aligned} \tag{4.4}$$

  where all probabilities are calculated with reference to the ground truth distribution. We expect this ratio to approach 1 if the trial is a better approximation than the adversary, for similar reasons to the effectiveness of HOG. It seems similar to the Bayesian test– in particular, it has a similar form to the confidence ratio resulting from the Bayesian test, but has the additional advantage that it requires sampling from the adversary distribution, but not calculating individual probabilities (of the adversary distribution – it still requires calculation of ground truth probabilities), which may be more challenging[2]. As described in the Supplementary Material of [161], the intuition for CHOG comes from rewarding distributions that imitate the constructive quantum interference of the ground truth.

- Cross entropy benchmarking (XEB) [225, 233]:

$$H_{\text{XEB}} = -\sum_{j=1}^{N} p_{\text{trial}}(x_j) \ln(p_{\text{ideal}}(x_j)). \tag{4.5}$$

[2]This will be particularly relevant when we use densities – which are efficient to calculate – as the ground truth.

This was developed due to useful properties when assessing the quality of random circuit sampling. The distribution of RCS approaches the Porter-Thomas distribution for sufficiently high circuit depths, and XEB is a useful measure of the fidelity to this distribution. Variations such as linear XEB also exist, and can be implemented without an adversary. However, recent results show it can be 'spoofed' for (G)BS [234], which makes use of well-chosen distributions that correlate with the ideal distribution but are easy to compute.

As these involve calculation of ground truth probabilities, they are in general inefficient – but in GBS, if many of the samples produced have a low photon number, they correspond to events for which the probability is easier to calculate, and hence the quantities described are easier to compute. In this chapter, we focus primarily on HOG and CHOG, due to the ease of implementing and interpreting their results, and to compare an adversarial and non-adversarial verification method.

Finally, we note an interesting link to graph theory in this area [235]. This method identifies feature vectors of graphs that can be produced by coarse-graining the outcomes of GBS, and hence do not rely on a prohibitively large number of samples, to compare expected and actual outcomes.

4.3.4 Rejection sampling

We consider a random variable distributed according to $p(x)$ (the 'target distribution'), which may be a discrete (in which case $p(x)$ is the probability mass function) or continuous variable (so $p(x)$ is a probability density function). Rejection sampling [236–238] is a technique used to sample from $p(x)$, without having to calculate the whole distribution, but assuming there exists the capability to calculate the probabilities of individual outcomes (a helpful pedagogical introduction is given in [81]). Generally this is the case when there are so many possible outcomes that it is not feasible to calculate the cumulative distribution function, which is the basis of simpler techniques.

Rejection sampling proceeds as follows:

1. Choose a 'proposal distribution', $q(x)$, e.g. the uniform distribution, from which it is easy to sample.

2. Choose H such that $p(x) \leq Hq(x)$ for all x.

3. Sample from $q(x)$.

4. Keep the sample with probability $\frac{p(x)}{Hq(x)}$, otherwise reject the sample and return to step 3.

H must be chosen so that the probability in step 4, $p(x)/Hq(x) \leq 1$. The overall probability that a sample is accepted is:

$$\begin{aligned} \Pr(\text{accept}) &= \sum_y \Pr(\text{accept}|y)\Pr(y) \\ &= \sum_y \frac{p(y)}{Hq(y)} q(y) \\ &= \frac{1}{H}. \end{aligned} \tag{4.6}$$

Therefore, the efficiency of the algorithm is $1/H$; H samples must be drawn for every accepted sample.

The overall probability of outcome x in the accepted samples is:

$$\begin{aligned} \Pr(x|\text{accept}) &= \frac{\Pr(\text{accept}|x)\Pr(x)}{\Pr(\text{accept})} \\ &= H\frac{p(x)}{Hq(x)} q(x) \\ &= p(x). \end{aligned} \tag{4.7}$$

Hence, samples appear in the output with the correct probability.

For more intuition, Fig. 3.2 in [81] is very helpful. We can understand it like this: the total number of samples we wish to end up with is N, but the expected number of samples we will draw is HN. The expected number of x that are in this total set is $HNq(x)$, but we would like there to be $Np(x)$ in the final set, so we keep $p(x)/Hq(x)$ of these samples.

A natural choice for $q(x)$ is the uniform distribution over x, although this may be far from optimal. If certain properties of the target distribution are known, a proposal distribution can be chosen, for example a 'stepped' distribution where H is varied for different regions, or chosen to match the peaks of the target distribution.

There are two cases to consider which may cause difficulties in implementing this. Firstly, it may not be possible to find a minimal value for $p(x)$. This may mean that H is poorly chosen such that there exists some y with $p(y)/Hq(y) > 1$. In this case, at step 4, the sample is accepted with certainty. The result of this alteration is that the outcomes for which $p(x)$ is very large will be under-represented in the final distribution. The relative outcome probability (see [81] for further details) is instead $Hq(x)$, so the relative error of this probability becomes $\frac{p(x)-Hq(x)}{p(x)}$.

Secondly, finding a minimal value of $p(x)$ may make H prohibitively large, meaning that the algorithm is very inefficient. In this case, it is possible to carry out adaptive rejection sampling, which adjusts the sampling process to iteratively improve the proposal distribution (or, more precisely, the 'envelope distribution', $Hq(x)$). When applying rejection sampling, each sample is independent, and therefore the accepted samples still have the correct probabilities if the distribution is adapted between samples. As each sample requires calculating

a probability $p(x)$, this can then be used to replace the probability $Hq(x)$ in the envelope distribution (or, equivalently, $p(x)/H$ replaces $q(x)$ in the proposal distribution, although note that this requires renormalising $q(x)$). However, in practice, this is not likely to have a significant effect on the accuracy of rejection sampling until many samples have been drawn, as many probabilities must be replaced to improve the accuracy.

How would this actually work in practice? First, we would draw a sample using the rejection sampling algorithm described above. However, say we get outcome y, we then replace the proposal distribution with:

$$q(x) = \begin{cases} p(y), & x = y \\ \frac{1-Hp(y)}{|X|-1} & x \neq y \end{cases} \tag{4.8}$$

where $|X|$ indicates the number of possible outcomes of x. We can sample from this distribution using the inverse transform method. We can then continue the algorithm as before.

4.3.5 Previous results

GBS results collected using the Jiŭzhāng experiment not only showed a practical realisation of weighted DkS using GBS, but this work also considered the Max-Haf problem, and includes theoretical evidence that supports the use of threshold detectors [162]. They find that the advantage of using GBS increases with the photon number for Max-Haf, but not DkS (with both experimental and simulated numerical evidence). They also consider numerical studies of the effect of photon loss, which shows worse robustness for loss from Max-Haf than DkS, using complex valued matrices. They also consider the impact of thermal noise. These results also focus on the amount of steps required to reach the target value, instead of the average density of samples, which is potentially a more relevant metric for the optimisation problem. It is interesting to see that they only see a very slight score advantage (i.e. the density of subgraphs found) for dense subgraph finding of complex-weighted graphs, when applying the random sampling algorithm. A more significant advantage can be achieved using simulated annealing, or when instead benchmarking the algorithm by the number of steps required to find the highest density subgraph.

As mentioned in the previous chapter, an experiment for dense DkS using a time-bin encoded interferometer was carried out in [180]. They also investigated the impact of loss on DkS, with higher squeezing to compensate, again finding high robustness to loss. Finally, we note the work of [150], which also includes a high level of robustness to loss, above their proposed quantum-inspired classical algorithm.

4.4 Other sampling tasks

4.4.1 Max Haf

We consider the Max Haf problem, as described in Section 4.3.2. In Fig. 4.2, we show the performance of the classical and quantum-enhanced random sampling algorithm (Alg. 1, appropriately revised to maximise the absolute value squared of the hafnian of the submatrix) either under the impact of loss, or with spectrally impure sources.

The matrices we use as input to this algorithm are the adjacency matrix of the dense 24-vertex example graph from the previous chapter (Fig. 3.2), or a random complex-valued matrix, constructed as UU^T, where U is a Haar-random unitary (the absolute values of the elements of this adjacency matrix are shown in Fig. 4.1). We consider submatrices of size 8×8. The maximum hafnian that can be attained by the subgraphs is 900 (which is given by a different subgraph to that with the highest density), and the maximum hafnian that can be attained by the complex-valued submatrices is 5.623×10^{-5}.

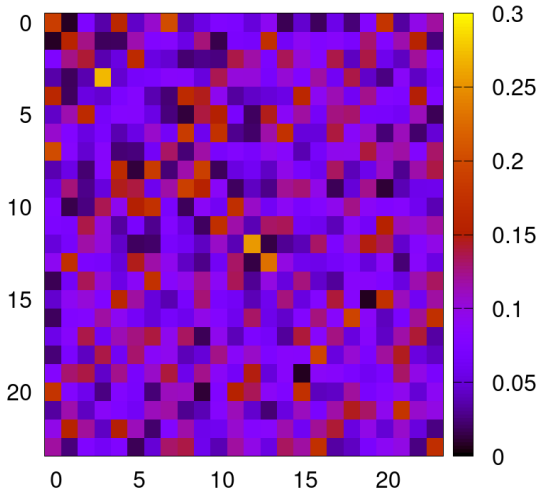


FIGURE 4.1. The absolute values of the adjacency matrix of the example complex-weighted 24-vertex graph, which represents the magnitude of the edge weights.

Firstly, we note that the results in Fig. 4.2 show that unlike in the case of DkS, the algorithm is not likely to be successful in finding the optimal submatrix. We also note that there is less separation between the quantum and classical performance in the complex-weighted case. These factors make it harder to draw conclusions about the robustness of the algorithm to error, as with many more samples the differences in performance could become more clear, as well as if this differs in the binary-valued and complex-valued cases (as we could expect, if the possible efficient classical simulation of GBS for the DkS problem arises from there being real, positive-valued matrices). Nonetheless, it is particularly interesting to note the

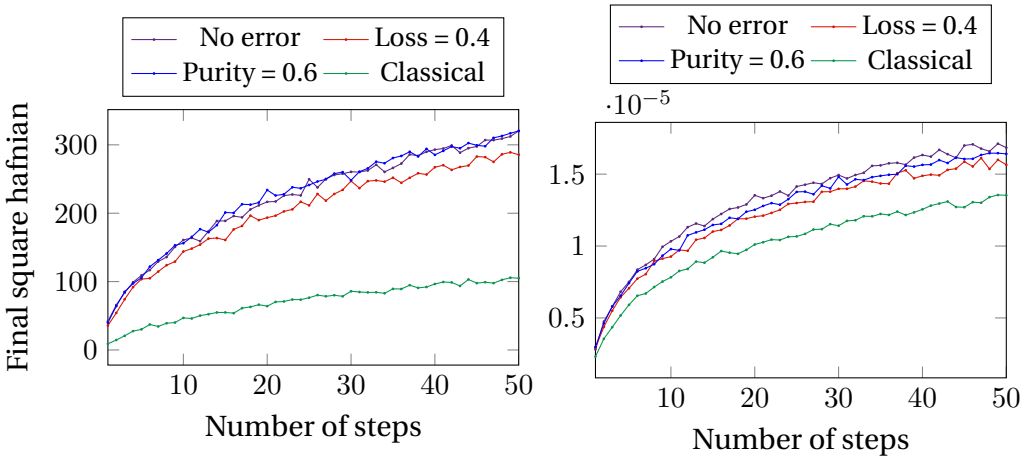


FIGURE 4.2. The performance of the random sampling algorithm for the Max Haf problem, with different sources of error, shown by the size of $|\text{Haf}|^2$ of the output graph (y-axis) as a function of the number of steps (x-axis). The left hand side uses an initial matrix with binary entries (from the adjancency matrix of an unweighted graph), whereas the right hand side uses an initial matrix with complex-weighted entries.

performance of the algorithm on unweighted graphs seems to be unaffected (at least in this case) by spectrally impure sources.

In Fig. 4.3, we once again consider a range of different initial matrix sizes (n). We use either the n-vertex random graphs considered in Section 3.5.4, or matrices of size $n \times n$ equal to UU^T, where U is a Haar-random matrix. This reflects the matrices sampled by a GBS device, although in the framework where there is equal squeezing in each mode (in this case the squeezing is set by the scaling parameter).

We draw samples of size k, where k is the nearest integer to $\sqrt{n}$, either from the uniform distribution (in the classical case) or using a simulation of GBS (with or without error). The scaling parameter is set to optimise the number of clicks to be k, as described in Section 3.4.3. We consider submatrices of size $k \times k$, but the hafnian is only defined for square matrices of even dimension. Therefore, we only consider n ranging from 31 to 42. We plot the average value of the absolute value of the hafnian squared - that is, for submatrix $A_{\mathbf{n},\mathbf{n}}$, we consider $|\text{haf}(A_{\mathbf{n},\mathbf{n}})|^2$. The standard deviation is omitted from these plots – in general, this is a similar size to the mean values. (For that reason, adding it makes the plots harder to read.)

In general, these do not seem to show the same robustness to loss as our previous results regarding dense subgraph finding, however it still remains robust to spectral impurity. This suggests that the possibility of efficient classical simulation may be due to using binary-valued matrices, but may also be due to the differences between sampling perfect matchings and sampling densities.

Furthermore, to run any classical algorithm that requires calculating hafnians (including

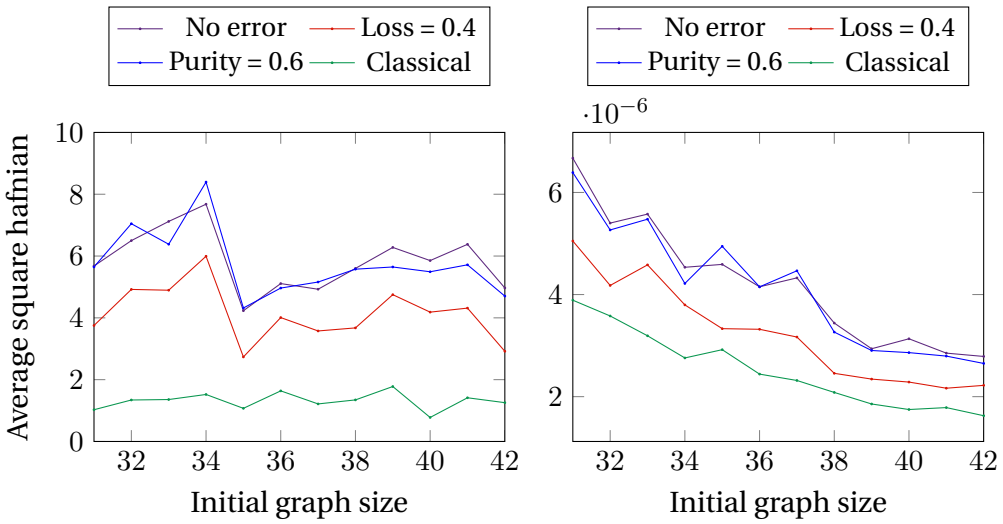


FIGURE 4.3. The average absolute square hafnian of samples of size $\sqrt{n}$ taken from initial matrices of size n, when sampled using simulated GBS (with error) or from the uniform distribution. The matrices are the adjacency matrices of randomly generated Erdős-Rényi graphs (left-hand side), or UU^T where U is a Haar-random unitary (right-hand side).

the random sampling algorithms shown above) is exponentially difficult. Therefore, even if an efficient classical sampler existed for an effective distribution, we would not be able to implement the algorithms as shown here.

4.4.2 Complex-weighted DkS

We now consider dense subgraph finding, but instead using the definition of density for complex-valued matrices, as in Eq. 4.1. We consider the randomly generated complex-valued matrix used in the previous chapter, and draw submatrices of size 8×8. We use the random sampling algorithm (Alg. 1) and consider the impact of loss and spectrally impure sources. The maximum density attained by a submatrix is 2.066.

We also show the average density obtained by samples, drawn by simulated GBS or from the uniform distribution, for the same initial graphs (formed by the matrices UU^T) as in the previous section (although for a larger range of sizes, taking k to be the nearest integer to $\sqrt{n}$). Once again, in the classical case, we sample from the uniform distribution.

These results are presented in Fig. 4.4. For the random sampling algorithm, there does not seem to be any advantage due to the quantum sampling results. This can be understood considering the results of [162], where the advantage of the quantum-enhanced algorithm is very slight, and only becomes clear with more samples, although it is able to find the densest subgraph sooner (which may be apparent if taking many more samples). When considering the average density of samples from initial graphs of different sizes, there does seem to

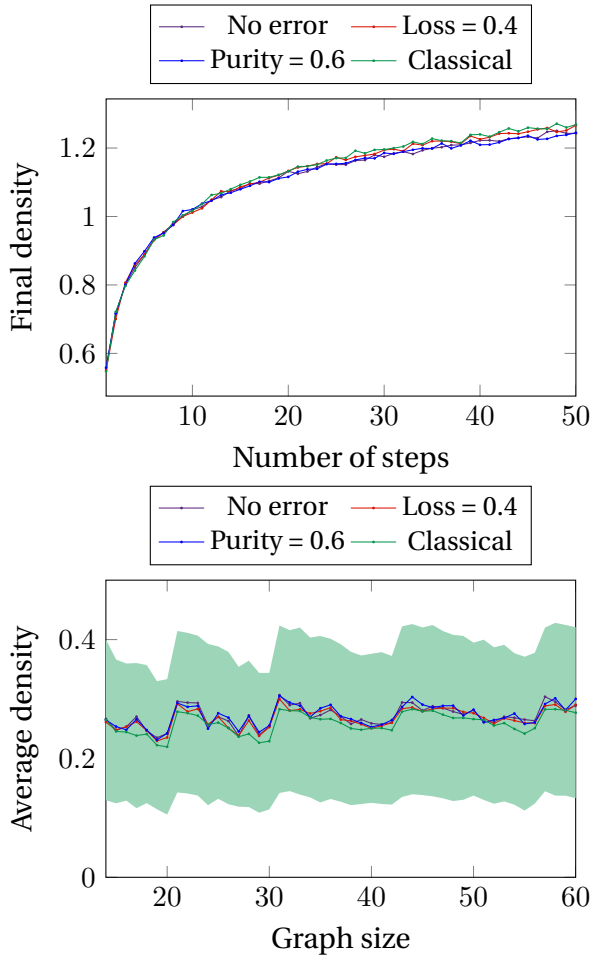


FIGURE 4.4. The performance of simulated GBS for DkS with complex-weighted graphs. We show the final density when using the random sampling algorithm (top), and the average density when drawing 1000 samples (bottom), with the standard deviation of classical samples shown on the latter (green region).

be some small difference from taking samples from the GBS distribution or the uniform distribution, however error does not seem to make any difference. Finally, it is interesting to note that there is little variation in the average density sampled, which is likely due to the properties of the randomly generated UU^T matrices.

In Fig. 4.5, we show the complex densities, and $|\text{haf}(A)|^2$ for the 8×8 submatrices A of the complex-weighted example graph. It is clear that there is significantly less variation in the relative sizes of the densities, which could indicate why the performance is more limited in this case. This could also explain what seems to be a better performance from GBS in DkS, as opposed to Max Haf, as there are many different subgraphs with densities close to the densest subgraph. Furthermore, we note that the study connecting hafnians and densities in [179] only focused on unweighted graphs, and more work may be necessary

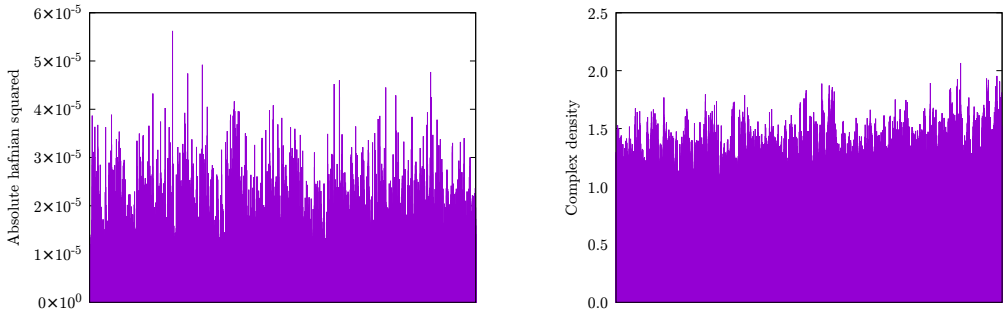


FIGURE 4.5. We consider the 8×8 submatrices of the initial 24×24 complex-weighted matrix, and show the square of the absolute value of the hafnian (left hand side) or the complex density (right hand side). The x-axis is omitted, but is the ordered list of binary strings of Hamming weight 8.

to confirm this relationship for complex weighted graphs. The Supplementary Materials of [162] consider this problem with numerical studies, revealing a modest, but statistically significant, correlation between the Torontonian and the density.

4.5 Density distribution

Sampling from hafnians, in order to solve the dense subgraph finding problem, is partly motivated by the correlations between graph hafnians and densities (at least in the unweighted case). In previous studies, we have compared to the classical case, which samples from the uniform distribution. However, it would be useful to consider whether there are other distributions – particularly those which are efficient to classically sample from – which can be used as a basis of stochastic algorithms, instead of the hafnian distribution that GBS draws from.

4.5.1 Normalisation

We wish to construct a distribution over all subgraphs of size k of an (unweighted) initial graph, G. Each point represents a different subgraph S, with $p(S) \propto d(S)$, that is, the probability of S is proportional to the density of that subgraph. We will refer to this as the 'density distribution'.

The normalisation factor of the density distribution is the sum of the densities of all possible subgraphs of size k. Equivalently, for the number of edges $|E_S|$, $p(S) \propto |E_S|$, and the normalisation factor is the sum of edges in all of the subgraphs:

$$\mathcal{N} = \sum_{S:|V_S|=k} |E_S|. \tag{4.9}$$

This is equivalent to a sum over all of the edges, multiplied by the number of times it is included in a subgraph, as each edge is included in the same number of subgraphs.

We note that each edge is included in this factor every time that a subgraph includes both of its terminating vertices. There are $\binom{n-2}{k-2}$ of these subgraphs, and hence the total normalisation factor is $|E_G|\binom{n-2}{k-2}$. Therefore:

$$\begin{aligned} p(S) &= \frac{|E_S|(k-2)!\,(n-k)!}{|E_G|(n-2)!} \\ &= \frac{k!\,(n-k)!}{2|E_G|(n-2)!} d(S). \end{aligned} \tag{4.10}$$

It is possible to see that this generalises to weighted graphs if all edge weights are real and positive, as $|E_S|$ can be replaced by the sum of edge weights (although in this case we do not include the normalisation factor $k(k-1)/2$ which comes from the unweighted density definition in Eq. 2.37). This is not the case if graph weights are negative or complex, as $\sum_{S:|V_S|=k} \mathcal{W}(S)$ is not necessarily equal to $c\mathcal{W}(G)$ for some constant c.

4.5.2 Verification with density as the ground truth

Verification schemes are useful to give an insight into the similarity of different distributions when drawing samples. We find that this is particularly valuable when validating quantum advantage claims, as calculating probabilities or full distributions can be prohibitively difficult. Furthermore, for quantum sampling schemes, the difficulty of spoofing these tests is often underpinned by complexity arguments that limit the possible performance of a classical sampling algorithm.

In this section, we consider verification tasks where the ground truth is the density distribution. In particular, we focus on the HOG and CHOG tests as defined in Section 4.3.3, and we use randomly generated graphs of size n (unweighted, or with adjacency matrix UU^T in the weighted case), as used in other sections, considering subgraphs of size k, where k is the closest integer to $\sqrt{n}$. When considering the density distribution, we do not have the same complexity-theoretic evidence that these tests are hard to spoof, however they have useful intuition regarding the dense subgraph finding problem, as they consider the relative probabilities of samples in the ground truth, which we aim to maximise. They also do not require normalisation, as they compare relative sizes.

In Fig. 4.6, we show the performance of simulated quantum sampling with different sources of error, and classical sampling from the uniform distribution, at the HOG test, when the ground truth is the density distribution. We draw 1000 samples, and plot the number of samples that correspond to subgraphs above the median density of subgraphs of that size. We show results up to a maximum size of 56 due to the computational difficulty of calculating the median. The grey line shows 666 samples – points above this line correspond to passing

the HOG test. We do not necessarily expect any of the sampling schemes to pass or fail this test, but instead focus on the comparison between results.

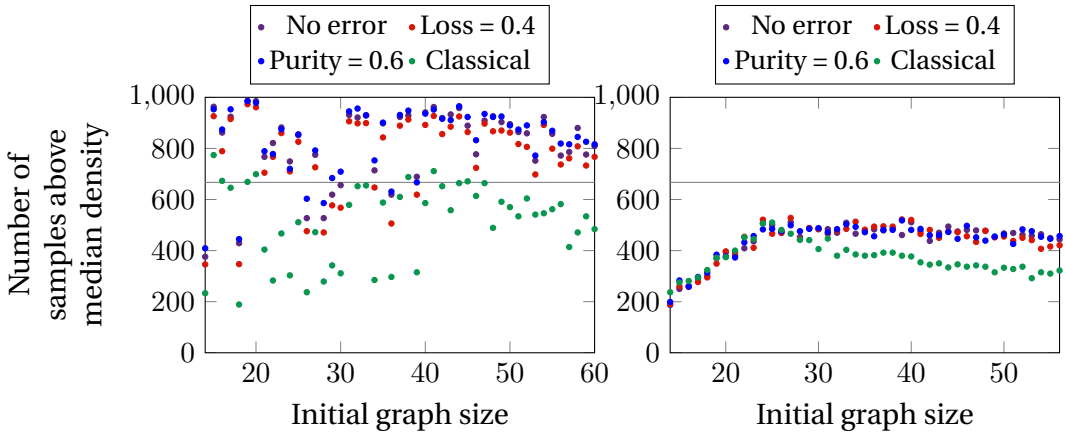


FIGURE 4.6. Results of the HOG test: the number of samples, out of 1000, that are above the median density, for unweighted subgraphs (left hand side) or complex weighted (right hand side), for a range of initial graph sizes. The grey line shows a 'pass', which is more than 2/3 of samples drawn above median density. Samples are drawn using simulated GBS or from the uniform distribution ('classical').

The HOG test appears to show that there is a noticeable improvement in results from using the quantum samples, in the unweighted case and, unlike the previous results we have presented, for the complex-weighted case. Overall, the results are better for the unweighted case, although both cases show robustness to error, which challenges the case for quantum advantage in the complex case, going beyond previous understanding. In the unweighted case, it is particularly interesting to notice that the performance does degrade slightly under the impact of loss, but not using spectrally impure sources. These results also suggest that there is limited advantage to be gained for using GBS for the complex-weighted case.

We now consider the CHOG test. This is an adversarial test that compares the performance of a trial distribution against an adversary, hence we compare the simulated quantum sampling with no error, to the distributions with error, and to classical sampling from the uniform distribution. Once again, we consider the same set of randomly generated graphs of size 14 to 60.

The CHOG ratio, defined in Eq. 4.4, should tend towards 1 if the trial distribution (the quantum distribution with no error) is favoured, and towards 0 if the adversary distribution is favoured. As we are using ratios, we once again do not require normalisation. In Fig. 4.7, we first plot how the ratio changes as more samples are drawn, from the example 24-vertex graphs. In Fig. 4.8, we plot how many samples are required before the ratio converges to within 1×10^{-9} of either 0 or 1, subtracted from 1000. A positive value indicates that the ratio

converges to 1 (favouring the no error quantum distribution) and a negative value indicates that the ratio converges to 0 (favouring the adversary). A value of 0 indicates the ratio does not converge. A more positive (negative) value means that the ratio converges to 1 (0) faster.

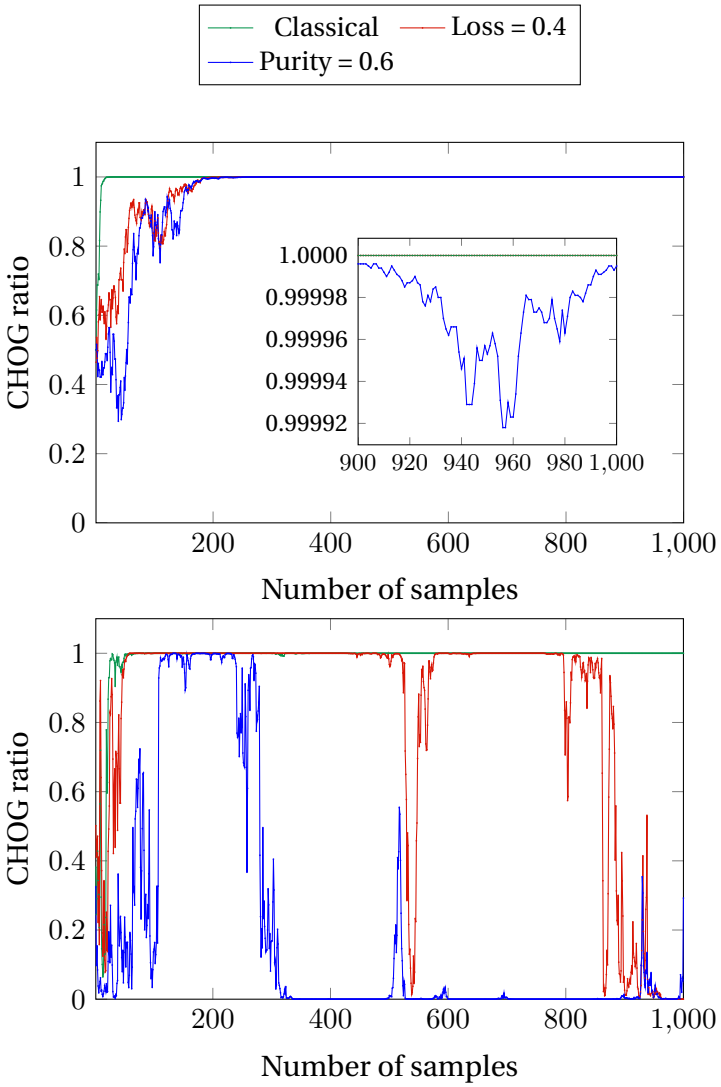


FIGURE 4.7. Examples of the CHOG test on 24-vertex graphs. We show how the CHOG ratio (Eq. 4.4) changes with the number of samples, comparing the no-error GBS simulation to statistics with error, or from the uniform distribution. The top (bottom) plot samples from the unweighted (complex-weighted) graph. The inset shows the ratio for the unweighted graph as the number of samples approaches 1,000.

Similarly to the HOG test, these results show that, in the case of unweighted graphs, the quantum sampling with no error outperforms sampling from both the uniform distribution and using GBS with loss. On the other hand, using spectrally impure sources does not seem to have a significant impact. Furthermore, there seems to be little evidence from the CHOG

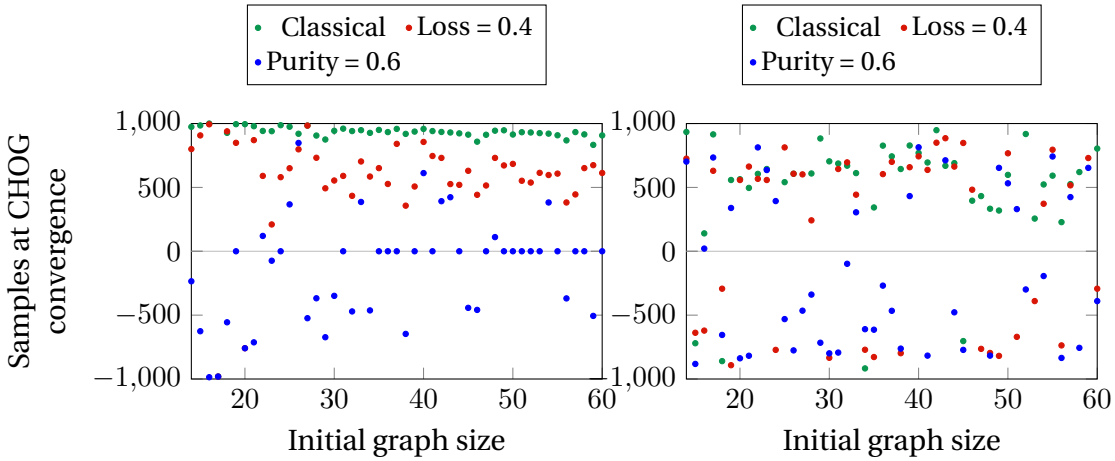


FIGURE 4.8. Results of the CHOG test (how many samples before the ratio converges to +1 (positive) or 0 (negative number of samples), on randomly generated graphs of different sizes, drawing samples of size $\sqrt{n}$. The left hand side are unweighted graphs and the right hand side are complex-weighted graphs.

test supporting the use of GBS for dense subgraph finding in the complex-weighted case.

4.6 Rejection sampling

4.6.1 Complexity of rejection sampling

First, let us consider sampling from the density distribution of unweighted graphs. For this analysis, we assume that the proposal distribution is the uniform distribution over subgraphs S of size k, so $q(S) = \frac{k!(n-k)!}{n!}$. However, there are other proposal distributions that can be easily constructed. For example, the distribution in which the probability of selecting a subgraph is proportional to the sum of the degrees (Δ) of its vertices, for which the normalisation factor is $\binom{n-1}{k-1}\sum_{i\in V}\Delta_i$.

The overall probability of acceptance is given by $\frac{1}{H}$, so for an efficient algorithm, H should be at most poly(n) (calculating densities is efficient, so that means that the rejection sampling scheme is efficient overall). We require:

$$\begin{aligned} H &\geq \frac{p(S)}{q(S)} \\ &= \frac{|E_S|n(n-1)}{|E_G|k(k-1)} \\ &= \frac{d(S)}{d(G)}. \end{aligned} \tag{4.11}$$

The minimum value of $|E_G|$ is $\frac{n}{2}$ (assuming that every vertex has degree at least 1), and the maximum value of $|E_S|$ is $\frac{k(k-1)}{2}$, so $H \geq n-1$. This holds for any graph, hence, we can

conclude that it is efficient to sample from the density distribution using rejection sampling in the case of unweighted graphs.

Nonetheless, this is the expected efficiency, over many samples. If there are many subgraphs with very low density (i.e. the density distribution is very concentrated around certain outcomes) then many samples may be rejected before accepting samples. This also generalises to real, positive-weighted graphs, although this will depend on the definition of density which is used, as well as the variation of weights that are permitted.

There is an important caveat to this, if applying it to dense subgraph finding. We could use this sampling as the search subroutine to Alg. 1, however, as the underlying process relies on first sampling from the uniform distribution and then calculating the density, there is no reason that it should give an efficiency advantage over a completely random search. Therefore, in this case, we suggest that the ability to efficiently sample from the underlying distribution does not imply a greater-than-polynomial speedup to dense subgraph finding. Despite this, much like GBS, it could be useful as a subroutine to more complicated algorithms.

Now we consider complex-weighted graphs, although for simplicity (w.l.o.g.) rescaled to have unit maximum absolute size of each edge weight. We consider the same $q(S)$ as before, but now with $p(S)$ weighted by the complex density from Eq. 4.1:

$$p(S) = \frac{1}{\mathcal{N}}\mathcal{W}(S), \tag{4.12}$$

where there is now the difficulty of calculating $\mathcal{N}$.

Once again, we require:

$$\begin{aligned} H &\geq \frac{p(S)}{q(S)} \\ &= \frac{\mathcal{W}(S)}{\mathcal{N}} \frac{n!}{k!\,(n-k)!} \end{aligned} \tag{4.13}$$

By the Cauchy-Schwarz inequality, the maximum value for $\mathcal{W}(S)$ is $k(k-1)/2$, although $\mathcal{N}$ may be very small. Let us assume that there is at least one subgraph with $\mathcal{W}(S) \geq 1$, which is the case if there exists one subgraph containing a single edge of weight 1. For example, we could modify the graph to ensure this by adding an additional vertex, connected by a single edge of weight 1 to another vertex. Then $\mathcal{N} \geq 1$, in which case we require:

$$H \geq \frac{k \cdot (k-1) \cdot n!}{2(k!\,)(n-k)!}. \tag{4.14}$$

We should also consider how to apply rejection sampling in the case that the normalisation coefficient is not known. To do this, we can consider applying the rejection sampling algorithm described in Section 4.3.4, but we now choose to accept with probability $\mathcal{W}(S)/Hq(S)$, choosing H such that $\mathcal{W}(S)/Hq(S) \leq 1$.

Now:

$$\begin{aligned}\Pr(\text{accept}) &= \sum_S \Pr(\text{accept}|S)\Pr(S) \\ &= \sum_S \frac{\mathcal{W}(S)}{Hq(S)} q(S) \\ &= \frac{\mathcal{N}}{H}.\end{aligned} \tag{4.15}$$

Then:

$$\begin{aligned}\Pr(S|\text{accept}) &= \frac{\Pr(\text{accept}|S)\Pr(S)}{\Pr(\text{accept})} \\ &= \frac{H}{\mathcal{N}} \frac{\mathcal{W}(S)}{Hq(S)} q(S) \\ &= \frac{\mathcal{W}(S)}{\mathcal{N}},\end{aligned} \tag{4.16}$$

as required.

Therefore, our classical algorithm can sample any weighted graph, even those with complex weights, although in this case, we expect the algorithm to be inefficient in the worst case.

4.6.2 Rejection sampling and anti-concentration

Let us consider the requirements on $\mathcal{N}$ that must be satisfied in order for rejection sampling to be efficient for complex weighted graphs.

We require that:

$$H \geq \frac{\mathcal{W}(S)}{q(S)}, \tag{4.17}$$

so we choose $H = \frac{k(k-1)n!}{2(k!)(n-k)!}$. We would like $H/\mathcal{N} \leq \text{poly}(n)$, as this corresponds to the number of samples for each accepted sample (the inverse of the efficiency).

Assume that some proportion β of subgraphs have weight $\mathcal{W}(S) \geq \tilde{\mathcal{W}}$. Then:

$$\mathcal{N} \geq \beta \frac{n!}{k!\,(n-k)!} \tilde{\mathcal{W}}. \tag{4.18}$$

We now have:

$$H/\mathcal{N} \leq \frac{k(k-1)}{2\beta\tilde{\mathcal{W}}}. \tag{4.19}$$

Thus, we require the condition:

$$\Pr\left(\mathcal{W}(S) \geq \frac{1}{\beta \text{poly}(n)}\right) \geq \beta. \tag{4.20}$$

We cannot provide evidence for such a condition applying to complex-weighted graphs. However, we can compare this to the anti-concentration conjecture of boson sampling (with outcome x) [84]: (page 65 on the ArXiv version)

$$\Pr\left(p(x) \geq \frac{n!}{\text{poly}(n, 1/\beta)}\right) \geq 1 - \beta \tag{4.21}$$

which was initially studied for permanents, but has also been conjectured for hafnians, as we will discuss further in the following chapter. We can choose, for example, that β is constant, so that these have a more similar appearance.

The anti-concentration condition in Eq. 4.21 applies for $p(x) = |\text{per}(A)|^2$, where A are the Gaussian matrices corresponding to the outcomes of BS, and it is an important aspect (although unproven) of the BS complexity proof. Given the correspondence between graph densities and permanents [209], this could suggest an interesting connection: the conditions required for efficient rejection sampling from the density distribution seem to coincide with the conditions required for (G)BS *not* to permit efficient classical simulation.

It should be noted that the condition in Eq. 4.20 is not genuinely an anticoncentration condition, due to the reliance on the weights but not probabilities (that is, the lack of normalisation). Therefore, a uniform distribution with very low weights would be considered anti-concentrated, but would not satisfy Eq. 4.20. On the other hand, very concentrated distributions would fail both conditions. Intuitively, we can see why this wouldn't be good for rejection sampling - you end up drawing many samples with very low acceptance probabilities before you find one you can accept.

Further work is needed to see to what extent this holds for dense subgraph finding, particularly in cases of interest (either in their relation to GBS, or uses of DkS).

4.6.3 Results

We now implement the rejection sampling schemes described previously on the randomly generated graphs of sizes 14 to 60, drawing samples of size $k = \sqrt{n}$, as described in previous sections of this chapter.

As we have shown, exact rejection sampling produces samples according to the correct distribution, and therefore typical verification tasks are not necessary in this case. In general, they also do not have the same complexity-theoretic justification, as it is not known the difficulty of spoofing these tasks for certain distributions. However, these are useful benchmarks for comparing this scheme to other sampling schemes we have discussed so far.

In Fig. 4.9, we show the results of the HOG test applied to these samples. This provides evidence that the sampling scheme is functioning as intended; it has an intuitive meaning in the case of sampling dense subgraphs, corresponding to the likelihood of drawing subgraphs above the median weight; and it gives further illustration of the usefulness of the HOG test for

the density distribution, and therefore whether it is relevant as a verification test for quantum sampling schemes.

In Fig. 4.9, the left-hand column corresponds to the unweighted graphs, and the right-hand column corresponds to the complex-weighted graphs (this only shows graphs up to size 56 due to the difficulty of calculating the median at larger sizes). The number of subgraphs sampled with density above the median density is shown, when 1000, 10 000, and 100 000 samples are drawn (top to bottom, respectively).

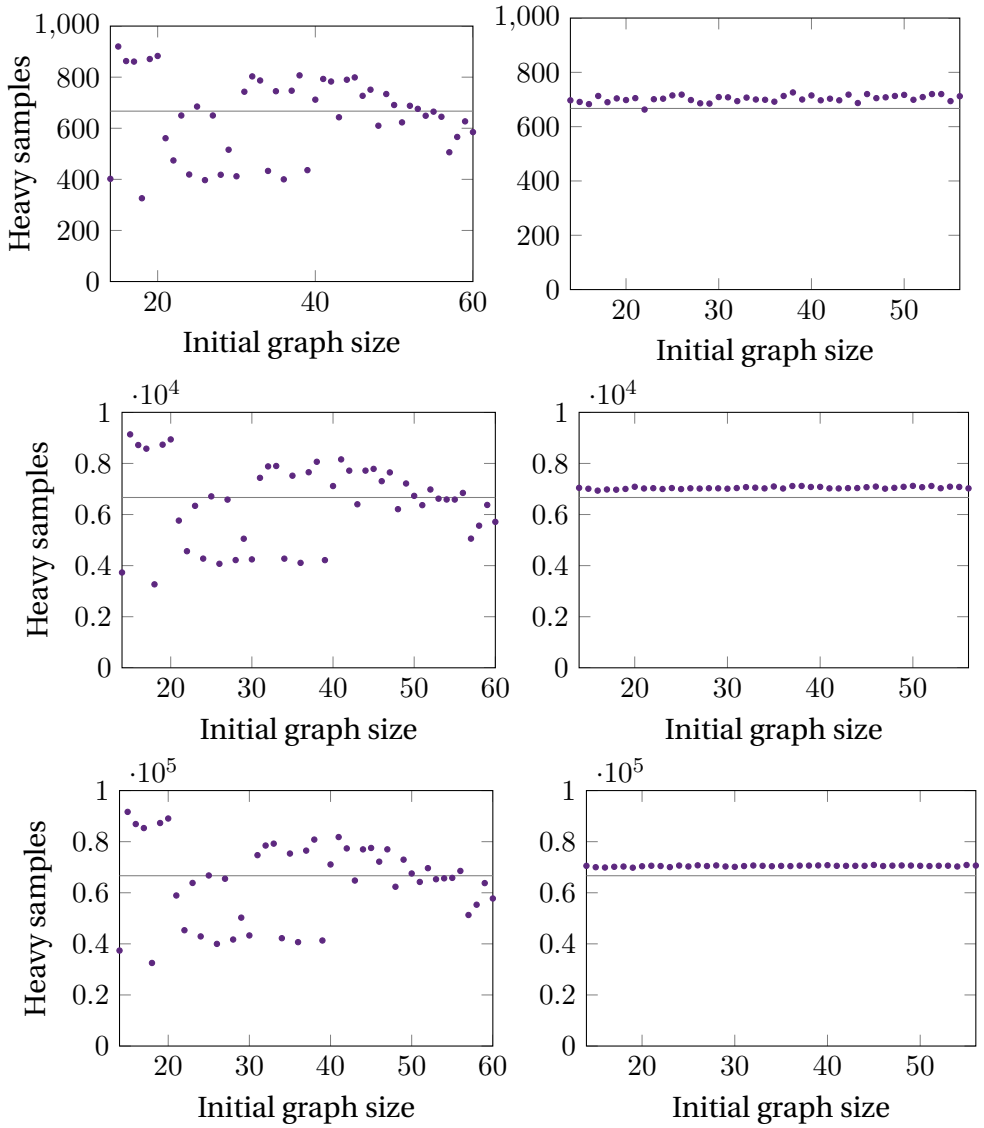


FIGURE 4.9. The HOG test performed on 1000, 10000, and 100 000 samples (top to bottom) collected using rejection sampling on unweighted (left) and complex weighted (right) graphs. The y-axis shows the number of samples drawn which had density above the median.

This shows some interesting features. Firstly, focusing on the unweighted graphs, this seems to perform worse at the HOG test than the quantum results. This could suggest that this test is not an effective method of verifying samples according to the density distribution. It could also indicate that there is some advantage in sampling from hafnians than sampling directly from the density, if the goal is to find higher density graphs (possibly because the hafnian distribution shows more significant differences between the heights of the peaks). On the other hand, as expected, it does outperform uniform sampling.

Secondly, focusing on the weighted graphs, it is interesting that the results are so consistent. This is likely due to the properties of the randomly generated matrices. We expect submatrices to have Gaussian i.i.d entries, and hence it is reasonable that in each case, the distributions of weights should have similar features.

We also show the results of the CHOG test, in Fig. 4.10. The trial distribution is the results from simulated GBS, and the adversary distribution is rejection sampling. These results are consistent, with quantum sampling almost always being favoured for unweighted graphs, and rejection sampling from the density distribution almost always being favoured for complex weighted graphs. As rejection sampling samples exactly from the density distribution, this firstly shows that CHOG is ineffective as a verification method for the density distribution. However, it confirms the prior results of this chapter, that sampling from hafnians seems to be an effective way to sample dense subgraphs in the unweighted case, but not in the complex weighted case.

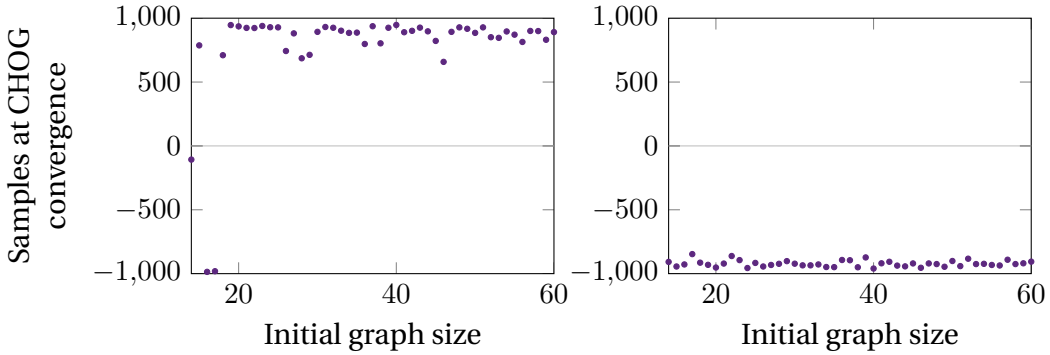


FIGURE 4.10. The CHOG test, with density as the ground truth, quantum sampling as the trial distribution, and rejection sampling as the adversary distribution. The left hand side is for unweighted graphs, and the right hand side is for complex weighted graphs.

We are also interested in the efficiency of running rejection sampling with complex-weighted graphs. Fig. 4.11 shows the number of samples that must be drawn for each accepted sample, the expected value of which is $H/\mathcal{N}$. This number increases significantly when the size of the subgraph increases (as it is chosen to be the nearest integer to $\sqrt{n}$), but otherwise seems to increase approximately linearly.

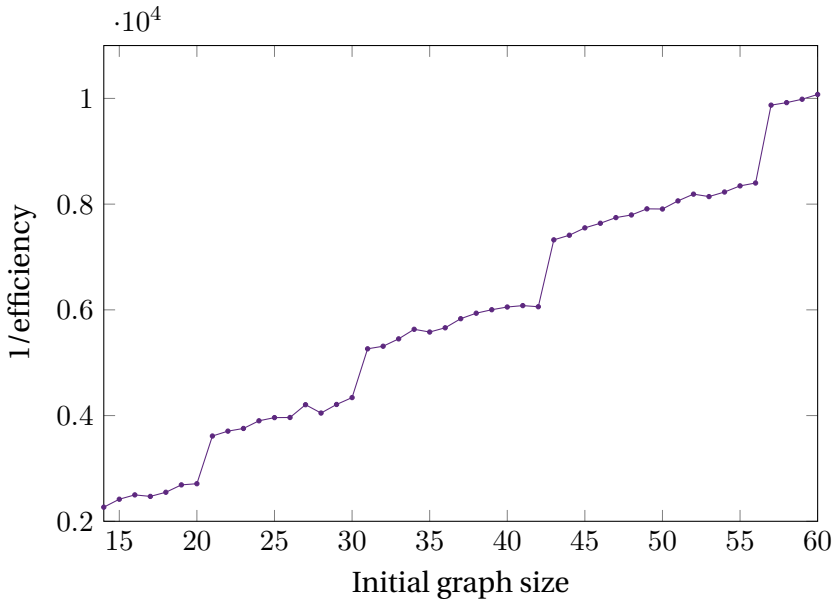


FIGURE 4.11. The number of samples that must be drawn for each accepted sample, applying rejection sampling to sample subgraphs of complex weighted graphs.

4.7 Discussion

In the previous chapter, as well as in former results such as [150], it seems as though the advantage that can be gained from using GBS to solve the DkS problem is limited, as we focus on positive-weighted graphs. In this chapter, we have extended this study to consider complex-weighted graphs as well. We have considered several possible reasons for the previous results: whether the robustness to error (particularly loss) arises from the use of positive weighted graphs, and hence the lack of quantum interference, whether this is due to the use of efficiently calculable densities instead of hafnians, or whether this is due to inherent limitations of sampling.

The Max Haf problem seems to be significantly more affected by loss than the dense subgraph finding problem, in both the unweighted and complex-weighted cases. This suggests that the results of Chapter 3 originate not just from using unweighted graphs, but from important differences between the density and the hafnian. Overall, a convincing case for quantum advantage primarily remains with the Max Haf problem on complex graphs (perhaps not surprisingly). However, given the complexity-theoretic evidence for the quantum advantage of GBS, we expect this to continue.

Our study focuses on sampling from the density distribution. At the simplest level, this is the motivation for the link between GBS and DkS. However, it is not necessarily the root cause of any advantage that quantum-enhanced algorithms may show. Given a distribution $p(x)$, the expected number of samples N that must be drawn before an exact sampler outputs x is:

	Unweighted	**Complex-weighted**
Density	– Quantum-enhanced results are robust to error, Chapter 3, [180] – Underlying GBS is efficiently simulable [150] – Sampling from the density is efficient, Section 4.6	– Less significant advantage from GBS expected [162] – Sampling from the density is efficient in certain cases, Section 4.6
$\|\text{haf}\|^2$	– Sampling is efficient [150] – Less significant robustness to loss, Section 4.4.1, [162]	– Sampling is not efficient – Less robustness to loss, Section 4.4.1

Table 4.1: The results of considering GBS for different optimisation problems. In particular, we note the existing results in this work and in the literature, for the Max Haf or DkS problem on unweighted or complex-weighted graphs.

$\bar{N} = \sum_{k=1}^{\infty} kp(x)(1-p(x))^{k-1} = \frac{1}{p}$. Therefore, if aiming to output the outcome associated with the highest probability, the objective is simply to sample from a distribution that maximises the probability associated with that outcome. The ideal distribution to sample from would have a single peak, at the densest subgraph – instead of sampling densities, sampling from a similar distribution, such as $|\text{haf}(A)|^2$, or even some power of the densities, may give a greater speedup. As is considered in [203], the hafnians of dense graphs can be greater than the hafnians of sparse graphs by orders of magnitude. Therefore, we do not claim that quantum advantage cannot exist for this task, but that it cannot be conjectured just from the difficulty of calculating hafnians, and the similarity of hafnians and density.

These results may extend beyond the DkS problem. The rejection sampling scheme that we have considered used the following conditions in order to sample from an underlying distribution of interest:

- the relative probabilities are efficient to calculate;
- the maximum probability is well defined and does not scale exponentially in comparison to the proposal distribution;
- the distribution obeys an overall weight condition, Eq. 4.20.

These conditions may not be unique, or necessary, but were sufficient in our construction to ensure efficient sampling. In particular, we could not ascertain analytically whether the final condition applies for the density distribution over the matrices that we have considered, but it agrees with numerical results.

Despite potentially offering an efficient sampling scheme, rejection sampling does not offer any advantage – when used as a subroutine to random sampling – over sampling from the uniform distribution. Sampling from the uniform distribution must be done as a first

step to rejection sampling, and then stochastic algorithms will often follow this by calculating the density to decide on how the sample is used, much like rejection sampling.

The verification tasks considered in this chapter may not have been particularly effective in benchmarking whether the samples were drawn from a distribution close to the density distribution, as rejection sampling failed these tests, which samples from the ground truth. However, they are useful as they represent the important property of whether the samples produced have high density. Therefore, they have provided insight that rejection sampling is a more valuable tool than GBS for sampling dense subgraphs for complex-weighted graphs, but that simulations or implementations of high-loss GBS may be more useful for unweighted graphs. They could also be a valuable tool when assessing future potential use cases of GBS.

To summarise, we expect that problems requiring the identification of subgraphs with many perfect matchings (related to the hafnian) are appropriate areas to expect a quantum advantage (using GBS). However, when considering the DkS problem, it is unlikely that GBS will be able to provide a considerable advantage over classical algorithms, even when using complex-weighted graphs – however, this may vary on a case-by-case basis.

4.8 Outlook and further work

For any groups interested in using GBS for complex DkS, further work is required to understand the relationship between the hafnian, or Torontonian, and complex density. In particular, if the correlation is minimal, then other methods for sampling from density-related distributions may be preferable over the use of GBS.

We have considered rejection sampling, which is a very basic proposal for sampling, and could be improved further by considering different sampling schemes (such as Metropolis independence sampling [81, 239]), or other Monte Carlo methods. The advantage of these algorithms often comes from similar 'hill-climbing' techniques that we have seen in simulated annealing, which can improve the efficiency over rejection sampling, and provides a greater advantage over sampling from the uniform distribution. This could be improved further by using a well-chosen distribution to sample from, such as a power of the density, or using further information about the graph (such as degree weights).

The results regarding the effectiveness of GBS with spectrally impure sources for graph problems are particularly interesting, and the assumption presented thus far that pure sources are required for accurate quantum interference may be challenged. As mentioned previously, it is still not well-understood how this source of error effects Gaussian states and the subsequent simulability of non-Gaussian measurements.

Finally, this work suggests that there could be advantages for graph problems in using classical, or more easily accessible, optical technologies. A further branch of study could be to focus on the application of thermal, or squashed, states to these problems.

CHAPTER 5

A complexity transition in displaced Gaussian boson sampling

Oh, I have three kids and no money. Why can't I have no kids and three money?

Homer Simpson

5.1 Preface

Statement of work: This work was carried out in collaboration with Zhenghao Li (who led the work on the project following initial discussions), Jacob F. F. Bulmer, Ryan L. Mann, Raj B. Patel and Ian A. Walmsley.

The author contributed to the development of the complexity proof in Section 5.5, and carried out the majority of the analysis in Section 5.6, with help from Z.L. and J.F.F.B. The author also contributed to the description and analysis of the efficient classical algorithms in Section 5.4. The numerical data presented in Section 5.4 and Appendix Section D were produced by Z.L.

Acknowledgements: I would like to give my sincere thanks to everyone with whom I worked on this project. I learned a lot and it was a very valuable experience for me.

5.2 Introduction

As previously mentioned, the state preparation stage of GBS involves squeezing and displacement, although many applications, analyses, and experiments involve zero displacement. In the initial GBS proposal, displacement is considered, but it is noted that 'displaced light does not increase the complexity of the problem' [145]. This matches current understanding,

with displacement having no major (asymptotic) complexity impact on simulations such as tensor network methods [177].

Nonetheless, there are several advantages of considering the complexity of simulating GBS with displacement. Displacements are needed for vibronic spectra calculations [240], and hence have been implemented as part of recent GBS demonstrations [147]. Displaced light may not increase the complexity of the problem, but it may be important to consider the ratio of the brightness of displaced and squeezed sources, to ensure that the complexity of simulating within a certain error does not decrease. This is described in [147], which presents one possible classical approximation technique if there is sufficient displacement. This may also be related to the decreased complexity of simulating GBS in different error regimes, particularly loss [175]. On the other hand, adding displacement increases the average photon number, which increases the probability of the harder-to-simulate measurement events that have a high photon number. Therefore, adding displacement is sometimes seen as a 'cheap' way to increase the complexity of simulating GBS experiments.

Calculating the probability of measurement outcomes in displaced GBS requires the calculation of a different matrix function, the loop hafnian [146], which is closely related to the matching polynomial of a graph [153, 241]. Ongoing research into the complexity of approximating the matching polynomial could translate into results about the hardness of approximately simulating displaced GBS. Nonetheless, additional work is necessary: firstly, to show how these two results are related, using the original complexity proof of boson sampling [84], and bipartite GBS [158], as example frameworks. There are also some restrictions on the regime in which certain results hold, and hence this could be used to show a transition in the complexity of simulating GBS for certain experimental parameters.

There are two main approaches in our work. Firstly, we employ algorithms used to approximate 'diagonally dominant' matching polynomials to show how GBS with high levels of displacement can be simulated efficiently, in Section 5.4. We then use results concerning matching polynomials to prove the complexity of GBS with displacements in the low-displacement regime, in Section 5.5. Finally, we consider the link between displacement and the simulation of lossy states, and how this relates to our complexity results, in Section 5.6.

5.3 Background

5.3.1 Evidence for the computational complexity of GBS

In Section 2.6.2, we considered the argument for the computational complexity of boson sampling. Understanding this seminal result, presented in [84], is an important foundation for proving the computational complexity of other models. Therefore, it is reviewed in the Appendix Section C. However, the proof that applies to BS does not apply directly to GBS. The general structure is the same, but here we highlight some aspects that are different.

Firstly, we must consider the different arguments that apply to hafnians. Using an estimate of the permanent, it is possible to approximate the value of the hafnian of the same matrix [169], although this incurs exponential error – hence, if there is a fully polynomial approximation scheme for the permanent of a matrix, this may not hold for a hafnian. Furthermore, hafnian calculation is in the same complexity class as permanents (as a generalisation of the permanent of counting the perfect matchings of general, instead of bipartite, graphs), but in general there is not the same level of evidence, or results, regarding either the hardness of approximation for various matrix classes, or the existence of efficient algorithms for this task.

It is interesting to note one difference between permanents and hafnians – a size $2n$ hafnian has the complexity of a size n permanent. This is clear from the matrix definition, as the matrices input to hafnians must be symmetric, and it is also clear from the graph theory definition, as the bipartite graph described by an $n \times n$ matrix has $2n$ vertices. Similarly, we can gain intuition in the quantum case – in boson sampling, non-Gaussian resources are required in photon creation (input state) and measurement, whereas in GBS, they are only required for measurement. This aligns with BS experiments having twice the stellar rank of GBS [242], a formalism which enables us to consider the non-Gaussian properties of different states and operations.

We also need to consider the problem of hiding. In the BS framework, we are able to consider the average-case hardness of the permanents of i.i.d. matrices as these can be 'hidden' inside unitary matrices. Using GBS, we are not considering submatrices of unitary matrices, but instead the B matrix (in the case of a pure state), URU^T. Alternative evidence for hiding in this case is given in [159].

Furthermore, we must consider anti-concentration, which is an important aspect of the complexity proof of BS, but remains a conjecture that is largely supported by numerical evidence. This is considered for GBS in [243], requiring that a sufficient number of modes must be squeezed at the input, compared to the number of photons that are measured at the output.

Finally, we note that the relationship between the number of photons and number of modes in BS experiments is chosen partly to avoid collisions. In GBS there is not a fixed photon number, but collisions become more likely with a higher average photon number, and so the squeezing parameters are chosen appropriately to keep the probability low. Although a certain amount of collisions can be acceptable [126, 158], if this is too high, it can allow classical spoofing [176].

5.3.2 Displaced Gaussian boson sampling

Displacement is easier to realise experimentally than squeezing, and coherent state preparation is more tolerant to loss [244]. It is also needed when using GBS to simulate molecular

vibronic spectra [240]. However, it has only been implemented in GBS experiments more recently [147, 208], which was realised by generating a coherent state and interfering it with a squeezed state. The work in [147] also uses an approximation to DGBS (the 'k-th order approximation') that relies on upper bounding the number of photons that are assumed to have originated with squeezing.

A displaced Gaussian state has non-zero mean, so is described by both the covariance matrix σ and a displacement vector $\boldsymbol{\delta}$. As described in [145–147], the probabilities associated with PNR measurements on these states are given by:

$$p(\mathbf{n}) = \frac{\mathrm{e}^{-\frac{1}{2}\boldsymbol{\delta}^\dagger \sigma_Q^{-1}\boldsymbol{\delta}}}{n_1!...n_m!\sqrt{|\sigma_Q|}}\mathrm{lhaf}\left(\mathrm{filldiag}(A_{\mathbf{n}}, \boldsymbol{\gamma}_{\mathbf{n}})\right), \tag{5.1}$$

in which filldiag$(X, \mathbf{v})$ replaces the diagonal elements of the matrix X with the elements of the vector $\mathbf{v}$, and $\boldsymbol{\gamma} = \boldsymbol{\delta}^\dagger \sigma_Q^{-1}$. Recall the loop hafnian from Section 2.8:

$$\begin{aligned}\mathrm{lhaf}\left(\mathrm{filldiag}(A, \boldsymbol{\gamma})\right) =& \mathrm{haf}(A) + \sum_{i_1, i_2, i_1 \neq i_2} \gamma_{i_1}\gamma_{i_2}\mathrm{haf}(A_{S-\{i_1,i_2\}}) \\ &+ \sum_{i_1,i_2,i_3,i_4,i_1\neq i_2\neq i_3\neq i_4} \gamma_{i_1}\gamma_{i_2}\gamma_{i_3}\gamma_{i_4}\mathrm{haf}(A_{S-\{i_1,i_2,i_3,i_4\}}) + ... + \prod_i^M \gamma_i.\end{aligned} \tag{5.2}$$

Note that $S - \{i_1, i_2\}$ is the set of modes in which photons were measured, except i_1 and i_2.

For a graph G that contains loops – edges that connect vertices to themselves, which occupy the diagonal elements of the adjacency matrix – this is:

$$\mathrm{lhaf}(G) = \sum_{\mathcal{M}\in \mathrm{SPM}(G)} \prod_{(i,j)\in\mathcal{M}} G_{(i,j)}, \tag{5.3}$$

in which SPM(G) is the set of perfect matchings that permit matching a vertex to itself - equivalently, a set of edges in the graph such that each vertex is connected to exactly one chosen edge.

It is generally considered that the complexity of simulating a displaced GBS (DGBS) experiment comes entirely from the difficulty of calculating the largest hafnian in Eq. 2.109, and that adding displacements has no significant impact on the complexity [145]. However, it seems intuitive that increasing the number of hafnians that must be calculated, as well as increasing the average photon number, should both increase the difficulty of simulating DGBS. On the other hand, sampling from a Gaussian state with only coherent light sources is classically efficient, and therefore with weak squeezing and bright coherent sources it is likely that the sampling error would be low with an efficient approximation.

In Section 5.5, we will use a restricted framework for DGBS in order to identify properties of the distribution (this is similar to the paradigmatic structure of GBS introduced in [144]). We assume that of the m modes, we input photons into K, with equal squeezing parameter

r and displacement parameter β, before entering an interferometer described by U. This means we can rewrite the outcome probabilities as:

$$\begin{aligned} p(\mathbf{n}) &= \frac{p_0}{n_1!\ldots n_m!}\left|\text{filldiag}(B_{\mathbf{n}}, \boldsymbol{\gamma}_{\mathbf{n}}^{(m)})\right|^2 \\ &= \frac{p_0}{n_1!\ldots n_m!}\tanh^K(r)\left|\text{filldiag}((UU^T)_{\mathbf{n}}, \boldsymbol{\gamma}_{\mathbf{n}}^{(m)})\right|^2 \\ &= \frac{p_0}{n_1!\ldots n_m!}\tanh^K(r)\left|\text{lhaf}\left(\text{filldiag}\left(U_{\mathbf{n},\mathbf{1}_K}U_{\mathbf{n},\mathbf{1}_K}^T, \boldsymbol{\gamma}_{\mathbf{n}}^{(m)}\right)\right)\right|^2 \end{aligned} \tag{5.4}$$

in which $U_{\mathbf{n},1_K}$ is constructed by repeated the i-th row of U n_i times and taking the first K columns, and

$$p_0 = \frac{\mathrm{e}^{-\frac{1}{2}\boldsymbol{\delta}^\dagger\sigma_Q^{-1}\boldsymbol{\delta}}}{\sqrt{|\sigma_Q|}} \tag{5.5}$$

is the vacuum probability. We also use $\boldsymbol{\gamma}^{(m)}$ to indicate the second m elements of $\boldsymbol{\gamma}$.

We find that:

$$\boldsymbol{\gamma}_i^{(m)} = (\beta^* - \beta\tanh(r))\sum_{\text{col}} U_{\mathbf{n},\mathbf{1}_K}, \tag{5.6}$$

where $\sum_{\text{col}}$ indicates that we sum across the columns in each row.

Hence, we can express the outcome probabilities as (using $\text{lhaf}(X, \mathbf{v})$ to indicate $\text{lhaf}(\text{filldiag}(X, \mathbf{v}))$):

$$p(\mathbf{n}) \propto \left|\text{lhaf}\left(U_{\mathbf{n},\mathbf{1}_K}U_{\mathbf{n},\mathbf{1}_K}^T, w\sum_{\text{col}} U_{\mathbf{n},\mathbf{1}_K}\right)\right|^2 \tag{5.7}$$

where we have introduced the parameter

$$w = \frac{\beta^* - \beta\tanh(r)}{\sqrt{\tanh(r)}} \tag{5.8}$$

which will be used to demonstrate the displacement/squeezing ratio. This will be useful as it represents a factor indicating the relative weights of the loops of the graph.

With zero displacement, the mean photon number is given by [145]:

$$\bar{n} = K\sinh^2(r). \tag{5.9}$$

We can see this by considering the sum of the marginals of photon counting measurements on each mode, as in Section 3.4.3.

5.3.3 The matching polynomial

There are several different ways of constructing functions related to the loop hafnian, which are relevant to different physical systems (other than just photons, which we have seen so far) [115].

Consider an edge-weighted and vertex-weighted graph G with vertices V and edges E. We will call the vector of vertex weights $\mathbf{v}$, and the matrix of edge weights is the adjacency matrix $\mathcal{A}$. Let us introduce a generating function [153]:

$$Z(G,\mathbf{v}) = \sum_{\mathcal{M}\in \mathrm{SPM}(G)} \prod_{\langle i,j\rangle\in\mathcal{M}} \mathcal{A}_{i,j} \prod_{k\notin\mathcal{M}} v_k. \tag{5.10}$$

We note that this is the same as the loop hafnian in Eq. 5.2.

If all vertices are equal weight (with that weight being some variable $x \in \mathbb{C}$) we produce the partition function, or matching polynomial [153]:

$$Z(G,x) = \sum_{\mathcal{M}\in \mathrm{SPM}(G)} \prod_{\langle i,j\rangle\in\mathcal{M}} \mathcal{A}_{i,j} x^{n-2|\mathcal{M}|}. \tag{5.11}$$

This is a polynomial in x, which parametrises the vertex weights.

Alternatively, we can parametrise the matching polynomial by the edge weights, multiplying $\mathcal{A}$ by the variable z, and setting $x = 1$, which gives:

$$\begin{aligned} g(z,\mathcal{A}) &= \sum_{\mathcal{M}\in \mathrm{SPM}(G)} z^{|\mathcal{M}|} \prod_{\langle i,j\rangle\in\mathcal{M}} \mathcal{A}_{i,j} \\ &= \mathrm{lhaf}(z\mathcal{A}, \mathbf{1}_n). \end{aligned} \tag{5.12}$$

Finally, we also consider the univariate case in which all edge weights are the same (we set $\mathcal{A}_{ij} = z$) [241]:

$$g_G(z) = \sum_{\mathcal{M}\in \mathrm{SPM}(G)} z^{|\mathcal{M}|}. \tag{5.13}$$

An example of this on a fully-connected 4-vertex graph is given in Fig. 5.1. The matching polynomial for the empty graph ($|V|= 0$) is defined to be 1.

We can link Eqs. 5.10 and 5.12, following [153]:

$$\begin{aligned} Z(G;\mathbf{v}) &= \prod_{l\in V} v_l \sum_{M\in \mathrm{SPM}(G)} \prod_{\langle i,j\rangle\in M} \mathcal{A}_{ij} \left(\prod_{p\in V} v_p\right)^{-1} \prod_{k\notin M} v_k \\ &= \prod_{l\in V} v_l \sum_{M\in \mathrm{SPM}(G)} \prod_{\langle i,j\rangle\in M} \mathcal{A}_{ij} \left(\prod_{k\in M} v_k\right)^{-1} \\ &= \left(\prod_{l\in V}\right) v_l g_{\tilde{G}}(z). \end{aligned} \tag{5.14}$$

where $g_{\tilde{G}}$ is the multivariate matching polynomial for the reweighted graph $\tilde{G}$, where $\tilde{\mathcal{A}}_{ij} = \tilde{\mathcal{A}}_{ij}/v_i v_j$.

The confusing use of many different definitions and somewhat clashing notation reflects the variety of research done on the topic. In general, the different types of matching polynomial presented are used in different contexts, but it is possible to convert between them using

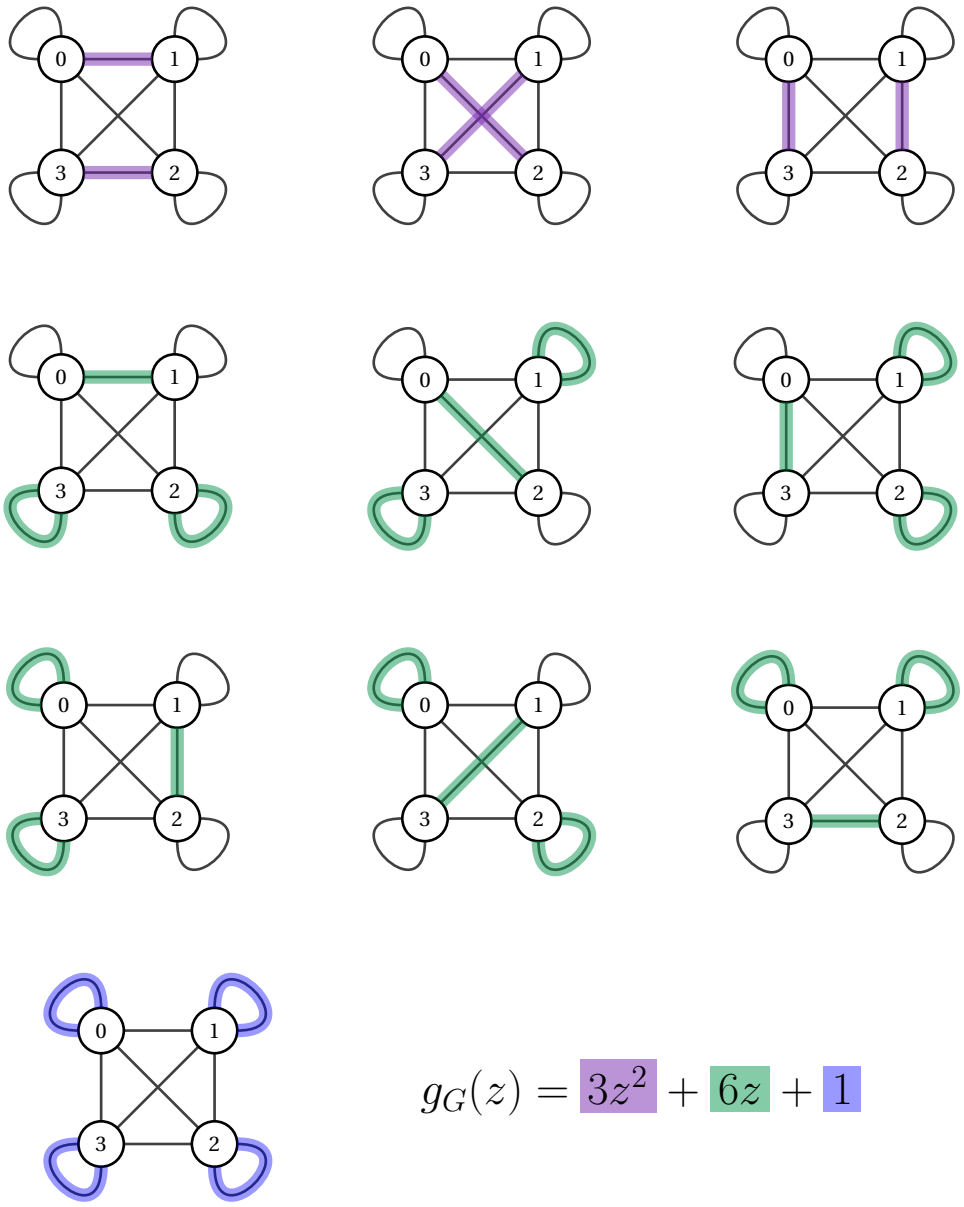


FIGURE 5.1. The univariate matching polynomial of a fully-connected 4-vertex graph.

a change of variable, although we note that being able to make this connection necessitates that all vertex (or loop) weights are non-zero. Similarly to the hafnian, exact calculation of the matching polynomial is worst-case #**P**-hard. However, we will once again be interested in the difficulty of approximating the matching polynomial, which will depend on the graph structure and the value of the variable(s).

In [241], the definition in Eq. 5.13 is used, and, building on prior work, presents several results for graphs of maximum degree Δ. Firstly, if z is not a negative real number such that $z < -1/(4(\Delta-1))$, there exists a FPRAS (fully polynomial-time randomised approximation scheme) [167], or a deterministic FPTAS if the maximum degree of the graph is constant (with regards to the graph size) [241, 245]. On, the other hand, for all graphs such that $\Delta \geq 3$, for real $z < -1/(4(\Delta-1))$, it is $\#\mathbf{P}$-hard to approximate $Z_G(z)$ [241]. We will go on to consider the implications of both aspects of this work – firstly, how approximation methods can be applied to DGBS, and also how confident we can be in the hardness of approximating DGBS in the appropriate regimes.

5.3.4 Noise and displacement

Let us consider in more detail the action of noise upon a Gaussian state. As we will see, this will demonstrate the relevance of considering approximation schemes for DGBS, even when a zero-mean state is input to a GBS experiment. We will follow [81, 132, 148].

We can write the density matrix of a quantum state in terms of its characteristic function:

$$\rho = \frac{1}{(2\pi)^n} \int_{\mathbb{R}^{2n}} \mathrm{d}(\boldsymbol{\xi}) \chi(\boldsymbol{\xi}) \hat{D}(\boldsymbol{\xi}) \tag{5.15}$$

where, for Gaussian states:

$$\chi_G(\boldsymbol{\xi}) = \exp\left(-\frac{1}{2}(\Omega^T\boldsymbol{\xi})^T V(\Omega^T\boldsymbol{\xi}) - i(\Omega\mathbf{d})^T\boldsymbol{\xi}\right). \tag{5.16}$$

In this case, we are using the real representation of the covariance matrix and displacement vector. This is particularly useful for mixed states, that can be expressed as a convex sum over pure states.

Now consider that we modify the state, so that $V \mapsto V + W$ and $\rho \mapsto \rho'$:

$$\rho' = \frac{1}{(2\pi)^n} \int_{\mathbb{R}^{2n}} \mathrm{d}\boldsymbol{\xi} \exp\left(-\frac{1}{2}(\Omega^T\boldsymbol{\xi})^T (V+W)(\Omega^T\boldsymbol{\xi}) - i(\Omega\mathbf{d})^T\boldsymbol{\xi}\right) \hat{D}(\boldsymbol{\xi}). \tag{5.17}$$

In order for the state to continue to satisfy the uncertainty relation, we require $W \geq 0$ (that is, W is positive semidefinite). Instead of the pure state with covariance matrix $V+W$ and displacement vector $\mathbf{d}$, we now show that we can use the state with covariance matrix V and displacement vector $\mathbf{d}+\mathbf{r}$, where $\mathbf{r}$ is distributed according to the multivariate Gaussian distribution:Note that here, W is acting as a covariance matrix in the classical probability sense.

$$p(\mathbf{r}) = \frac{\mathrm{e}^{-\frac{1}{2}\mathbf{r}^T W^{-1}\mathbf{r}}}{(2\pi)^n \sqrt{\det(W)}}. \tag{5.18}$$

That is, we use a mixed state over different values of $\mathbf{r}$ distributed according to Eq. 5.18.

We will assume that $\det(W) > 0$, but the proof for $\det(W) = 0$ is similar, although slightly more complicated. In a GBS scheme, this would mean choosing a random displacement from Eq. 5.18 for each sample drawn.

The state is then (we hope to show that this is equal to Eq. 5.17):

(5.19)
$$\begin{aligned}\rho' &= \int_{\mathbb{R}^{2n}} \mathrm{d}\mathbf{r} \frac{\mathrm{e}^{-\frac{1}{2}\mathbf{r}^T W^{-1}\mathbf{r}}}{(2\pi)^n\sqrt{\det(W)}} \hat{D}(\mathbf{r})\rho\hat{D}(\mathbf{r})^\dagger \\ &= \frac{1}{(2\pi)^n}\int_{\mathbb{R}^{4n}} \mathrm{d}\mathbf{r}\mathrm{d}\boldsymbol{\xi}\hat{D}(\mathbf{r})\frac{\exp\left(-\frac{1}{2}\mathbf{r}^T W^{-1}\mathbf{r} - \frac{1}{2}(\Omega^T\boldsymbol{\xi})^T V(\Omega^T\boldsymbol{\xi}) - i(\Omega\mathbf{d})^T\boldsymbol{\xi}\right)}{(2\pi)^n\sqrt{\det(W)}}\hat{D}(\boldsymbol{\xi})\hat{D}(-\mathbf{r}) \\ &= \frac{1}{(2\pi)^n}\int_{\mathbb{R}^{4n}} \mathrm{d}\mathbf{r}\mathrm{d}\boldsymbol{\xi}\frac{\exp\left(-\frac{1}{2}\mathbf{r}^T W^{-1}\mathbf{r} - \frac{1}{2}(\Omega^T\boldsymbol{\xi})^T V(\Omega^T\boldsymbol{\xi}) - i(\Omega\mathbf{d})^T\boldsymbol{\xi}\right)}{(2\pi)^n\sqrt{\det(W)}}\hat{D}(\mathbf{r})\hat{D}(\boldsymbol{\xi})\hat{D}(-\mathbf{r}).\end{aligned}$$

Using $\hat{D}(\mathbf{r}_1)\hat{D}(\mathbf{r}_2) = \exp\left(i\mathbf{r}_1^T\Omega\mathbf{r}_2/2\right)\hat{D}(\mathbf{r}_1+\mathbf{r}_2)$: (we've seen this before, but you can confirm using the BCH expressions)

$$\rho' = \frac{1}{(2\pi)^n}\int_{\mathbb{R}^{4n}} \mathrm{d}\mathbf{r}\mathrm{d}\boldsymbol{\xi}\frac{\exp\left(-\frac{1}{2}\mathbf{r}^T W^{-1}\mathbf{r} - \frac{1}{2}(\Omega^T\boldsymbol{\xi})^T V(\Omega^T\boldsymbol{\xi}) - i(\Omega\mathbf{d})^T\boldsymbol{\xi} + i\mathbf{r}^T\Omega\boldsymbol{\xi}\right)}{(2\pi)^n\sqrt{\det(W)}}\hat{D}(\boldsymbol{\xi}). \tag{5.20}$$

We can now integrate over $\mathbf{r}$ by using the equality:

$$\int_{\mathbb{R}^{2n}} \mathrm{d}\mathbf{r}\mathrm{e}^{-\mathbf{r}^T A\mathbf{r}+\mathbf{r}^T\mathbf{b}} = \frac{\pi^n}{\sqrt{\det(A)}}\mathrm{e}^{\mathbf{b}^T A^{-1}\mathbf{b}/4}, \tag{5.21}$$

and we arrive at Eq. 5.17, as required.

This representation of error is particularly useful when considering the simulation of GBS. For some mixed state, in order to simulate it using this scheme, we first wish to decompose the covariance matrix into $V = T + W$, where T is the covariance matrix of a pure state (with lower average photon number than V), and the Gaussian noise $W \geq 0$. Using the Williamson decomposition, this can alternatively be represented by a symplectic transformation S acting on a thermal state with diagonal covariance matrix D:

$$V = SDS^\dagger = \frac{1}{2}SS^\dagger + S(D - \frac{1}{2}\mathbb{1})S^\dagger = T + W. \tag{5.22}$$

To sample from this state we can use the process described in [148]. We sample (i.e. choose according to the classical distribution) a vector $\mathbf{R}$ from:

$$p(\mathbf{r}) = \frac{\exp\left(-(1/2)(\mathbf{r}-\mathbf{d})^T W^{-1}(\mathbf{r}-\mathbf{d})\right)}{\sqrt{\det(2\pi W)}}. \tag{5.23}$$

(d represents the vector of means of our input state - given that we are discussing a non-displaced state that undergoes loss, we will take this to be zero). We then continue by sampling the pure state with covariance matrix T and vector of means $\mathbf{r}$.

This decomposition is part of the scheme for classical simulation using tensor networks described in [177]. By separating the state into the 'classical' and 'quantum' parts (i.e. $V = T+W$), they are able to simulate the classical part efficiently and use tensor network methods to simulate the quantum part with high accuracy.

5.4 Approximation schemes to the loop Hafnian

In this section we will present an approximation scheme to the loop Hafnian. This is based on the work in [115, 246]. This method is based on finding the Taylor expansion of the logarithm of the matching polynomial, which is only valid in regions in which the polynomial is non-zero. Intuitively, we would expect classical approximation schemes to be valid in regions where the magnitude of the displacement/squeezing ratio w is high. Let's see if this intuition is good!

5.4.1 Taylor expansion approximation

Let $g(z)$ be a complex polynomial of degree d, which is non-zero for some region of z (and therefore the logarithm is well-defined). Within this region, let $f(z) = \ln(g(z))$, with $|z| \leq 1$, and consider its Taylor polynomial (truncated to degree l)

$$T_l(z) = f(0) + \sum_{k=1}^{l} \left(\frac{d^k}{dz^k} f(z)|_{z=0} \right) \frac{z^k}{k!}. \tag{5.24}$$

In [115] (with further details in [246]), it is shown that, if $g(z) \neq 0$ for all $|z| \leq \beta$, then:

$$|f(z) - T_l(z)| \leq \frac{d|z|^{l+1}}{(l+1)\beta^l(\beta - |z|)} \; \forall \; |z| \leq 1. \tag{5.25}$$

(In brief, this is shown by expanding $g(z)$ and its logarithm $f(z)$ in terms of its roots α_i, and given that $|\alpha_i| > \beta$ we can then use this to upper bound the size of higher order terms.) This is shown on page 24 of [115] and it's worth mentioning that the proof is not too involved, although omitted here, so is worth looking at. There is a more detailed proof in [246]. In order to use $T_l(z)$ to evaluate $f(z)$ with error $< \epsilon$, we therefore require:

$$l = \frac{\ln d - \ln \epsilon}{1 - |z|/\beta}. \tag{5.26}$$

You can double check this by substitution into Eq. 5.25. This is an additive error approximation to $f(z)$, which is a multiplicative error approximation to $g(z)$, with factor $\mathrm{e}^{\epsilon} - 1$.

To see this, note that we are interested in:

$$\left| \frac{\mathrm{e}^{f(z)} - \mathrm{e}^{T_l(z)}}{\mathrm{e}^{f(z)}} \right| = \left| 1 - \mathrm{e}^{T_l(z) - f(z)} \right|, \tag{5.27}$$

but $|e^{T_l(z)-f(z)}| \leq e^{|T_l(z)-f(z)|} = e^{\epsilon}$. Further details are given in [246].

Now let us return to DGBS. Consider an $n \times n$ matrix $\tilde{A}$, which defines a matching polynomial $g(z; \tilde{A}) = \mathrm{lhaf}(z\tilde{A}, \mathbf{1}_n)$ with maximum degree $d = \lfloor \frac{n}{2} \rfloor$. We assume as well that there is a disc of radius β of the complex plane on which it is guaranteed that $g(z; \tilde{A}) \neq 0$ (this will be addressed further in the following section).

As previously described, we can then take the Taylor expansion of $g(z;\tilde{A})$ at z, and truncate it to degree l, chosen according to Eq. 5.26, for a multiplicative error approximation. The runtime of computing $T_l(z)$ scales with the coefficient of its leading term, $\frac{d^l}{dz^l}\ln g(0)$. This requires the evaluation of a polynomial number of derivatives of maximum degree l of $g(z;\tilde{A})$ at $z=0$ [115]. Full derivatives are given in [115]; we require computing $O(l^2)$ terms constructed from $g^{(k)}$, and the coefficients $g^{(k)}(0)$. These can be expressed as a sum over hafnians:

$$\begin{aligned} g(0;\tilde{A}) &= 1 \\ g'(0;\tilde{A}) &= \sum_{i,j,i\neq j} \tilde{A}_{ij} \\ g^{(k)}(0;\tilde{A}) &= k! \sum_{\substack{S\subseteq\{1,\dots,n\} \\ |S|=2k}} \mathrm{haf}(\tilde{A}_S) \end{aligned} \tag{5.28}$$

where $\tilde{A}_S$ is constructed from $\tilde{A}$ by keeping the i-th row and column of $\tilde{A}$ if and only if $i\in S$. Therefore, $g^{(k)}(0;\tilde{A})$ can be calculated in time $n^{O(l)}$, and the approximation $T_l(z)$ is quasipolynomial, with runtime $n^{O(\ln(n)-\ln(\epsilon))}$.

The exponential scaling of the error may make it seem as though the term 'efficient' is not appropriate. However, we can see that an arbitrarily small error can be produced without causing an exponential increase in runtime. To see this, we first describe the error as $\tilde{\epsilon}$-multiplicative, where $\tilde{\epsilon}=\mathrm{e}^{\epsilon}-1$. Then the algorithm runs in time $n^{O(\ln(n)-\ln(\ln(\tilde{\epsilon}+1)))}$, and hence we retain the quasi-polynomial runtime.

5.4.2 Non-zero regions

We have seen a quasi-polynomial approximation scheme, however in order for this to be applicable, the remaining task is to identify on what regions of the complex plane $g(z)$ is non-zero. This will depend on the matrix that defines the matching polynomial. We would expect that experiments with higher displacement are more efficient to simulate, corresponding to diagonally-dominant matrices.

To understand this connection, first observe that:

$$\mathrm{lhaf}(z\tilde{A},\mathbf{1}_n) = 1 + \sum_{\mathcal{M}\in \mathrm{SPM}(G)\setminus\emptyset} \prod_{\langle i,j\rangle\in\mathcal{M}} z\tilde{A}_{i,j}, \tag{5.29}$$

where we use $\emptyset$ to indicate the matching containing no edges between two vertices. Hence, if $|\tilde{A}_{i,j}|\ll 1$, then the higher-order terms decay quickly and $\mathrm{lhaf}(z\tilde{A},\mathbf{1}_n)\approx 1$, particularly if z is also small.

To reduce the run time of the classical approximation scheme, we would like the order l to be low, which means $|z|/\beta$ should be as small as possible. In particular, as we wish to approximate $g(z)$ at $z=1$, we would like the radius β in which $g(z)$ is non-zero to be very large, which is the case if $|\tilde{A}_{i,j}|$ is very small for all i,j.

Nonetheless, this is not a well-defined condition. In this section, we consider two previously studied results that give two different bounds on A_{ij}, in order to guarantee the existence of a non-zero region of $g(z)$. For a complex symmetric $n \times n$ matrix $\tilde{A}_{ij}$, these are [247]:

$$|\tilde{A}_{ij}| < \frac{1}{\mathrm{e}(2n-3)} \; \forall \; i \neq j, i, j \in [1, n] \tag{5.30}$$

and [248]:

$$\sum_{j \neq i} |\tilde{A}_{ij}| < \frac{1}{n-1} \; \forall \; i, j \in [1, n]. \tag{5.31}$$

The condition in Eq. 5.30 was originally developed in terms of the abstract polymer model, and has been adapted for quantum problems [249]. The result is:

Lemma 5.1. *Let $\tilde{A}$ be a complex symmetric $n \times n$ matrix whose elements satisfy:*

$$|\tilde{A}_{ij}| \leq \frac{\lambda}{\mathrm{e}(2n-3)} \; \forall \; i \neq j \tag{5.32}$$

for some real $0 < \lambda < 1$. Then the matching polynomial $g(z; \tilde{A}) \neq 0$ for $|z| < \frac{1}{\lambda}$.

The abstract polymer model describes a finite set of polymers (in this case, edges of the graph) which have a compatibility relation, $\sim$. We consider edges to be compatible if they do not share a vertex. If each edge $z_i \in E$ has a (complex) weight $w(z_i)$, then [247] shows that $g(1, \mathcal{A})$ (as defined in Eq. 5.12) is non-zero if:

$$\sum_{z' \in E : z' \not\sim z} |w(z')| \, \mathrm{e}^{f(z')+g(z')} \leq f(z) \qquad \forall z \in E, \tag{5.33}$$

where f and g are real, positive weighted functions on E.

In order to arrive at Eq. 5.30, we choose constant functions so that $f(z) = a$ and $g(z) = d$. We also note that, for any graph, the maximum number of edges that can share a vertex with the chosen edge z_{ij} is $2n-3$: these are $z_{ij'}$, where $j' \neq i, j$, of which there are $n-2$, and $z_{i'j}$, where $i' \neq i, j$, of which there are $n-2$, as well as the edge itself z_{ij}, which we include in this set. Therefore, we can rewrite the left hand side of Eq. 5.33:

$$\sum_{z' \in E : z' \not\sim z} |w(z')| \, \mathrm{e}^{a+d} \leq (2n-3) \mathrm{max}(|\mathcal{A}_{ij}|) \mathrm{e}^{a+d}. \tag{5.34}$$

Therefore, we require:

$$\mathrm{max}(|\tilde{A}_{ij}|) \leq \frac{a}{\mathrm{e}^{a+d}(2n-3)} \tag{5.35}$$

for some $a, d \in \mathbb{R}$ – and it is always possible to find a, d to satisfy this inequality if Eq. 5.30 is satisfied.

The condition in Eq. 5.31 is developed in [248], and is used in a similar manner to show regions where the permanent can be calculated efficiently. The result is:

Lemma 5.2. *Let $\tilde{A}$ be a complex symmetric $n \times n$ matrix whose elements satisfy:*

$$\sum_{j \neq i} |\tilde{A}_{ij}| < \frac{\lambda}{n-1} \; \forall \, i, j \in [1, n] \tag{5.36}$$

for some real $0 < \lambda < 1$. Then the matching polynomial $g(z; \tilde{A}) \neq 0$ for $|z| < \frac{1}{\lambda}$.

We can then state the main result of this section:

Theorem 5.3. *Let $\tilde{A}$ be a complex symmetric $n \times n$ matrix, and $\lambda < 1$ a positive real number, that together satisfy at least one of Eqs. 5.36 and 5.32. Then lhaf$(\tilde{A}, \mathbf{1}_n)$ can be approximated to multiplicative error $\mathrm{e}^{\epsilon} - 1$ in time $n^{O_\lambda(\ln(n) - \ln(\epsilon))}$, where the subscript λ indicates that the coefficient implicit in O_λ only depends on λ.*

Proof. Firstly, by using lemmas 5.1 and 5.2, then lhaf$(z\tilde{A}, \mathbf{1}_n)$ is non-zero in the region of $|z| < \frac{1}{\lambda}$, and $\frac{1}{\lambda} > 1$.

Therefore, we can use the Taylor approximation method in Section 5.4, with the order truncated to:

$$l = \frac{\ln d - \ln \epsilon}{1 - \lambda}, \tag{5.37}$$

to approximate $\ln\Big(\text{lhaf}(z\tilde{A}, \mathbf{1}_n)\Big)$ at $z = 1$, in time $n^{O(l)}$ with error ϵ, which gives multiplicative error factor $\mathrm{e}^{\epsilon} - 1$ to lhaf$(\tilde{A}, \mathbf{1}_n)$. □

This may be the case for some submatrices of the A matrix, i.e. for certain sampling outcomes $\mathbf{n}$. If it is true for all outcomes, then we are able to sample from the entire distribution with multiplicative error. We note that in general, Eq. 5.36 is more likely to occur than Eq. 5.32, however Eq. 5.32 can still occur independently, and unlike Eq. 5.36, if it holds for the A matrix, it also holds for any submatrix. Beware, however, that in our construction, the submatrices that are associated with different outcomes are also rescaled by different amounts, due to the fact that there may be different displacement in each mode. Hence, that doesn't mean that if this holds for the main $\tilde{A}$ matrix, it will hold for any outcome that we may like to calculate! We assume that λ is some known constant; the algorithm is more efficient for lower values of λ. If λ is chosen incorrectly, so that it defines a disc containing zeros, then the error bound ϵ is no longer valid and therefore the approximation is no longer useful – however, this may not be apparent when implementing the method described, so λ must be chosen with care.

We now return to the framework described in Section 5.3.2, where K input modes are equally squeezed. The function that would we like to calculate is:

$$\text{lhaf}\left(U_{\mathbf{n},1_K} U^T_{\mathbf{n},1_K}, w \sum_{\text{col}} U_{\mathbf{n},1_K}\right).$$

However, using the rescaling described in Eq. 5.14, we can instead use:

$$\left(\prod_{k=1}^{m} w\left(\sum_{\text{col}} U_{\mathbf{n},1_K}\right)_k\right) \text{lhaf}\left(\frac{1}{w^2}\tilde{B}, \mathbf{1}_{\mathbf{n}}\right), \tag{5.38}$$

where

$$\tilde{B}_{ij} = \frac{(U_{\mathbf{n},1_K} U^T_{\mathbf{n},1_K})_{ij}}{(\sum_{\text{col}} U_{\mathbf{n},1_K})_i (\sum_{\text{col}} U_{\mathbf{n},1_K})_j}. \tag{5.39}$$

Note that here we have made the assumption that all loop weights are non-zero (that is, the displacement at the output of each mode is non-zero). Experimentally, using a fully-connected interferometer, this only requires displacement on a single input mode. However, it is a more complex question whether this holds for the average unitary matrix. Assuming w is non-zero, this is the question of whether the sum over the columns of a unitary matrix is non-zero (for each column). Although we leave this as an open question, we note that in the case of simulation of certain experiments, this can be used as a check to see whether our simulation methods apply, and furthermore only a small deviation from zero is required, and therefore as an approximation could be implemented by changing a zero loop weight to be (arbitrarily) small.

Furthermore, assuming there are no collisions, and that the number of photons scales as $O(\sqrt{m})$, we expect that the submatrices $U_{\mathbf{n},1_K}$ should have i.i.d. entries distributed according to Gaussians, that is, $X \sim \mathcal{N}(0,1)^{n\times n}_{\mathbb{C}}$. Although we cannot directly use this to find maximum values of $|\tilde{B}_{ij}|$ as required, we can use numerical data to understand the likely dependence of non-zero regions on w.

These data[1] are shown in Fig. 5.2. For each n, 10 000 random matrices of dimension $n \times n$ are generated of the form:

$$\tilde{X}_{ij} = \frac{(XX^T)_{ij}}{(\sum_{\text{col}} X)_i (\sum_{\text{col}} X)_j}. \tag{5.40}$$

Then the minimum roots of the corresponding matching polynomial $g(z, \tilde{X})$ were found, using NumPy. The median and first quartile of the absolute values of these were found using bootstrapping, using the SciPy function, with a confidence level of 0.95. This takes the original sample set, of size N, and draws N samples from this set (with replacement), and calculates the median of these. This process done repeatedly to produce the bootstrap distribution; when this produces a distribution that is sufficiently symmetric about the median and contains a high enough proportion of the initial N values, then the median is returned. This method is useful based on assuming that the statistics of interest have a normal

[1] Numerical data shown in this chapter were collected by Zhenghao Li, and the code used and data shown can be made available on request. Fig. 5.2 is reproduced, with permission, from a figure produced by Z. L. for an upcoming publication.

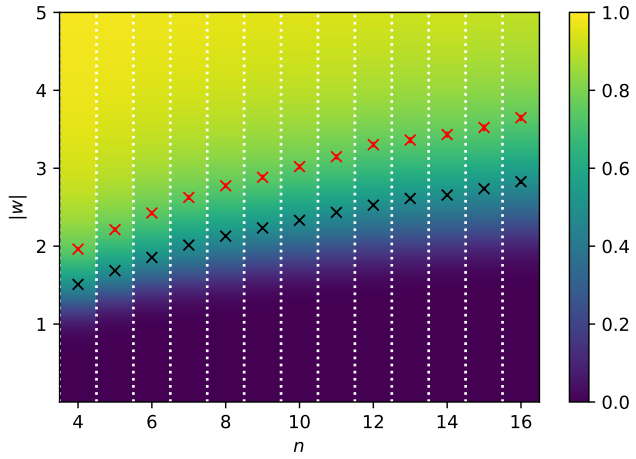


FIGURE 5.2. Given a random $n \times n$ matrix $\tilde{X}$ (see Eq. 5.40) and value $|w|$, we show the probability for the magnitude of the minimum root of the matching polynomial of $g(z, \tilde{X})$ (that is, z such that $g(z, \tilde{X}) = 0$) to be greater than $1/|w|^2$. The black crosses correspond to a probability of 0.5, and the red crosses correspond to a probability of 0.75 – this is the probability that the efficient simulation algorithms are effective for this example.

distribution, and it allows us to gain a more detailed understanding of the overall population using a limited number of samples [250]. Using this, we can estimate the probability that $g(z, \tilde{X})$ is non-zero for $z < \frac{1}{|w|^2}$.

In Fig. 5.3, we show the ratio $|\beta|^2/\sinh^2(r)$, where β is the displacement value and r is the squeezing value, that corresponds to the value of $|w|$ represented by the red and black crosses in Fig. 5.2, which are the $|w|$ values required for 50% or 75% of the roots to be greater than $\frac{1}{|w|^2}$, respectively. This ratio corresponds to the ratio of the expected number of photons from displacement, against the expected number of photons from squeezing.

These results show that the size of w above which we expect the approximations to be effective increases slowly, although this corresponds to regimes in which displacement dominates over squeezing in terms of the number of photons generated, by one or more orders of magnitude.

5.5 The computational complexity of DGBS

Now we will consider the argument for the computational complexity of DGBS. Having considered schemes that can classically simulate DGBS with very high displacement, it is useful to instead consider the evidence that DGBS cannot be efficiently simulated if the displacement is sufficiently low. Typically, the complexity of the GBS scheme is discussed

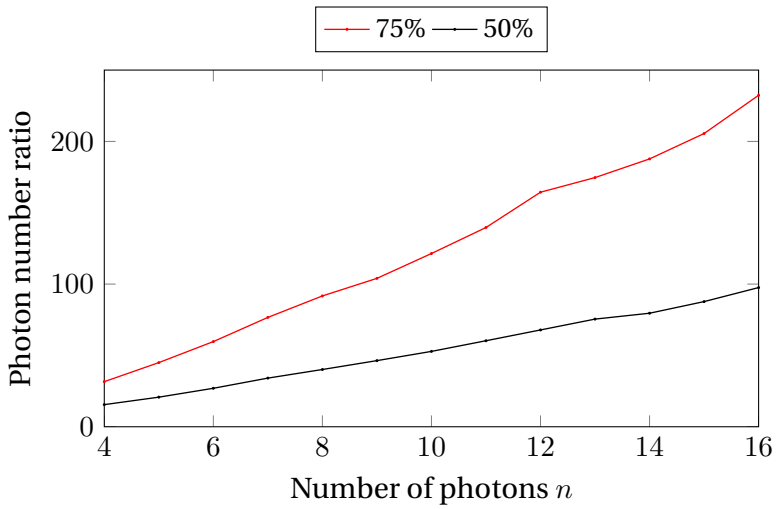


FIGURE 5.3. We show the ratio of photons from displacement vs squeezing associated to the values of $|w|$ corresponding to the points in Fig. 5.2.

based solely on the complexity of calculating the hafnian. As we have seen, displacement has an impact on the complexity that must be taken into account when considering any claim of quantum advantage. However, we are able to use existing results concerning the hardness of approximating the matching polynomial – as described in Section 5.3.3, this is closely related to the loop hafnian. Overall, we consider the framework of [84] – further details are included in Section 2.6.2, and Appendix Section C.

5.5.1 Complexity of exact simulation

Firstly, we consider the hardness of exactly simulating DGBS. That is, we assume that there exists a classical algorithm, $\mathcal{C}$, that can sample exactly from the ground truth described by the experimental settings, returning outcomes according to the probability distribution in Eq. 5.1. Using Stockmeyer's theorem, this algorithm would be able to estimate the probability of an outcome in the complexity class $\mathbf{FBPP}^{\mathbf{NP}^{\mathcal{C}}}$.

It is first important to note that any complex-weighted graph with loops can be encoded into an arbitrary DGBS framework. That is, the B matrix can be chosen to be any complex symmetric adjacency matrix (up to some scaling parameter, as we have seen), and then the displacement values can be chosen so that γ is any complex-weighted vector encoding the loop weights. These can also be constructed to 'hide' matrix $B_{\mathbf{n}}$ and vector $\gamma_{\mathbf{n}}$ corresponding to a particular outcome $\mathbf{n}$.

The loop hafnian is worst-case $\#\mathbf{P}$-hard to approximate with multiplicative error, even in the case of non-zero diagonal elements [167, 241]. Therefore, the efficient classical algorithm $\mathcal{C}$ would be able to solve a problem in $\#\mathbf{P}$ in $\mathbf{FBPP}^{\mathbf{NP}}$, which, according to Toda's theorem

(Thm. 2.1), would cause a collapse of the polynomial hierarchy.

5.5.2 Complexity of approximate simulation

For a more realistic picture, we would like to know the computational complexity of sampling from a distribution with a constant TVD from the distribution described by DGBS. We must therefore consider the average-case hardness of calculating the matching polynomial in the DGBS setup. If there are only a few outcomes that correspond to probabilities that are difficult to calculate, these could be approximated with high error, while the other output probabilities are approximated with low error, therefore still maintaining a constant TVD.

In order to do this, we use the model for DGBS described in Section 5.3.2, that has equal squeezing in K input modes. We require that the displacement in each mode is non-zero at the output in order to use the specified reduction to the matching polynomial. That is, we will consider the difficulty of approximating $g(z = \frac{1}{w^2}; \tilde{X})$, where $\tilde{X}$ is constructed according to Eq. 5.40. Note that this is slightly different from the previous section; instead of choosing to evaluate the polynomial at $z = 1$, we use z to parametrise the displacement/squeezing ratio.

In order to have the 'hiding' property, that the square submatrices of U have elements that have i.i.d. elements according the normal distribution, we impose the conditions that $K \leq \sqrt{m}$ and $\bar{n} \leq \sqrt{m}$ [145, 159]. Given this condition, we can post-select to only consider samples with $n = \bar{n}$ samples, in polynomial time [145]. This also enables us to assume that the probability of collisions is negligibly low (although recent work suggests the collision-free condition is not required for the hardness of BS [126], and future work may show the same for GBS).

5.5.2.1 Average-case hardness of the loop hafnian

The complexity proof of GBS relies on the hafnians-of-Gaussians conjecture, that approximating the hafnian of a Gaussian matrix (X) with multiplicative error is in $\#\mathbf{P}$ in the average case (that is, for a sufficiently high proportion of matrices across the set X). We would like to consider a similar case, but for our function of interest, $g(z = \frac{1}{w^2}; \tilde{X})$.

As well as the input matrix $\tilde{X}$, this relies on the parameter w. We conjecture the main statement of this section, which is that this function is average case hard to approximate with multiplicative error as long as $|w|$ is sufficiently low:

Conjecture 5.4. *Let $\tilde{X}$ be an $n \times n$ matrix constructed according to Eq. 5.40, where $X \sim \mathcal{N}(0,1)_{\mathbb{C}}^{n\times n}$. Then, there exists some positive real $\tilde{w} > 0$, which may depend on n, such that the following problem is $\#\mathbf{P}$-hard for all complex w that satisfies $|w| \leq \tilde{w}$:*

For any $\epsilon, \delta > 0$, estimate $g(\frac{1}{w^2}; \tilde{X})$ to within multiplicative error ϵ with probability at least $1 - \delta$ in $\mathrm{poly}(n, 1/\epsilon, 1/\delta)$ time.

The 'average-case' part is here represented in the parameter δ. The probability $1-\delta$ that the efficient algorithm (which shouldn't exist) will depend on the choice of input matrix. Imagine that we choose δ to be exponentially low. In that case, we think that the algorithm should work for almost any matrix. However, in this case, $\mathrm{poly}(n, 1/\epsilon, 1/\delta)$ is no longer efficient – this corresponds to now including more hard cases in the potential inputs.

It may seem confusing that a polynomial runtime and $\#\mathbf{P}$-hard are included in the same statement about complexity. However, we cannot say that the polynomial runtime doesn't exist, as it may turn out that problems in $\#\mathbf{P}$ can be solved efficiently. Therefore, $\#\mathbf{P}$-hard says that if this runtime was possible, then it would be possible for all problems in $\#\mathbf{P}$ (with possible polynomial overheads).

There are several reasons to suggest the dependence on the magnitude of w. Firstly, physical intuition, as discussed previously, suggests that DGBS should become easier to simulate as the displacement increases. In the limit of $w = 0$ (in which case the matching polynomial described is no longer well-defined, but the loop hafnian reduces to the hafnian), this becomes zero-mean GBS, where the hafnian-of-gaussians conjecture holds.

The work of [241] considers the matching polynomial of unweighted graphs, and shows that this is efficient to approximate in regions of the independent variable ('edge parameter') z where the matching polynomial contains zeros (as described in Section 5.4), whereas it is $\#\mathbf{P}$-hard to approximate in regions of z that contain zeros of the matching polynomial. As discussed in Section 5.4.2, these regions depend on the ratio w, with the zeros being more likely to occur in regions of low displacement.

It is also possible that there is some dependence on the phase of w, given that the worst-case hardness of the matching polynomial is dependent on the phase of the independent variable [241]. We leave this as an open question.

Further work is also needed to evidence the average-case hardness. For the case of boson sampling, this is conjectured for the permanent of Gaussian matrices, due to the worst-case/average-case equivalence of the hardness of the permanent over finite fields [84, 251]. Furthermore, it is also average case $\#\mathbf{P}$-hard to exactly compute $\mathrm{per}(X)$. Nonetheless, this does not hold for the permanent of Gaussian matrices, and is an open problem. We also note that, if the anti-concentration condition holds, this is also equivalent to additive error approximation, with error $\epsilon n!\,/\mathrm{poly}(n)$.

There is no evidence that the construction of $\tilde{X}$ provides any additional structure that would facilitate the efficient approximation of $g(\frac{1}{w^2}; \tilde{X})$. Furthermore, similarly to the permanent, computing the partition function $Z(G, \mathbf{v})$ is $\#\mathbf{P}$-hard to do exactly unless the graph has a particular structure [252]. Therefore, Conjecture 5.4 is not unlikely to be true, but further work is needed to understand whether this is the case.

We also note that the hardness of calculating the matching polynomial under certain circumstances cannot be used to design hard DGBS experiments. We require average-case

hardness to ensure that the submatrices related to the sampled outcomes are likely to correspond to difficult-to-calculate probabilities, to prevent the error in the distribution being concentrated around a few probabilities.

5.5.2.2 Anti-concentration

The final condition that is necessary for the computational complexity of DGBS is the anti-concentration condition. This is necessary to link the multiplicative hardness in Conjecture 5.4 to the additive error that a hypothetical classical sampler is able to achieve using Stockmeyer's theorem.

The probability distribution of measurement outcomes depends on the amount of displacement in the experiment, and therefore we expect that the anti-concentration condition should as well. Similarly to the case of Conjecture 5.4, we expect anti-concentration to hold for $w = 0$ (for the hafnian), but the evidence does not apply to experiments with a high amount of displacement. Therefore, we propose:

Conjecture 5.5. *Let n be a positive integer. Then there exists a real $\tilde{w} > 0$, such that for all complex w that satisfy $|w| \leq \tilde{w}$, and for any real $\beta > 0$, there exists a polynomial $P = \mathrm{poly}(n, 1/\beta)$ such that*

$$\Pr_{X \in \mathcal{G}_{n,n}(0,1)} \left[\left| \mathrm{lhaf}\left(XX^T, w \sum_{col} X \right) \right|^2 < \frac{1}{P} \frac{n!\, F_n(w)}{2^n} \right] < \beta.$$

The factor of $n!$ accounts for the standard deviation ($\sqrt{n!}$) of loop hafnians, and the normalisation factor $\frac{2^n}{F_n(w)}$ is required to account for the postselection into the subspace of outcomes with n photons and no collisions [145]:

$$F_n(w) = \sum_{\substack{\mathbf{m} \geq 0 \\ \sum m_j = n}} \prod_{j=1}^{n} \frac{1}{m_j!} \left| H_{m_j}\left(\frac{iw}{\sqrt{2}} \right) \right|^2, \tag{5.41}$$

where H_i are the Hermite polynomials. That is, if we consider the post-selected probability $\tilde{p}(\mathbf{n})$ of drawing a certain outcome from the set of 'good' outcomes Ω (having total photon number n and no collisions), the condition in Eq. 5.41 can instead be expressed as:

$$\Pr_{\mathbf{n} \in \Omega} \left[\tilde{p}(\mathbf{n}) \geq \frac{n!}{P} \right] > 1 - \beta. \tag{5.42}$$

Previous studies into anti-concentration have failed to reach analytical proofs of their correctness (see e.g. [84, 126]), and therefore these remain as conjectures. As in most other work on this topic, we rely primarily on numerical evidence to support this proposal, which is presented in the Appendix Section D. In particular, it seems that the condition generally holds for constant w, although it is less clear whether this is the case if $|w|$ increases with

n. Hence, this suggests that the hardness transition could require a fixed displacement to squeezing ratio (as represented by w).

To understand the significance of the anti-concentration conjecture (here, we follow the argument from [78]), recall that we have proposed a classical simulator that draws samples from the distribution $q(x)$, where $\sum_x |p(x) - q(x)| < 2\epsilon$. Then, we can use Markov's inequality [253]:

$$\begin{aligned} \text{Pr}_{x\in\Omega}\left[|p(x)-q(x)| \geq \frac{\mathbb{E}\left(|p(x)-q(x)|\right)}{\delta}\right] &\leq \delta \\ \text{Pr}_{x\in\Omega}\left[|p(x)-q(x)| \geq \frac{2\epsilon}{\delta|\Omega|}\right] &\leq \delta, \end{aligned} \tag{5.43}$$

where Ω is the space of outcomes that we are interested in. Similarly, we can find that:

$$\begin{aligned} \text{Pr}_{q(x)\in\Omega'}\left[q(x) \geq \frac{\mathbb{E}\left(q(x)\right)}{\delta'}\right] &\leq \delta' \\ \text{Pr}_{x\in\Omega'}\left[q(x) \geq \frac{1}{\delta'|\Omega|}\right] &\leq \delta', \end{aligned} \tag{5.44}$$

where this time, $q(x)$ is considered over all possible outcomes, but also all possible problem instances, represented by Ω' (in our case, this is over all Gaussian i.i.d. matrices X). Therefore, although the expected probability is still $\frac{1}{|\Omega|}$, δ and δ' are independent.

Using Stockmeyer's theorem, the hypothetical classical algorithm gives an approximation $\tilde{q}(x)$ to $q(x)$ such that $|\tilde{q}(x) - q(x)| \leq \tilde{\epsilon} q(x)$, so:

$$|\tilde{q}(x) - p(x)| \leq |\tilde{q}(x) - q(x)| + |q(x) - p(x)| \leq \frac{\tilde{\epsilon}}{\delta'|\Omega|} + \frac{2\epsilon}{\delta|\Omega|}, \tag{5.45}$$

with probability $(1-\delta)(1-\delta')$.

This currently represents an additive error approximation. However, we now assume that:

$$\text{Pr}_{x\in\Omega'}\left[p(x) \geq \frac{1}{P|\Omega|}\right] \geq 1 - \beta, \tag{5.46}$$

where $P = \text{poly}(n, 1/\epsilon)$, and we also set $\tilde{\epsilon} = 2\epsilon$, $\delta' = \delta$, to give:

$$|\tilde{q}(x) - p(x)| \leq \frac{4\epsilon P}{\delta} p(x), \tag{5.47}$$

which is a polynomial multiplicative estimation, with probability that is polynomial in $1/\beta$, $1/\delta$.

We also require a proof that estimation of the loop hafnian is polynomial-time reducible to estimation of its absolute value, which is omitted here, but can be done using the same procedure as [84]. Furthermore, we introduce the factor of $n!$ in our anti-concentration conjecture to match other anti-concentration studies in this area, although we note that this requires a more complicated proof structure that is also omitted here (but is found in [84]).

We also note that this is a sketch proof; the argument of [84] uses a different technique that allows us to introduce a factor of $n!$. Inspired by this, we therefore focus on this weaker form of the anti-concentration condition (as presented in Conjecture 5.5) when gathering numerical evidence. However, if this factor is omitted (or shown to be unnecessary) we can use the proof as described here.

We therefore arrive at the main result of this section:

Theorem 5.6. *Consider a DGBS problem as described in Section 5.3.2, with $K = \bar{n} = \sqrt{m}$, that satisfies $|w| \leq |\tilde{w}|$ for the $|\tilde{w}|$ required by Conjectures 5.4 and 5.5, and whose n-photon, collision-free post-selected distribution is $\tilde{p}_{\mathbf{n}}$. Suppose there exists a classical algorithm that, given some error bound $\epsilon > 0$, can sample from $q_{\mathbf{n}}$, such that $\mathrm{TVD}(\tilde{p}_n, q_n) \leq \epsilon$, in time $\mathrm{poly}(n, 1/\epsilon)$. If Conjectures 5.4 and 5.5 are true, then the polynomial hierarchy collapses to the third level.*

5.6 Loss and DGBS

5.6.1 Lossy GBS as DGBS

As described in Section 5.3.4, mixed Gaussian states can be decomposed into a pure Gaussian state and classical noise:

$$V = SDS^\dagger = \frac{1}{2}SS^\dagger + S(D - \frac{1}{2}\mathbb{1})S^\dagger = T + W, \tag{5.48}$$

where we are then able to sample from measurement on this state by instead considering a Gaussian state with covariance matrix T, and displacement randomly chosen according to Eq. 5.23. We would like to use this method to simulate a zero-mean Gaussian state affected by loss. If the loss is sufficiently high, then the component W will dominate, meaning that the state can be simulated by considering states where the majority of photons are generated through displacement.

In this section, we consider an analytic decomposition of the state. This does not result in a clear loss threshold over which the efficient techniques described can be used. However, we instead present some comparisons to our numerical results that imply that the scaling of this technique may make it favourable for simulating high-loss, high-photon-number experiments. Furthermore, the expressions given here that characterise the relationship between displacement and loss may be useful for future classical simulation efforts.

Consider a lossy channel that acts equally on all modes, with transmission $\eta = 1 - L$. This applies the transformation

$$V \mapsto \eta V + \frac{1}{2}(1 - \eta)\mathbb{1} \tag{5.49}$$

to the Gaussian state. A pure Gaussian state with no displacement initially has $W = 0$, $T = \frac{1}{2}SS^\dagger$, where S is a symplectic transformation acting on the vacuum.

A potential first guess for the decomposition would be:

$$\tilde{V} = \frac{1}{2}SS^\dagger + S\frac{1}{2}(\eta-1)S^\dagger + \frac{1}{2}(1-\eta)\mathbb{1}, \tag{5.50}$$

but this is not allowed, because W in this case is not positive semidefinite (can see this by looking at the one mode squeezing example, and if S is positive semidefinite, can show this using Sylvester's criteria). We can also understand this intuitively – in this case the T matrix stays the same, and we have $W = S\frac{1}{2}(\eta-1)S^\dagger + \frac{1}{2}(1-\eta)\mathbb{1}$. But if T is the same, the squeezing of our pure state stays the same, and we are adding displacement (i.e. increasing the mean photon number) to simulate loss (which should decrease the mean photon number). So we should modify T to decrease the brightness of the squeezed sources, compensating for added displacement. So let's do the Williamson decomposition properly and not cut corners.

The decomposition $V = T+W$ is not unique. Here, we use the Williamson decomposition to do this analytically [254, 255], using the following method [132, 256]: We use this method (linked) to do the Williamson decomposition (note that we're working with real matrices here, and we will do a basis change to the complex basis later on).

1. We find the eigenvalues of the matrix $V^{1/2}\Omega V^{1/2}$ (these are the same as the eigenvalues of $Z = \Omega V$). These have the form $i\lambda_k$, and the values on the diagonal of D are the $\lambda_k > 0$ (i.e. $D = \text{diag}(\lambda_1...\lambda_n, \lambda_1...\lambda_n)$). These have associated eigenvectors $\mathbf{z}_k = \mathbf{u}_k + i\mathbf{v}_k$ (these are different to the eigenvectors of Z).

2. We set $U = \sqrt{2}[\mathbf{u}_1, ..., \mathbf{u}_n, \mathbf{v}_1, ..., \mathbf{v}_n]$, and set

$$S = D^{-1/2}U^T V^{1/2}. \tag{5.51}$$

First, consider the state created by single-mode squeezing and no interferometer (we will say all m modes have the same squeezing value r). We have:

$$V = \frac{1}{2}\text{diag}(\mathrm{e}^{-2r}, ..., \mathrm{e}^{-2r}, \mathrm{e}^{2r}, ..., \mathrm{e}^{2r}), \tag{5.52}$$

$$Z = \Omega V = \frac{1}{2}\begin{pmatrix} 0 & \cdots & 0 & \mathrm{e}^{2r} & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & \mathrm{e}^{2r} \\ -\mathrm{e}^{-2r} & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & -\mathrm{e}^{-2r} & 0 & \cdots & 0 \end{pmatrix}. \tag{5.53}$$

We have $\lambda = \frac{1}{2}i$ and so $D = \frac{1}{2}\mathbb{1}$ as we would expect.

After adding loss:

$$Z \mapsto \frac{1}{2}\begin{pmatrix} 0 & (1-\eta)\mathbb{1} + \eta\text{diag}(\mathrm{e}^{2r}, ..., \mathrm{e}^{2r}) \\ -(1-\eta)\mathbb{1} - \eta\text{diag}(\mathrm{e}^{-2r}, .., \mathrm{e}^{-2r}) & 0 \end{pmatrix}. \tag{5.54}$$

We find the eigenvalues:

$$\begin{aligned}\lambda^2 &= \frac{1}{4}(1-\eta+\eta e^{2r})(-\eta e^{-2r}+\eta-1)\\ &= -\frac{1}{4}-(\eta-\eta^2)\sinh^2(r).\end{aligned} \tag{5.55}$$

So $\lambda = \pm i\sqrt{\frac{1}{4}+(\eta-\eta^2)\sinh^2(r)}$, and $\lambda_k = +\sqrt{\frac{1}{4}+(\eta-\eta^2)\sinh^2(r)}$. (Finding the associated eigenvectors is omitted for now).

We end up with $T = \frac{1}{2}SS^\dagger$, where $S = D^{-1/2}U^TV^{1/2}$, with $D = \text{diag}(\lambda_k)$.

T can now be expressed as $\frac{1}{2}SS^T$, where

$$S = \frac{1}{\sqrt{2}}\lambda_k^{-1/2}U^T\ \text{diag}\Big((\eta e^{-2r}+1-\eta)^{1/2},...,(\eta e^{-2r}+1-\eta)^{1/2},(\eta e^{2r}+1-\eta)^{1/2},...,(\eta e^{2r}+1-\eta)^{1/2}\Big).$$

We can then implement the Bloch-Messiah (or Takagi-Autonne, as this is complex-symmetric) decomposition [212] to find the squeezing parameters of our lossy state – that is, the effective squeezing parameter that would be required to construct the pure state described by T. Given that U is orthogonal, we can write this directly as

$$S = U^T\text{diag}(e^{-r'}...,e^{-r'},e^{r'},...,e^{r'}),$$

where r' is the new squeezing parameter. We have:

$$e^{-r'} = \frac{1}{\sqrt{2}}\lambda_k^{-1/2}(\eta e^{-2r}+1-\eta)^{1/2} \tag{5.56}$$

and

$$e^{r'} = \frac{1}{\sqrt{2}}\lambda_k^{-1/2}(\eta e^{2r}+1-\eta)^{1/2}. \tag{5.57}$$

As a sanity check, we can see that when multiplied together, these give 1. Therefore,

$$\begin{aligned}r' &= \ln\left(\frac{\sqrt{\eta e^{2r}+1-\eta}}{\sqrt[4]{1+4(\eta-\eta^2)\sinh^2(r)}}\right)\\ &= \frac{1}{4}\ln\left(\frac{\eta e^{2r}+1-\eta}{\eta e^{-2r}+1-\eta}\right),\end{aligned} \tag{5.58}$$

where we have recovered the result from [177] (which was derived for a one-mode squeezed state).

Intuitively we would expect that if we just need the magnitude of squeezing and displacement, this should be independent of the interferometer settings. Indeed, in the decomposition, this should just be equivalent to just changing U, which does not change the squeezing value.

Now we have seen how the squeezing brightness changes when simulating loss, and we should also consider the expected displacement value. We have:

$$\begin{aligned} W &= S(D - \frac{1}{2}\mathbb{1})S^T \\ &= 2(\lambda_k - \frac{1}{2})T. \end{aligned} \tag{5.59}$$

Hence we sample a displacement from the multivariate normal distribution described by Eq. 5.23.

Now is a good time to return to part 2 of the Williamson decomposition procedure (using Eq. 5.51) to calculate U^T (still assuming no interferometer). We need to find the eigenvectors of the matrix:

(5.60)

$$\begin{aligned} \sigma^{1/2}\Omega\sigma^{1/2} &= \frac{1}{2}\begin{pmatrix} 0 & \sqrt{(\eta e^{2r} + 1 - \eta)(\eta e^{-2r} + 1 - \eta)}\mathbb{1} \\ -\sqrt{(\eta e^{2r} + 1 - \eta)(\eta e^{-2r} + 1 - \eta)}\mathbb{1} & 0 \end{pmatrix} \\ &= \begin{pmatrix} 0 & -\lambda_k\mathbb{1} \\ \lambda_k\mathbb{1} & 0 \end{pmatrix}, \end{aligned}$$

which has eigenvalues $\pm i\lambda_k$. These look like $\mathbf{z}_k = \mathbf{u}_k + i\mathbf{v}_k$:

$$\begin{aligned} \mathbf{u}_k = &\frac{1}{\sqrt{2}}(1, 0, ..., 0)^T, \\ &\frac{1}{\sqrt{2}}(0, 1, 0, ..., 0)^T ... \end{aligned} \tag{5.61}$$

$$\begin{aligned} \mathbf{v}_k = &\frac{1}{\sqrt{2}}(0, ..., 0, -1, 0, ...)^T, \\ &\frac{1}{\sqrt{2}}(0, ..., 0, 0, -1, ...)^T, \\ &\frac{1}{\sqrt{2}}(0, ..., -1)^T \end{aligned} \tag{5.62}$$

so that

$$U = \begin{pmatrix} \mathbb{1} & 0 \\ 0 & -\mathbb{1} \end{pmatrix}. \tag{5.63}$$

Hence, $W = (\lambda_k - \frac{1}{2})SS^T = (\lambda_k - \frac{1}{2})\frac{1}{\lambda_k}\sigma = (1 - \frac{1}{2\lambda_k})\sigma$. This means that W is diagonal, and so the elements of the displacement vector are independently normally distributed.

We then draw the displacement vector from Eq. 5.23. If the displacement vector chosen is $\mathbf{d} = (\mathbf{d}_1, \mathbf{d}_2)$, then the complex displacement vector is $\boldsymbol{\delta} = \frac{1}{\sqrt{2}}(\mathbf{d}_1 + i\mathbf{d}_2, \mathbf{d}_1 - i\mathbf{d}_2)$.

Then we see that we can apply to each mode a displacement β that is normally distributed with mean 0. The real part has variance $\frac{1}{4}(1 - \frac{1}{2\lambda_k})(\eta e^{-2r} + 1 - \eta)$ and the imaginary part has

variance $\frac{1}{4}(1-\frac{1}{2\lambda_k})(\eta e^{2r}+1-\eta)$. We introduce the notation $\mu_\pm = \eta e^{\pm 2r}+1-\eta$, and note that $\lambda = \frac{1}{2}\sqrt{\mu_+\mu_-}$, and $e^{2r'} = \sqrt{\frac{\mu_+}{\mu_-}}$.

We are interested in the magnitude of $w = \frac{\beta^* - \beta\tanh(r')}{\sqrt{\tanh(r')}}$,which follows a Hoyt distribution [257]. To understand this, we can use prior research into 2-dimensional distributions. It is expected that half of the sampled displacements have $|w|$ less than or equal to the circular error probable (CEP), which is given by [258]:

$$\text{CEP}(w) = 1.17741 s'(1-0.163357\rho'^2 - 0.041694\rho'^4), \tag{5.64}$$

where

$$s' = \frac{1}{2}(\text{STD}(\text{Re}(w))^2 + \text{STD}(\text{Im}(w))^2) \tag{5.65}$$

$$\rho' = \frac{\text{STD}(\text{Im}(w))^2 - \text{STD}(\text{Re}(w))^2}{\text{STD}(\text{Re}(w))^2 + \text{STD}(\text{Im}(w))^2}. \tag{5.66}$$

Using:

$$\text{STD}(\text{Re}(w)) = \sqrt{\frac{(1-\tanh(r'))^2}{2\tanh(r')}(1-\frac{1}{2\lambda})\mu_-} \tag{5.67}$$

$$\text{STD}(\text{Im}(w)) = \sqrt{\frac{(1+\tanh(r'))^2}{2\tanh(r')}(1-\frac{1}{2\lambda})\mu_+}, \tag{5.68}$$

we find that:

$$\begin{aligned}\rho' &= \frac{(1+\tanh(r'))^2\mu_+ - (1-\tanh(r'))^2\mu_-}{(1+\tanh(r'))^2\mu_+ + (1-\tanh(r'))^2\mu_-} \\ &= \frac{\mu_+^2-\mu_-^2}{\mu_+^2+\mu_-^2} \\ &= \frac{4\eta\sinh(r)\cosh(r)(2\eta\sinh^2(r)+1)}{4\eta^2(2\sinh^4(r)+\sinh^2(r))+4\eta\sinh^2(r)+1}\end{aligned} \tag{5.69}$$

and:

$$\begin{aligned}s' &= \frac{1-(1/2\lambda)}{4\tanh(r')}((1+\tanh(r'))^2\mu_+ + (1-\tanh(r'))^2\mu_-) \\ &= (\mu_+\mu_- - \sqrt{\mu_+\mu_-})\frac{\mu_+^2+\mu_-^2}{\mu_+\mu_-(\mu_+-\mu^-)} \\ &= \frac{\eta^2(4\sinh^2(r)+8\sinh^4(r))+4\eta\sinh^2(r)+1}{2\eta\sinh(r)\cosh(r)}(1-1/\sqrt{4\eta\sinh^2(r)(1-\eta)+1}).\end{aligned} \tag{5.70}$$

Here these are in full:

$$
\begin{aligned}
\rho' &= \frac{(1+\tanh(r'))^2\mu_+ - (1-\tanh(r'))^2\mu_-}{(1+\tanh(r'))^2\mu_+ + (1-\tanh(r'))^2\mu_-} \\
&= \frac{e^{2r'}(\eta e^{2r}+1-\eta) - e^{-2r'}(\eta e^{-2r}+1-\eta)}{e^{2r'}(\eta e^{2r}+1-\eta) + e^{-2r'}(\eta e^{-2r}+1-\eta)} \\
&= \frac{(\eta e^{2r}+1-\eta)^2 - (\eta e^{-2r}+1-\eta)^2}{(\eta e^{2r}+1-\eta)^2 + (\eta e^{-2r}+1-\eta)^2} \\
&= \frac{\eta^2(e^{4r}-e^{-4r}) + 2\eta(e^{2r}-e^{-2r}) - 2\eta^2(e^{2r}-e^{-2r})}{\eta^2(e^{4r}+e^{-4r}) + 2\eta(e^{2r}+e^{2r}) - 2\eta^2(e^{2r}+e^{-2r}) + 2 - 4\eta + 2\eta^2} \\
&= \frac{2\eta^2\sinh(4r) + 4\eta\sinh(2r) - 4\eta^2\sinh(2r)}{2\eta^2\cosh(4r) + 4\eta\cosh(2r) - 4\eta^2\cosh(2r) + 2 - 4\eta + 2\eta^2} \\
&= \frac{4\eta^2\cosh(2r)\sinh(2r) + 8(\eta-\eta^2)\cosh(r)\sinh(r)}{2\eta^2(1+2\sinh^2(2r)) + 4(\eta-\eta^2)(1+2\sinh^2(r)) + 2 - 4\eta + 2\eta^2} \\
&= \frac{8\eta\sinh(r)\cosh(r)(2\eta\sinh^2(r)+1)}{2\eta^2(1+8\sinh^2(r)\cosh^2(r)) + 4(\eta-\eta^2)(1+2\sinh^2(r)) + 2 - 4\eta + 2\eta^2} \\
&= \frac{8\eta\sinh(r)\cosh(r)(2\eta\sinh^2(r)+1)}{2\eta^2(8\sinh^4(r)+4\sinh^2(r)) + 8\eta\sinh^2(r) + 2} \\
&= \frac{4\eta\sinh(r)\cosh(r)(2\eta\sinh^2(r)+1)}{4\eta^2(2\sinh^4(r)+\sinh^2(r)) + 4\eta\sinh^2(r) + 1}
\end{aligned} \tag{5.71}
$$

and:

$$
\begin{aligned}
s' &= \frac{1-(1/2\lambda)}{4\tanh(r')}((1+\tanh(r'))^2(\eta e^{2r}+1-\eta) + (1-\tanh(r'))^2(\eta e^{-2r}+1-\eta)) \\
&= \frac{(1-1/\sqrt{\mu_+\mu_-})(e^{r'}+e^{-r'})}{4(e^{r'}-e^{-r'})}(4\frac{e^{2r'}}{(e^{r'}+e^{-r'})^2}\mu_+ + 4\frac{e^{-2r'}}{(e^{r'}+e^{-r'})^2}\mu_-) \\
&= \frac{1-1/\sqrt{\mu_+\mu_-}}{e^{2r'}-e^{-2r'}}(e^{2r'}\mu_+ + e^{-2r'}\mu_-) \\
&= \frac{\sqrt{\mu_+\mu_-}-1}{\sqrt{\mu_+\mu_-}(\sqrt{\frac{\mu_+}{\mu^-}} - \sqrt{\frac{\mu_-}{\mu_+}})}(\sqrt{\frac{\mu_+}{\mu_-}}\mu_+ + \sqrt{\frac{\mu_-}{\mu_+}}\mu_-) \\
&= (\mu_+\mu_- - \sqrt{\mu_+\mu_-})\frac{\mu_+^2+\mu_-^2}{\mu_+\mu_-(\mu_+-\mu^-)} \\
&= \left(1-1/\sqrt{(\eta e^{2r}+1-\eta)(\eta e^{-2r}+1-\eta)}\right)\frac{(\eta e^{2r}+1-\eta)^2 + (\eta e^{-2r}+1-\eta)^2}{\eta(e^{2r}-e^{-2r})} \\
&= \frac{2\eta^2\cosh(4r) + 4\eta\cosh(2r) - 4\eta^2\cosh(2r) + 2 - 4\eta + 2\eta^2}{4\eta\sinh(r)\cosh(r)}(1-1/\sqrt{4\eta\sinh^2(r)(1-\eta)+1}) \\
&= \frac{\eta^2(2\cosh^2(2r) - 4\cosh^2(r) + 2) + 4\eta\sinh^2(r) + 1}{2\eta\sinh(r)\cosh(r)}(1-1/\sqrt{4\eta\sinh^2(r)(1-\eta)+1}) \\
&= \frac{\eta^2(4\sinh^2(r) + 8\sinh^4(r)) + 4\eta\sinh^2(r) + 1}{2\eta\sinh(r)\cosh(r)}(1-1/\sqrt{4\eta\sinh^2(r)(1-\eta)+1}).
\end{aligned} \tag{5.72}
$$

Using Eq. 5.64, we can see how the CEP of w varies based on transmission, η, and the initial squeezing parameter, r. However, we are particularly concerned with the amount of loss that can be tolerated as the expected photon number from the sources is increased, $\bar{n} = K\sinh^2(r)$. In the simplified construction that we have been considering, $K = m$, and we assume that r is chosen so that $\bar{n} = \sqrt{m}$, in which case $\sinh^2(r) = \bar{n}^{-1}$. We assume that the proportion of surviving photons is some function of the input photons like: $\eta\bar{n} = \bar{n}^{\alpha}$. This non-linear form of the relationship is not necessarily required, but here we use it as an ansatz in alignment with previous work (e.g. [259]), and to compare to previously known loss thresholds, such as $\alpha = 1/2$ [175].

We have

$$\sinh^2(r) = \frac{1}{\bar{n}},\ \cosh^2(r) = \left(1 + \frac{1}{\bar{n}}\right),\ \sinh(r)\cosh(r) = \frac{\sqrt{\bar{n}+1}}{\bar{n}},\ \eta = \bar{n}^{\alpha-1}, \tag{5.73}$$

which we use to find:

$$\begin{aligned} \rho' &= \frac{4\eta\sqrt{\bar{n}+1}\bar{n}^{-1}\left(2\eta\bar{n}^{-1}+1\right)}{4\eta^2\left(2\bar{n}^{-2}+\bar{n}^{-1}\right)+4\eta\bar{n}^{-1}+1} \\ &= \frac{4\left(\bar{n}+1\right)^{\frac{1}{2}}\bar{n}^{\alpha-2}\left(2\bar{n}^{\alpha-2}+1\right)}{4\bar{n}^{2(\alpha-1)}\left(2\bar{n}^{-2}+\bar{n}^{-1}\right)+4\bar{n}^{\alpha-2}+1} \end{aligned} \tag{5.74}$$

and

$$\begin{aligned} s' &= \frac{\eta^2\left(8\bar{n}^{-2}+4\bar{n}^{-1}\right)+4\eta\bar{n}^{-1}+1}{2\eta\left(\bar{n}+1\right)^{\frac{1}{2}}\bar{n}^{-1}}\left(1-\frac{1}{\sqrt{4\eta(1-\eta)\bar{n}^{-1}+1}}\right) \\ &= \frac{\bar{n}^{2(\alpha-1)}\left(8\bar{n}^{-2}+4\bar{n}^{-1}\right)+4\bar{n}^{\alpha-2}+1}{2\left(\bar{n}+1\right)^{\frac{1}{2}}\bar{n}^{\alpha-2}}\left(1-\frac{1}{\sqrt{4(1-\bar{n}^{\alpha-1})\bar{n}^{\alpha-2}+1}}\right). \end{aligned} \tag{5.75}$$

The CEP of w can be thought of as representing the median value of $|w|$ across many samples. It is still not clear on the growth of $|w|$ with increasing photon number that it is required in order for the efficient classical algorithms described in Section 5.4 to be applicable. However, in Fig. 5.4 compare the median $|w|$ value produced in this method, for different values of α, to the numerical results from Fig. 5.2.

As expected, a value of approximately $\alpha = 0.5$ seems to be appropriate for a simulation that is effective 75% of the time. A higher value of α corresponds to a lower loss value, hence, as expected, this gives a lower median value of $|w|$. In order for the simulation to be effective, this value should be above the $|w|$ value required to assure a sufficiently large non-zero region of the matching polynomial – i.e., this is the value of α for which the red or purple line lies above the black or green lines.

5.6.2 Loss tolerance of DGBS

Loss is a major source of error in photonic experiments, and it is worth considering the extent to which photon loss impacts the effectiveness or simulability of DGBS. Tensor network

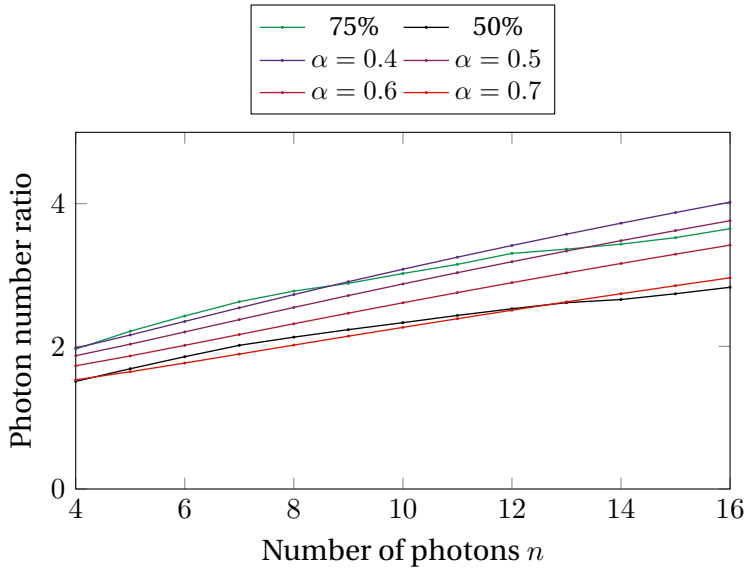


FIGURE 5.4. We show the median value of $|w|$ (representing the displacement/squeezing ratio) when the previous described method for simulating lossy GBS is used, for different values of α. We also compare this to the numerical results that show what $|w|$ value is required to have a 50% or 75% probability of our efficient simulation methods to be effective, as a lower threshold for the required values of loss (represented by α) for efficient simulation using this method.

methods have been particularly successful in exploiting photon loss to simulate large-scale GBS demonstrations [177], in which the complexity scales with the effective photon number of the 'quantum' component of the state, T. The addition of displacement, as a local operation, does not change this and only adds polynomial overheads, and therefore the loss tolerance of GBS experiments (before permitting efficient tensor network simulations) is not affected by adding displacement.

This is also the case for the phase space simulation scheme in [175]. The approximation is based on the Rényi relative entropy between two states, which upper bounds the total variation distance of measurement distributions on the states. Displacement has no effect on this, and therefore does not change the lower bound for the error of the simulation.

Nonetheless, in both of the above cases, adding displacement may increase the accuracy of classical simulation, effectively reducing the loss tolerance of quantum advantage experiments. In the case of tensor network simulations, the relative size of the 'classical' component of the state, W, (that is, the effective photon number) is increased by displacement, which increases the accuracy of the efficient part of the computation. Similarly, for phase space simulation schemes, the Rényi relative entropy is not changed by displacement, but this is only an upper bound for the TVD, and does not reflect that the accuracy of the simulation

may improve. This aspect is not taken into account when evaluating the complexity of these algorithms, which was a motivation for our study, although it should certainly be taken into account for future large-scale experiments.

Both of these simulation schemes rely on loss being above a certain level to remain efficient and accurate. Below this threshold, the method in [175] is no longer sufficiently accurate to be effective, and [177] is no longer efficient, and may not be feasible to run. Exponential, exact methods that rely on loop-hafnian calculations can be useful in this case, however the complexity then scales exponentially with the number of photons (including if generated by displacement). This may also be the case for other simulation schemes which are not currently known or used. Therefore, displacement may yet gain relevance as a method of increasing the complexity of a GBS experiment in the pursuit of quantum advantage.

5.7 Discussion

In order to understand the computational complexity of a problem, it is possible to consider arguments that the problem is 'easy', by finding efficient algorithms to solve it, or arguments that the problem is 'hard', through comparison to existing problems with more well-understood complexities. In this chapter, we have taken both approaches to consider the complexity of simulating DGBS, where the amount of displacement added determines the difficulty of simulation (and therefore which paradigm is appropriate). We have also considered how these results impact the study of zero-mean (non-displaced) GBS, via the relationship between loss and displacement.

We have introduced an efficient scheme with which to simulate DGBS in the case of high displacement. Although this applies to a limited class of matrices, it could be improved by further results regarding the matching polynomial. It also agrees with current intuition that increasing the level of displacement should make GBS easier to simulate; our numerical results show a gradual transition in the expected probability of the algorithm succeeding as the displacement to squeezing ratio increases, by studying the roots of the matching polynomial.

Our study of the complexity of DGBS has introduced a framework that, although currently relying on some conjectures and numerical results, much like traditional GBS, gives an overview of how to consider the displacement/squeezing ratio and how it affects the simulability of GBS. In particular, we see that the displacement to squeezing ratio is relevant not only in the hardness of the matching polynomial, but when considering the anti-concentration of the loop hafnian. This may present as a major restriction in the possibility of using DGBS experiments with high amounts of displacement as a foundation for quantum advantage. Our numerical results suggest a smooth transition in the hardness of simulating DGBS, which would correspond to efficient simulation schemes being effective for a higher proportion of

experimental instances, and with lower error, as the displacement is increased.

We also considered the relationship between loss and displacement. We have presented numerical results, which, due to limits on the computational power available, have only been extended to 16 photons, and a larger study would be useful. Nonetheless, as discussed, the zeros of the matching polynomial are closely related to their simulability and therefore this provides insight into the potential effectiveness of exploiting displacement for lossy GBS simulation. Our results also show that $|w|$ tends to increase even for low loss levels, and hence for $\bar{w}$ with a large magnitude, this may exceed the threshold required for simulation. Once more is known about the relationship between the displacement/squeezing ratio and simulability, this could be a useful framework to consider asymptotic efficiency of simulation, even for lower loss thresholds. We also note that efficient classical methods to simulate GBS with high amounts of loss would not necessarily work to simulate DGBS using the method presented, due to the fact that the displacement is not fixed but chosen randomly across samples.

Finally, although our suggested algorithms for the efficient classical simulation of DGBS do not appear to give any advantage over previously known methods when attempting to simulate lossy GBS, this connection could provide another useful avenue in the study of GBS simulation. If other simulation protocols are developed that allow for lower values of w to be effective, the scaling of $|w|$ seems to be similar for varying amounts of loss, including when loss rates are below currently known classical simulation thresholds. Therefore, this could become a more useful simulation method in the future, when considering quantum advantage experiments with significantly lower loss and high photon numbers.

5.8 Outlook and further work

The efficient algorithms here are just one possible method of simulating DGBS, but we hope that continued work on the matching polynomial could provide fruitful results in this area. The leading current proposals for the simulation of GBS [147, 172, 175, 177] rely on a variety of parameters (e.g. loss thresholds, or maximum feasible number of photons) in order to function effectively, and therefore it is important to continue this study to better understand where the boundary between what is possible with either quantum or classical resources will lie in the future.

Further work is needed to strengthen the complexity proof regarding DGBS. Much like with GBS, the average-case hardness of the loop hafnian, and the anti-concentration condition, are both conjectures. Numerical evidence is typically required to justify the anti-concentration condition, and, as explained in [84], new approaches beyond what is currently understood are necessary to provide an average-case/worst-case equivalence. However, the most important direction for further research into the complexity of DGBS is a better

understanding of how the hardness relies on the parameter w, representing the displacement/squeezing ratio of the experiment. Both existing analytical results, which consider the zeros of the matching polynomial and – as we have discussed – relate to the ratio of displacement to squeezing, and the numerical results presented in this chapter, give evidence for the transition of hardness as w changes. Despite this, there is a considerable difference between the low-w region (where we have higher confidence in the difficulty of simulating DGBS) and the high-w region (where our efficient simulation results hold), where we cannot give a definitive statement on the complexity, and hence the boundary between regions of high and low complexity of simulation remains poorly defined. In particular, further results will be relevant if they shed more light on the regions in which the matching polynomial is non-zero for the class of matrices we have considered, and how that relates to w.

Finally, these methods may have further implication for GBS without displacement, which is currently the more popular framework for practical realisations. This connection could be explored further, particularly in the case of comparing loss to displacement, in the same vein that recent simulation methods have been successful in exploiting loss to split GBS experiments into quantum and classical components [177]. There are several features of the data presented which could be improved in further research. Firstly, we consider the median $|w|$, hence this would only allow efficient simulation for approximately 50% of the samples. These would also be the samples which would correspond to a higher displacement - that is, it would bias the data to instead simulate an experiment with higher loss (similar to other phase space simulation methods [175]). Therefore, as expected, higher amounts of loss would provide a more effective simulation, although these are within regimes already considered by other simulated methods.

Also, the analytic decomposition used is not optimal. As described in [177], semi-definite programming can be used to optimise the decomposition to minimise the photon number of the pure state T. In this case, the efficient classical algorithms we have described may prove to be more useful than current schemes, for specific examples. We leave this as an open question for further research. Finally, if the relationship between $|w|$ and simulability becomes better understood, exploiting this to simulate lossy GBS could be a promising avenue for efficient algorithms.

The effect of displacement in GBS is significant and should not be overlooked, particularly when developing new use cases. As displacement is currently not utilised by efficient classical simulation methods, or considered in the main proofs of GBS complexity, it is an important study to further our understanding of quantum advantage. The connection between DGBS and the matching polynomial is a useful foundation with which to study this, particularly as the ongoing study of approximating the matching polynomial and its relationship to the structure of the underlying graph may give useful insights into other features of DGBS.

CHAPTER 6

Discussion and outlook

The possible use of the telephone is limited.

Engineer-in-Chief, British Post Office, 1887

Gaussian boson sampling is an elegant framework for analysing the native power of quantum optics, that has not only inspired landmark feats of experimental work, but provides us with a detailed framework to compare the capabilities of near-term quantum and classical machines. However, every proposed use of GBS that is interesting and viable outside of quantum technologies thus far has been unable to translate the exponential difficulty of simulating GBS into an exponential quantum advantage for a useful problem.

Our current understanding of the computational complexity of (G)BS relies on the description of a classical simulator which would, by construction, be more powerful than the quantum sampler. The exact class of problems solvable in polynomial time by a (G)BS device, and its relationship to other complexity classes, is still not fully understood. Importance sampling has long been understood as a potential method for improving heuristics in stochastic algorithms [203], and in certain cases, a physical device which can sample directly from the desired distribution is preferable to more complex sampling schemes. Furthermore, complexity classes defined with regards to sampling problems can be equivalently defined in terms of specifically constructed search problems [260]. Despite these existing techniques, justification for an advantage for certain problems using GBS still frequently relies on numerical evidence.

When considering the dense subgraph finding problem, the usefulness of GBS is motivated by both the difficulty of classically simulating GBS, and the potential speedup that can be gained for DkS due to the similarities in hafnians and densities. In the case of unweighted

graphs, this is no longer thought to be an exponentially difficult sampling problem for classical algorithms [150]. We have given evidence that the quantum-enhanced DkS problem is resilient to error, which may allow for efficient classical simulation methods or for simpler experimental methods to succeed, which undermines the necessity of using GBS. We have also shown that it is efficient to sample directly from the density distribution itself – this may not provide a speedup for the DkS problem, however it could be used as a basis for stochastic algorithms, much like GBS, and this further disputes that the underlying sampling problem that GBS is here being used for is in fact classically hard. In the case of complex-weighted graphs, there is more limited evidence that sampling from hafnians is useful for sampling more dense subgraphs [162]. Our results seem to confirm this, while also showing similar robustness to error. This further motivates the use of sampling directly from the density distribution, which seems to be efficient in the cases most relevant to GBS.

We have considered verification techniques in order to benchmark the performance of GBS for this problem, which seems to confirm these results, both in the resilience to error, and the increased effectiveness of sampling directly from the density, instead of hafnians, to produce high-density complex-weighted subgraphs. In general, more sophisticated techniques could be useful in order to benchmark the performance of GBS for different use cases, particularly if more become apparent in the future.

Hence, overall, our results fit into a wider pattern of waning confidence in the use of GBS for this, and related, graph problems. Furthermore, this follows the reasoning that introducing additional structure into the problem for GBS to solve also makes it vulnerable to classical simulation techniques without contradicting the original statement of the complexity proof.

For this reason, a careful complexity analysis is needed in order be able to confidently incorporate the use of displacement into demonstrations of quantum advantage, or to disregard it as a potential extension of useful tools in quantum optics. We have presented classical simulation techniques that apply in a limited range of cases, which also illustrate the link between the zeros of the matching polynomial and the simulability of GBS, which has the potential to lead to further results in the future. Importantly, this challenges the current understanding which does not consider displacement to be an important consideration in the complexity of GBS experiments, giving clear numerical evidence that experiments with a sufficiently high level of density can by simulated by efficient classical algorithms (which we have described). We have also compared the current evidence for the computational complexity of GBS with and without displacement, and highlighted areas – including the average-case hardness of calculating the loop hafnian, and the anticoncentration conjecture as a function of the relative loop weights – where additional displacement weakens the arguments for the complexity of simulation, and so further study is required to pinpoint the complexity transition. Finally, we suggest that this study might prove useful in understanding the impact of different sources of error, particularly loss. Although it does not change current

loss thresholds to allow classical simulation, this builds on previously known techniques for separating quantum and classical components of the state, and has potential to be a useful technique as our understanding of the matching polynomial is improved.

When considering further work, it is clear that more understanding is needed in the effect of spectrally impure sources of quantum interference. Previous studies of spectral impurity have focused on the effect on spontaneous sources, which results in a mixed state after the herald photon has been measured. However, further work is needed to question whether this intuition can be extended to squeezed states as input to the calculation, and in particular if this affects the simulability or usefulness of GBS schemes impacted by this source of error.

Our understanding of the different states of light, and how these relate to graph problems, could augment the study of optical computers, and how an optical advantage (if not necessary a quantum advantage) can be gained using interference effects. Finally, using the graphical formalism may prove useful in other ways to understand CV quantum resources more generally, particularly as we introduce some non-Gaussian elements, and so this study could prove fruitful even if GBS is not able to solve a hard problem on its own.

It remains an open problem whether quantum linear optics will provide a technology with real, widespread use beyond what can be achieved classically. As elaborate physical systems, which we are still learning to construct and describe accurately, they may give us insight into other quantum systems via analogue simulation, or they could be a valuable component of more complex quantum technologies. Nonetheless, studying these systems has given us many improvements in the fields of quantum optics and theoretical computer science, and is likely to continue to do so.

APPENDIX A

Notation differences

We show the notation differences that are caused by setting the vacuum covariance matrix to either $\mathbb{1}$ or $\mathbb{1}/2$. We use the latter in this work, but give both here for easy comparison to other examples in the literature.

Vacuum	$\mathbb{1}$	$\mathbb{1}/2$
σ_Q	$\sigma + \mathbb{1}$	$\sigma + \mathbb{1}/2$
A	$(X \otimes \mathbb{1})(\mathbb{1} - 2\sigma_Q^{-1})$	$(X \otimes \mathbb{1})(\mathbb{1} - \sigma_Q^{-1})$
$p(\mathbf{n})$	$\frac{2^{\lvert S\rvert} \mathrm{e}^{-\frac{1}{2}\boldsymbol{\delta}^\dagger \sigma_Q^{-1}\boldsymbol{\delta}}}{n_1!\ldots n_m!\sqrt{\lvert\sigma_Q\rvert}}\mathrm{lhaf}\left(\mathrm{filldiag}(A_{\mathbf{n}}, \gamma_{\mathbf{n}})\right)$	$\frac{\mathrm{e}^{-\frac{1}{2}\boldsymbol{\delta}^\dagger \sigma_Q^{-1}\boldsymbol{\delta}}}{n_1!\ldots n_m!\sqrt{\lvert\sigma_Q\rvert}}\mathrm{lhaf}\left(\mathrm{filldiag}(A_{\mathbf{n}}, \gamma_{\mathbf{n}})\right)$
σ	$V_{ij} = \langle\{\hat{r}_i - d_i, \hat{r}_j - d_j, \}\rangle$	$V_{ij} = \frac{1}{2}\langle\{\hat{r}_i - d_i, \hat{r}_j - d_j, \}\rangle$
uncertainty	$\sigma + i\Omega \geq 0$	$\sigma + \frac{i}{2}\Omega \geq 0$
$p_{\text{vac}}(S)$	$\det((\sigma_Q)_S/2)^{-1/2}$	$\det((\sigma_Q)_S)^{-1/2}$

APPENDIX B

Range of scaling parameter

In this section, we summarise the derivation in [202], which finds the appropriate values of $c \in \mathbb{R}$ such that

$$\sigma = (\mathbb{1} - (X \otimes \mathbb{1})A)^{-1} - \mathbb{1}/2 \tag{B.1}$$

is a valid covariance matrix, with $A = c(\mathcal{A} \oplus \mathcal{A})$ for some binary adjacency matrix $\mathcal{A}$. Note that this differs from the notation used in [1], which uses $\sigma = 2(\mathbb{1} - (X \otimes \mathbb{1})A)^{-1} - \mathbb{1}$, and also [202], which uses $A = c\mathcal{A}$. Appendix A of [202] gives further details on how a valid covariance matrix can be constructed from an arbitrary graph.

The necessary and sufficient conditions on σ for it to be a valid covariance matrix are that:

1. It is Hermitian: $\sigma^\dagger = \sigma$.

2. It is positive semidefinite – all of its eigenvalues are nonnegative.

3. It satisfies the uncertainty principle, $\sigma + i\Omega/2 \geq 0$.

The first condition is immediately satisfied due to the structure of A, as this ensures $(X \otimes \mathbb{1})A$ is symmetric, and then σ is real and symmetric. This uses the fact that the difference between two symmetric matrices is itself symmetric, and that the inverse of a symmetric matrix is symmetric.

To show the second condition, first note that $(X \otimes \mathbb{1})$ and A commute. This means that they are simultaneously diagonalisable [261]. Let the eigenvalues of A be λ_i, and the eigenvalues of $(X \otimes \mathbb{1})A$ be λ_i^X. In that case, $|\lambda_i^X| = |\lambda_i|$ as the eigenvalues of $(X \otimes \mathbb{1})$ are ± 1.

We can then rewrite Eq. B.1:

$$\begin{aligned}\sigma =&(\mathbb{1}-(X\otimes\mathbb{1})A)^{-1}-(\mathbb{1}-(X\otimes\mathbb{1})A)^{-1}(\mathbb{1}-(X\otimes\mathbb{1})A)/2\\ =&(\mathbb{1}-(X\otimes\mathbb{1})A)^{-1}(\mathbb{1}+(X\otimes\mathbb{1})A)/2.\end{aligned} \tag{B.2}$$

We can then diagonalise σ to find its eigenvalues, $\lambda_i^\sigma = \frac{1+\lambda_i^X}{1-\lambda_i^X}$. Therefore, we require $|\lambda_i|< 1\ \forall\ i$. This is satisfied by $|c|< |\lambda_{\max}|$ (and the second condition also requires that c is positive).

Finally, the third condition is satisfied by using the construction $A = c(\mathcal{A}\oplus\mathcal{A})$.

APPENDIX C

Computational complexity of linear optics

In this section we give more detail on the statements of [84].

C.1 Exact case

We first consider a classical simulator which draws from the same distribution as a boson sampler.

Theorem C.1. *Let $\mathcal{O}$ be a classical oracle that takes as input a random string r (which $\mathcal{O}$ uses as its only source of randomness) and a description of a boson computer A, and returns a sample $\mathcal{O}_A(r)$ from the probability distribution $\mathcal{D}_A$ over possible outputs of A. Then $\mathbf{P}^{\#\mathbf{P}} \subseteq \mathbf{BPP}^{\mathbf{NP}^{\mathcal{O}}}$.*

Proof. . □

By Toda's theorem. Thm. 2.1, if $\mathcal{O}$ is polynomial, then the polynomial heirarchy collapses. However, this is also true if boson sampling is simulable by an efficient algorithm with access to an oracle in $\mathbf{PH}$.

Nonetheless, this hardness result relies on formulating a sampling problem as a decision problem, which required fixing the value of r. This is not possible for a quantum computer, and hence does not provide evidence that quantum computers can solve a $\#\mathbf{P}$-hard problem.

C.2 Approximate case

APPENDIX D

Anticoncentration evidence

We provide numerical evidence to support Conjecture 5.5:

$$\Pr_{X\in\mathcal{G}_{n,n}(0,1)}\left[\left|\mathrm{lhaf}\left(XX^T, w\sum_{\mathrm{col}} X\right)\right|^2 < \frac{1}{P}\frac{n!\,F_n(w)}{2^n}\right] < \beta.$$

In order to use prior results in anticoncentration, we compare the function:

$$\sqrt{\frac{2^n}{n!\,F_n(w)}}\left|\mathrm{lhaf}\left(XX^T, w\sum_{\mathrm{col}} X\right)\right| \tag{D.1}$$

to other, similar matrix functions (on i.i.d. Gaussian matrices X):

$$\sqrt{\frac{1}{n!}}\,|\det(X)| \tag{D.2}$$

$$\sqrt{\frac{1}{n!}}\,|\mathrm{per}(X)| \tag{D.3}$$

$$\sqrt{\frac{1}{n!\binom{n-1}{n/2}}}\,\left|\mathrm{haf}\left(XX^T\right)\right|. \tag{D.4}$$

As shown in [84], the anticoncentration condition is satisfied for the determinant, that is:

$$\Pr_{X\in\mathcal{G}_{n,n}(0,1)}\left[\sqrt{\frac{1}{n!}}\,|\det(X)|^2 < \frac{1}{\mathrm{poly}(n,1/\beta)}\right] < \beta.$$

They then conjecture that the same is true for the permanent of Gaussian matrices, partly by comparison to the determinant.

In Fig. D.1, we consider the condition:

$$\Pr\left[f_n(X) < \frac{1}{P}\right] < \beta, \tag{D.5}$$

by fixing β to be 0.25 or 0.5, and finding the value of P for which Eq. D.5 holds over 100,000 randomly generated Gaussian i.i.d. matrices (X), when $f_n(X)$ is one of the matrix functions in Eqs. D.1, D.2. We also consider the loop hafnian with both $w = 0.1$ and $w = 1.0$.

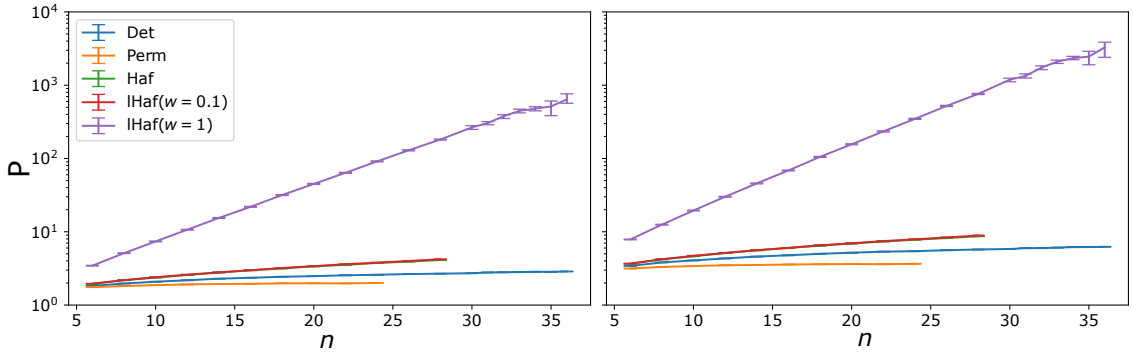


FIGURE D.1. The value of P required to fulfil Eq. 4.21 for different matrix functions. The left hand size corresponds to $\beta = 0.25$ and the right hand side is $\beta = 0.5$.

We can use this to compare the anti-concentration of the loop hafnian to other matrix functions that are more well studied. When $w = 0.1$, we can see that the loop hafnian behaves similarly to the hafnian (which is the equivalent for $w = 0$). It seems as though fixed values of w show similar behaviour.

The question remains on the variation of w with n that is required in order for DGBS to be simulable. If it is required that w increases with n, then the increasing gradient of the anti-concentration condition, as shown in Fig. D.1, would show a different behaviour to the other matrix functions, that have more currently-known evidence of anti-concentration. Therefore we suggest that a constant value of w may be required to ensure hardness using the complexity proof presented here - although we note that there are other proofs that do not rely on the anti-concentration conjecture.

In Fig. D.2, we plot the probability of Eq. D.5 for different matrix functions, again by estimating over 100,000 randomly chosen matrices X, but this time showing the dependence of the probability (i.e. the minimum value of β) on P.

As before, the loop hafnians show a qualitatively similarly shaped curve for these results as the other matrix functions, which supports that the anti-concentration condition holds for fixed values of w, although with higher values of P.

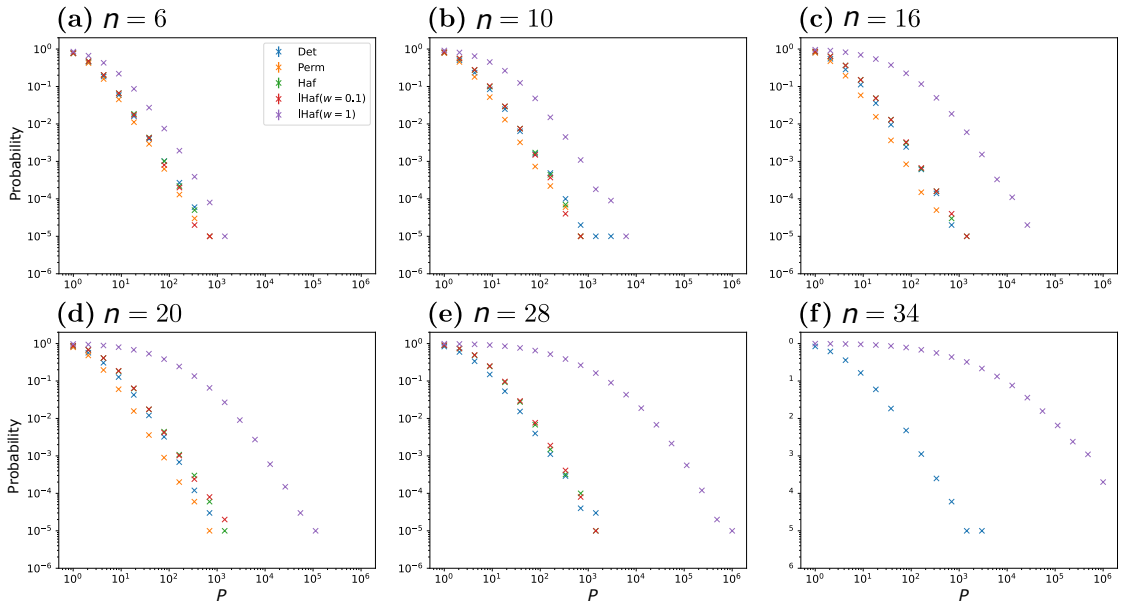


FIGURE D.2. The probability that a matrix function is below $1/P$ for different values of P and different matrix sizes.

Bibliography

[1] N. R. Solomons, O. F. Thomas, and D. P. S. McCutcheon, "Effect of photonic errors on quantum enhanced dense-subgraph finding," *Physical Review Applied*, vol. 20, no. 5, p. 054043, 2023.

[2] "'computer', Oxford English Dictionary," 2023.

[3] Centre for Computing History, "Richard Braithwaite first uses the word 'computer'." `https://www.computinghistory.org.uk/det/5829/Richard-Braithwaite-first-uses-the-word-computer/`. Accessed: 2023-08-07.

[4] R. B. Gent, Richard Brathwaite (attributed name), and Saint Bernard of Clairvaux, "The yong mans gleanings," 1614. Oxford Text Archive.

[5] P. E. Ceruzzi, "When computers were human," *Annals of the History of Computing*, vol. 13, no. 3, pp. 237–244, 1991.

[6] ENIAC programmers project, "ENIAC programmers memorials." `https://eniacprogrammers.org/eniac-programmers-project/memorials/`. Accessed: 2023-11-02.

[7] Editors of Encyclopaedia Britannica, "Abacus." `https://www.britannica.com/technology/abacus-calculating-device`, 2023.

[8] Editors of Encyclopaedia Britannica, "Slide rule." `https://www.britannica.com/science/slide-rule`, 2023.

[9] K. Efstathiou and M. Efstathiou, "Celestial gearbox," *Mechanical Engineering*, vol. 140, no. 09, pp. 31–35, 2018.

[10] J. E. Savage, *Models of computation*, vol. 136. Addison-Wesley Reading, MA, 1998.

[11] O. Bournez and A. Pouly, "A survey on analog models of computation," in *Handbook of Computability and Complexity in Analysis*, pp. 173–226, Springer, 2021.

[12] B. J. MacLennan, "A review of analog computing," *Department of Electrical Engineering & Computer Science, University of Tennessee, Technical Report UT-CS-07-601 (September)*, 2007.

[13] C. Petzold, *Code: The hidden language of computer hardware and software*. Microsoft Press, Upper Saddle River, NJ, 2000.

[14] A. M. Turing *et al.*, "On computable numbers, with an application to the entscheidungsproblem," *J. of Math*, vol. 58, no. 345-363, p. 5, 1936.

[15] B. Robič, *The foundations of computability theory*. Springer Heidelberg, Berlin, 2015.

[16] G. Tourlakis, *Computability*. Springer Nature, Switzerland, 2022.

[17] A. Church, "An unsolvable problem of elementary number theory," *American Journal of Mathematics*, vol. 58, pp. 345–363, 1936.

[18] S. Arora and B. Barak, *Computational complexity: a modern approach*. Cambridge University Press, 2009.

[19] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to algorithms*. MIT press, Cambridge, MA, 2022.

[20] A. C.-C. Yao, "Classical physics and the Church–Turing thesis," *Journal of the ACM (JACM)*, vol. 50, no. 1, pp. 100–105, 2003.

[21] N. Dershowitz and Y. Gurevich, "A natural axiomatization of computability and proof of church's thesis," *Bulletin of symbolic logic*, vol. 14, no. 3, pp. 299–350, 2008.

[22] A. Sterling, "A mathematical proof of the Church-Turing Thesis?." `https://nanoexplanations.wordpress.com/2011/07/04/a-mathematical-proof-of-the-church-turing-thesis/`. Accessed: 2025-06-19.

[23] S. Aaronson, G. Kuperberg, and C. Granade, "The complexity zoo." `https://complexityzoo.net/`.

[24] M. Agrawal, N. Kayal, and N. Saxena, "Primes is in p," *Annals of Mathematics*, vol. 160, pp. 781–793, 2004.

[25] C. C. Heckman, "The 2×2 matrix mortality problem and invertible matrices," *arXiv preprint arXiv:1912.09991*, 2019.

[26] W. I. Gasarch, "Guest Column: The Third P=?NP Poll," *SIGACT News*, vol. 50, p. 38–59, mar 2019.

[27] S. Aaronson, *P=?NP*, pp. 1–122. Cham: Springer International Publishing, 2016.

[28] S. A. Cook, "The complexity of theorem-proving procedures," *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, pp. 151–158, 1971.

[29] L. A. Levin, "Universal search problems," *Annals of the History of Computing*, vol. 6, no. 4, pp. 399–400, 1973. Translation from 1984 by B. A. Trakhtenbrot.

[30] R. M. Karp, *Reducibility among combinatorial problems*. Springer, Boston, MA, 1972.

[31] S. A. Fenner, L. J. Fortnow, and S. A. Kurtz, "Gap-definable counting classes," *Journal of Computer and System Sciences*, vol. 48, no. 1, pp. 116–148, 1994.

[32] R. O'Donnell and A. C. Say, "The weakness of ctc qubits and the power of approximate counting," *ACM Transactions on Computation Theory (TOCT)*, vol. 10, no. 2, pp. 1–22, 2018.

[33] S. Bravyi, D. P. Divincenzo, R. I. Oliveira, and B. M. Terhal, "The complexity of stoquastic local hamiltonian problems," *arXiv preprint quant-ph/0606140*, 2006.

[34] S. Aaronson, "Quantum computing, postselection, and probabilistic polynomial-time," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 461, no. 2063, pp. 3473–3482, 2005.

[35] V. Kreinovich, "Theory of computation lecture notes," 2009.

[36] J. van de Wetering and M. Amy, "Optimising T-count is NP-hard," *arXiv preprint arXiv:2310.05958*, 2023.

[37] S. Toda, "PP is as hard as the polynomial-time hierarchy," *SIAM Journal on Computing*, vol. 20, no. 5, pp. 865–877, 1991.

[38] B. H. Bransden and C. J. Joachain, *Introduction to quantum mechanics*. Longman Scientific & Technical, Harlow, 1989.

[39] A. Streltsov, U. Singh, H. S. Dhar, M. N. Bera, and G. Adesso, "Measuring quantum coherence with entanglement," *Physical review letters*, vol. 115, no. 2, p. 020403, 2015.

[40] P. Blood, "The classical atomic dipole oscillator," in *Quantum Confined Laser Devices: Optical gain and recombination in semiconductors*, Oxford University Press, 10 2015.

[41] R. L. Jaffe, "Quantum physics lecture notes," 1996.

[42] L. Masanes, T. D. Galley, and M. P. Müller, "The measurement postulates of quantum mechanics are operationally redundant," *Nature communications*, vol. 10, no. 1, p. 1361, 2019.

[43] S. Saunders, "Derivation of the born rule from operational assumptions," *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 460, no. 2046, pp. 1771–1788, 2004.

[44] S. Aaronson, "Is quantum mechanics an island in theoryspace?," *arXiv preprint quant-ph/0401062*, 2004.

[45] D. Howard, "Who invented the "Copenhagen interpretation"? a study in mythology," *Philosophy of Science*, vol. 71, no. 5, pp. 669–682, 2004.

[46] L. Vaidman, "Why the many-worlds interpretation?," *Quantum Reports*, vol. 4, no. 3, pp. 264–271, 2022.

[47] R. Tumulka, "Bohmian mechanics," *arXiv preprint arXiv:1704.08017*, 2021.

[48] A. Einstein, B. Podolsky, and N. Rosen, "Can quantum-mechanical description of physical reality be considered complete?," *Physical review*, vol. 47, no. 10, p. 777, 1935.

[49] M. Genovese, "Research on hidden variable theories: A review of recent progresses," *Physics Reports*, vol. 413, no. 6, pp. 319–396, 2005.

[50] B. Hensen, H. Bernien, A. E. Dréau, A. Reiserer, N. Kalb, M. S. Blok, J. Ruitenberg, R. F. Vermeulen, R. N. Schouten, C. Abellán, *et al.*, "Loophole-free Bell inequality violation using electron spins separated by 1.3 kilometres," *Nature*, vol. 526, no. 7575, pp. 682–686, 2015.

[51] S. Aaronson, "Introduction to quantum information science lecture notes," 2018.

[52] M. A. Nielsen and I. L. Chuang, *Quantum computation and quantum information*. Cambridge university press, 2010.

[53] R. I. Wakefield, *Creating and Simulating Parity-Time Symmetric Systems using Nonlinear Quantum Optics*. PhD thesis, University of Bristol, 2022.

[54] F. Wilczek, "Magnetic flux, angular momentum, and statistics," *Physical Review Letters*, vol. 48, no. 17, p. 1144, 1982.

[55] J. S. Sidhu, S. K. Joshi, M. Gündoğan, T. Brougham, D. Lowndes, L. Mazzarella, M. Krutzik, S. Mohapatra, D. Dequal, G. Vallone, *et al.*, "Advances in space quantum communications," *IET Quantum Communication*, vol. 2, no. 4, pp. 182–217, 2021.

[56] N. Gisin and R. Thew, "Quantum communication," *Nature photonics*, vol. 1, no. 3, pp. 165–171, 2007.

[57] G. Tóth and I. Apellaniz, "Quantum metrology from a quantum information science perspective," *Journal of Physics A: Mathematical and Theoretical*, vol. 47, no. 42, p. 424006, 2014.

[58] A. Y. Kitaev, "Quantum computations: algorithms and error correction," *Russian Mathematical Surveys*, vol. 52, no. 6, p. 1191, 1997.

[59] S. Forest, D. Gosset, V. Kliuchnikov, and D. McKinnon, "Exact synthesis of single-qubit unitaries over clifford-cyclotomic gate sets," *Journal of Mathematical Physics*, vol. 56, no. 8, 2015.

[60] S. Aaronson and D. Gottesman, "Improved simulation of stabilizer circuits," *Physical Review A*, vol. 70, no. 5, p. 052328, 2004.

[61] D. Deutsch, "Quantum theory, the church–turing principle and the universal quantum computer," *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, vol. 400, no. 1818, pp. 97–117, 1985.

[62] D. Deutsch and R. Jozsa, "Rapid solution of problems by quantum computation," *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, vol. 439, no. 1907, pp. 553–558, 1992.

[63] D. R. Simon, "On the power of quantum computation," *SIAM journal on computing*, vol. 26, no. 5, pp. 1474–1483, 1997.

[64] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM review*, vol. 41, no. 2, pp. 303–332, 1999.

[65] N.-H. Chia, A. P. Gilyén, T. Li, H.-H. Lin, E. Tang, and C. Wang, "Sampling-based sublinear low-rank matrix arithmetic framework for dequantizing quantum machine learning," *Journal of the ACM*, vol. 69, no. 5, pp. 1–72, 2022.

[66] R. P. Feynman *et al.*, "Simulating physics with computers," *Int. j. Theor. phys*, vol. 21, no. 6/7, 2018.

[67] C. Sparrow, E. Martín-López, N. Maraviglia, A. Neville, C. Harrold, J. Carolan, Y. N. Joglekar, T. Hashimoto, N. Matsuda, J. L. O'Brien, *et al.*, "Simulating the vibrational quantum dynamics of molecules using photonics," *Nature*, vol. 557, no. 7707, pp. 660–667, 2018.

[68] T. H. Johnson, S. R. Clark, and D. Jaksch, "What is a quantum simulator?," *EPJ Quantum Technology*, vol. 1, no. 1, pp. 1–12, 2014.

[69] G. H. Low, Y. Su, Y. Tong, and M. C. Tran, "Complexity of implementing trotter steps," *PRX Quantum*, vol. 4, no. 2, p. 020323, 2023.

[70] E. Knill, R. Laflamme, and W. H. Zurek, "Resilient quantum computation," *Science*, vol. 279, no. 5349, pp. 342–345, 1998.

[71] J. Preskill, "Quantum computing in the NISQ era and beyond," *Quantum*, vol. 2, p. 79, 2018.

[72] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, *et al.*, "Noisy intermediate-scale quantum algorithms," *Reviews of Modern Physics*, vol. 94, no. 1, p. 015004, 2022.

[73] J. Preskill, "Quantum computing and the entanglement frontier," *arXiv preprint arXiv:1203.5813*, 2012.

[74] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, *et al.*, "Quantum supremacy using a programmable superconducting processor," *Nature*, vol. 574, no. 7779, pp. 505–510, 2019.

[75] F. Pan, K. Chen, and P. Zhang, "Solving the sampling problem of the sycamore quantum circuits," *Physical Review Letters*, vol. 129, no. 9, p. 090502, 2022.

[76] T. Hoefler, T. Häner, and M. Troyer, "Disentangling hype from practicality: on realistically achieving quantum advantage," *Communications of the ACM*, vol. 66, no. 5, pp. 82–87, 2023.

[77] Q. E. Initiative, "Welcome." `https://quantum-energy-initiative.org/`. Accessed: 2024-01-30.

[78] D. Hangleiter and J. Eisert, "Computational advantage of quantum random sampling," *Reviews of Modern Physics*, vol. 95, no. 3, p. 035001, 2023.

[79] A. P. Lund, M. J. Bremner, and T. C. Ralph, "Quantum sampling problems, boson-sampling and quantum supremacy," *npj Quantum Information*, vol. 3, no. 1, p. 15, 2017.

[80] D. Shepherd and M. J. Bremner, "Temporally unstructured quantum computation," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 465, no. 2105, pp. 1413–1439, 2009.

[81] R. Chadwick, *Classical Simulations of Gaussian Boson Sampling*. PhD thesis, University of Bristol, 2022.

[82] C. Robens, I. Arrazola, W. Alt, D. Meschede, L. Lamata, E. Solano, and A. Alberti, "Boson sampling with ultracold atoms," *arXiv preprint arXiv:2208.12253*, 2022.

[83] A. W. Young, S. Geller, W. J. Eckner, N. Schine, S. Glancy, E. Knill, and A. M. Kaufman, "An atomic boson sampler," *arXiv preprint arXiv:2307.06936*, 2023.

[84] S. Aaronson and A. Arkhipov, "The computational complexity of linear optics," in *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pp. 333–342, 2011.

[85] C. D. Bruzewicz, J. Chiaverini, R. McConnell, and J. M. Sage, "Trapped-ion quantum computing: Progress and challenges," *Applied Physics Reviews*, vol. 6, no. 2, 2019.

[86] H.-L. Huang, D. Wu, D. Fan, and X. Zhu, "Superconducting quantum computing: a review," *Science China Information Sciences*, vol. 63, pp. 1–32, 2020.

[87] I. Georgescu, "The DiVincenzo criteria 20 years on," *Nature Reviews Physics*, vol. 2, no. 12, pp. 666–666, 2020.

[88] D. P. Hampshire, "A derivation of Maxwell's equations using the Heaviside notation," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 376, no. 2134, p. 20170447, 2018.

[89] R. Loudon, *The quantum theory of light*. OUP Oxford, 2000.

[90] D. Walls and G. Milburn, *Quantum Optics*. Springer Berlin Heidelberg, 2008.

[91] P. W. Yard, *Time as a resource in integrated quantum photonics*. PhD thesis, University of Bristol, 2021.

[92] W. E. Lamb, "Anti-photon," *Applied Physics B*, vol. 60, pp. 77–84, 1995.

[93] C. Fabre and N. Treps, "Modes and states in quantum optics," *Reviews of Modern Physics*, vol. 92, no. 3, p. 035005, 2020.

[94] S. Paesani, *Large-scale integrated quantum photonics: Development and applications*. PhD thesis, University of Bristol, 2019.

[95] W. Clements, *Linear quantum optics: Components and applications*. PhD thesis, University of Oxford, 2018.

[96] P. Kok and B. W. Lovett, *Introduction to optical quantum information processing*. Cambridge university press, 2010.

[97] M. Gimeno-Segovia, *Towards practical linear optical quantum computing*. PhD thesis, Imperial College London, 2015.

[98] C.-K. Hong, Z.-Y. Ou, and L. Mandel, "Measurement of subpicosecond time intervals between two photons by interference," *Physical review letters*, vol. 59, no. 18, p. 2044, 1987.

[99] W. Feller, *An introduction to probability theory and its applications, Volume 2*, vol. 81. John Wiley & Sons, New York, 1991.

[100] M. Karácsony, L. Oroszlány, and Z. Zimborás, "Efficient qudit based scheme for photonic quantum computing," *arXiv preprint arXiv:2302.07357*, 2023.

[101] M. Reck, A. Zeilinger, H. J. Bernstein, and P. Bertani, "Experimental realization of any discrete unitary operator," *Physical review letters*, vol. 73, no. 1, p. 58, 1994.

[102] W. R. Clements, P. C. Humphreys, B. J. Metcalf, W. S. Kolthammer, and I. A. Walmsley, "Optimal design for universal multiport interferometers," *Optica*, vol. 3, no. 12, pp. 1460–1465, 2016.

[103] S. Stanisic and P. S. Turner, "Discriminating distinguishability," *Physical Review A*, vol. 98, no. 4, p. 043839, 2018.

[104] R. D. Shaw, A. E. Jones, P. Yard, and A. Laing, "Errors in heralded circuits for linear optical entanglement generation," *arXiv preprint arXiv:2305.08452*, 2023.

[105] D. E. Browne and T. Rudolph, "Resource-efficient linear optical quantum computation," *Physical Review Letters*, vol. 95, no. 1, p. 010501, 2005.

[106] S. Stanisic, "Universal quantum computation by linear optics," 2015.

[107] E. Knill, R. Laflamme, and G. J. Milburn, "A scheme for efficient quantum computation with linear optics," *nature*, vol. 409, no. 6816, pp. 46–52, 2001.

[108] R. Raussendorf and H. J. Briegel, "A one-way quantum computer," *Physical review letters*, vol. 86, no. 22, p. 5188, 2001.

[109] T. Rudolph, "Why I am optimistic about the silicon-photonic route to quantum computing," *APL photonics*, vol. 2, no. 3, 2017.

[110] S. Popescu, "Klm quantum computation as a measurement based computation," *arXiv preprint quant-ph/0610025*, 2006.

[111] S. Bartolucci, P. Birchall, H. Bombin, H. Cable, C. Dawson, M. Gimeno-Segovia, E. Johnston, K. Kieling, N. Nickerson, M. Pant, *et al.*, "Fusion-based quantum computation," *Nature Communications*, vol. 14, no. 1, p. 912, 2023.

[112] J. W. Essam and M. E. Fisher, "Some Basic Definitions in Graph Theory," *Rev. Mod. Phys.*, vol. 42, pp. 271–288, Apr 1970.

[113] F. Harary, *Graph theory*. Wiley New York, 1969.

[114] T. F. Coleman and J. J. Moré, "Estimation of sparse Jacobian matrices and graph coloring blems," *SIAM journal on Numerical Analysis*, vol. 20, no. 1, pp. 187–209, 1983.

[115] A. Barvinok, *Combinatorics and complexity of partition functions*, vol. 30. Springer Cham, 2016.

[116] N. Quesada, "The hafnian." `https://the-walrus.readthedocs.io/en/latest/hafnian.html`. Accessed: 2024-01-16.

[117] D. E. Littlewood and A. R. Richardson, "Group characters and algebra," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol. 233, no. 721-730, pp. 99–141, 1934.

[118] L. G. Valiant, "The complexity of computing the permanent," *Theoretical computer science*, vol. 8, no. 2, pp. 189–201, 1979.

[119] J. Edmonds, "Paths, trees, and flowers," *Canadian Journal of mathematics*, vol. 17, pp. 449–467, 1965.

[120] S. (https://math.stackexchange.com/users/180683/secret), "What makes the permanent lot more difficult than the determinant." Mathematics Stack Exchange. URL:https://math.stackexchange.com/q/1369961 (version: 2015-07-22).

[121] J. R. Scott and K. C. Balram, "Timing constraints imposed by classical digital control systems on photonic implementations of measurement-based quantum computing," *IEEE Transactions on Quantum Engineering*, vol. 3, pp. 1–20, 2022.

[122] I. Bloch, J. Dalibard, and W. Zwerger, "Many-body physics with ultracold gases," *Reviews of modern physics*, vol. 80, no. 3, p. 885, 2008.

[123] S. Scheel, "Permanents in linear optical networks," *arXiv preprint quant-ph/0406127*, 2004.

[124] L. Stockmeyer, "The complexity of approximate counting," in *Proceedings of the fifteenth annual ACM symposium on Theory of computing*, pp. 118–126, 1983.

[125] K. Zyczkowski and M. Kus, "Random unitary matrices," *Journal of Physics A: Mathematical and General*, vol. 27, no. 12, p. 4235, 1994.

[126] A. Bouland, D. Brod, I. Datta, B. Fefferman, D. Grier, F. Hernandez, and M. Oszmaniec, "Complexity-theoretic foundations of BosonSampling with a linear number of modes," *arXiv preprint arXiv:2312.00286*, 2023.

[127] M. Rudelson and O. Zeitouni, "Singular values of gaussian matrices and permanent estimators," *Random Structures & Algorithms*, vol. 48, no. 1, pp. 183–212, 2016.

[128] K. P. Costello and V. Vu, "Concentration of random determinants and permanent estimators," *SIAM Journal on Discrete Mathematics*, vol. 23, no. 3, pp. 1356–1371, 2009.

[129] D. Hangleiter, J. Bermejo-Vega, M. Schwarz, and J. Eisert, "Anticoncentration theorems for schemes showing a quantum speedup," *Quantum*, vol. 2, p. 65, 2018.

[130] L. A. Ngah, O. Alibart, L. Labonté, V. d'Auria, and S. Tanzilli, "Ultra-fast heralded single photon source based on telecom technology," *Laser & Photonics Reviews*, vol. 9, no. 2, pp. L1–L5, 2015.

[131] G. Adesso, S. Ragy, and A. R. Lee, "Continuous variable quantum information: Gaussian states and beyond," *Open Systems & Information Dynamics*, vol. 21, no. 01n02, p. 1440001, 2014.

[132] A. Serafini, *Quantum continuous variables: a primer of theoretical methods*. CRC press, Boca Raton, 2017.

[133] J. F. Bulmer, *Quantum photonics for computation and computation of quantum photonics*. PhD thesis, University of Bristol, 2022.

[134] D. P. S. McCutcheon, "Quantum optics notes," 2020.

[135] F. Rioux, "Another look at the Wigner function (lecture notes),"

[136] A. Mari and J. Eisert, "Positive wigner functions render classical simulation of quantum computation efficient," *Physical review letters*, vol. 109, no. 23, p. 230503, 2012.

[137] S. D. Bartlett and B. C. Sanders, "Efficient classical simulation of optical quantum information circuits," *Physical review letters*, vol. 89, no. 20, p. 207903, 2002.

[138] V. Sivak, A. Eickbusch, B. Royer, S. Singh, I. Tsioutsios, S. Ganjam, A. Miano, B. Brock, A. Ding, L. Frunzio, *et al.*, "Real-time quantum error correction beyond break-even," *Nature*, vol. 616, no. 7955, pp. 50–55, 2023.

[139] I. Tzitrin, J. E. Bourassa, N. C. Menicucci, and K. K. Sabapathy, "Progress towards practical qubit computation using approximate Gottesman-Kitaev-Preskill codes," *Physical Review A*, vol. 101, no. 3, p. 032315, 2020.

[140] D. Gottesman, A. Kitaev, and J. Preskill, "Encoding a qubit in an oscillator," *Physical Review A*, vol. 64, no. 1, p. 012310, 2001.

[141] J. W. Silverstone, D. Bonneau, J. L. O'Brien, and M. G. Thompson, "Silicon quantum photonics," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 22, no. 6, pp. 390–402, 2016.

[142] A. P. Lund, A. Laing, S. Rahimi-Keshari, T. Rudolph, J. L. O'Brien, and T. C. Ralph, "Boson sampling from a gaussian state," *Physical review letters*, vol. 113, no. 10, p. 100502, 2014.

[143] M. Bentivegna, N. Spagnolo, C. Vitelli, F. Flamini, N. Viggianiello, L. Latmiral, P. Mataloni, D. J. Brod, E. F. Galvão, A. Crespi, *et al.*, "Experimental scattershot boson sampling," *Science advances*, vol. 1, no. 3, p. e1400255, 2015.

[144] C. S. Hamilton, R. Kruse, L. Sansoni, S. Barkhofen, C. Silberhorn, and I. Jex, "Gaussian boson sampling," *Physical review letters*, vol. 119, no. 17, p. 170501, 2017.

[145] R. Kruse, C. S. Hamilton, L. Sansoni, S. Barkhofen, C. Silberhorn, and I. Jex, "Detailed study of gaussian boson sampling," *Physical Review A*, vol. 100, no. 3, p. 032326, 2019.

[146] N. Quesada, "Franck-Condon factors by counting perfect matchings of graphs with loops," *The Journal of chemical physics*, vol. 150, no. 16, p. 164113, 2019.

[147] G. Thekkadath, S. Sempere-Llagostera, B. Bell, R. Patel, M. Kim, and I. Walmsley, "Experimental demonstration of Gaussian boson sampling with displacement," *arXiv preprint arXiv:2202.00634*, 2022.

[148] N. Quesada, R. S. Chadwick, B. A. Bell, J. M. Arrazola, T. Vincent, H. Qi, R. García, *et al.*, "Quadratic speed-up for simulating gaussian boson sampling," *PRX Quantum*, vol. 3, no. 1, p. 010306, 2022.

[149] M. Hardy, "Combinatorics of partial derivatives," *arXiv preprint math/0601149*, 2006.

[150] C. Oh, L. Jiang, and N. Quesada, "Quantum-inspired classical algorithm for graph problems by Gaussian boson sampling," *arXiv preprint arXiv:2302.00536*, 2023.

[151] N. Quesada, J. M. Arrazola, and N. Killoran, "Gaussian boson sampling using threshold detectors," *Physical Review A*, vol. 98, no. 6, p. 062322, 2018.

[152] N. C. Menicucci, S. T. Flammia, and P. van Loock, "Graphical calculus for Gaussian pure states," *Physical Review A*, vol. 83, no. 4, p. 042335, 2011.

[153] O. J. Heilmann and E. H. Lieb, "Theory of monomer-dimer systems," in *Statistical Mechanics*, pp. 45–87, Springer, 1972.

[154] M. Sureka and S. Guha, "Gaussian Boson Sampling to Accelerate NP-Complete Vertex-Minor Graph Classification," *arXiv preprint arXiv:2402.03524*, 2024.

[155] T. R. Bromley, J. M. Arrazola, S. Jahangiri, J. Izaac, N. Quesada, A. D. Gran, M. Schuld, J. Swinarton, Z. Zabaneh, and N. Killoran, "Applications of near-term photonic quantum computers: software and algorithms," *Quantum Science and Technology*, vol. 5, no. 3, p. 034010, 2020.

[156] C. Oh, Y. Lim, Y. Wong, B. Fefferman, and L. Jiang, "Quantum-inspired classical algorithm for molecular vibronic spectra," *arXiv preprint arXiv:2202.01861*, 2022.

[157] J. E. Bourassa, R. N. Alexander, M. Vasmer, A. Patil, I. Tzitrin, T. Matsuura, D. Su, B. Q. Baragiola, S. Guha, G. Dauphinais, *et al.*, "Blueprint for a scalable photonic fault-tolerant quantum computer," *Quantum*, vol. 5, p. 392, 2021.

[158] D. Grier, D. J. Brod, J. M. Arrazola, M. B. de Andrade Alonso, and N. Quesada, "The complexity of bipartite Gaussian boson sampling," *Quantum*, vol. 6, p. 863, 2022.

[159] A. Deshpande, A. Mehta, T. Vincent, N. Quesada, M. Hinsche, M. Ioannou, L. Madsen, J. Lavoie, H. Qi, J. Eisert, *et al.*, "Quantum computational advantage via high-dimensional gaussian boson sampling," *Science advances*, vol. 8, no. 1, p. eabi7894, 2022.

[160] L. S. Madsen, F. Laudenbach, M. F. Askarani, F. Rortais, T. Vincent, J. F. Bulmer, F. M. Miatto, L. Neuhaus, L. G. Helt, M. J. Collins, *et al.*, "Quantum computational advantage with a programmable photonic processor," *Nature*, vol. 606, no. 7912, pp. 75–81, 2022.

[161] H.-S. Zhong, H. Wang, Y.-H. Deng, M.-C. Chen, L.-C. Peng, Y.-H. Luo, J. Qin, D. Wu, X. Ding, Y. Hu, *et al.*, "Quantum computational advantage using photons," *Science*, vol. 370, no. 6523, pp. 1460–1463, 2020.

[162] Y.-H. Deng, S.-Q. Gong, Y.-C. Gu, Z.-J. Zhang, H.-L. Liu, H. Su, H.-Y. Tang, J.-M. Xu, M.-H. Jia, M.-C. Chen, *et al.*, "Solving graph problems using Gaussian boson sampling," *Physical Review Letters*, vol. 130, no. 19, p. 190601, 2023.

[163] A. Björklund, B. Gupt, and N. Quesada, "A faster hafnian formula for complex matrices and its benchmarking on a supercomputer," *Journal of Experimental Algorithmics (JEA)*, vol. 24, pp. 1–17, 2019.

[164] H. J. Ryser, *Combinatorial mathematics*, vol. 14. American Mathematical Soc., Providence, Rhode Island, 1963.

[165] G. Rempala and J. Wesolowski, *Symmetric functionals on random matrices and random matchings problems*, vol. 147. Springer Science & Business Media, Louisville, KY, 2007.

[166] C. Oh, Y. Lim, B. Fefferman, and L. Jiang, "Classical simulation of boson sampling based on graph structure," *Physical Review Letters*, vol. 128, no. 19, p. 190501, 2022.

[167] M. Jerrum and A. Sinclair, "Approximating the permanent," *SIAM journal on computing*, vol. 18, no. 6, pp. 1149–1178, 1989.

[168] A. Barvinok, "Polynomial time algorithms to approximate permanents and mixed discriminants within a simply exponential factor," *Random Structures & Algorithms*, vol. 14, no. 1, pp. 29–61, 1999.

[169] M. Rudelson, A. Samorodnitsky, and O. Zeitouni, "Hafnians, perfect matchings and Gaussian matrices," *Annals of probability: An official journal of the Institute of Mathematical Statistics*, vol. 44, no. 4, pp. 2858–2888, 2016.

[170] "The Walrus – the algorithms." `https://the-walrus.readthedocs.io/en/latest/algorithms.html`. Accessed: 2023-11-01.

[171] P. Clifford and R. Clifford, "The classical complexity of boson sampling," in *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 146–155, SIAM, 2018.

[172] J. F. Bulmer, B. A. Bell, R. S. Chadwick, A. E. Jones, D. Moise, A. Rigazzi, J. Thorbecke, U.-U. Haus, T. Van Vaerenbergh, R. B. Patel, *et al.*, "The boundary for quantum advantage in Gaussian boson sampling," *Science advances*, vol. 8, no. 4, p. eabl9236, 2022.

[173] J. F. Bulmer, S. Paesani, R. S. Chadwick, and N. Quesada, "Threshold detection statistics of bosonic states," *Physical Review A*, vol. 106, no. 4, p. 043712, 2022.

[174] S. Rahimi-Keshari, A. P. Lund, and T. C. Ralph, "What can quantum optics say about computational complexity theory?," *Physical review letters*, vol. 114, no. 6, p. 060501, 2015.

[175] H. Qi, D. J. Brod, N. Quesada, and R. García-Patrón, "Regimes of classical simulability for noisy Gaussian boson sampling," *Physical review letters*, vol. 124, no. 10, p. 100502, 2020.

[176] J. Martínez-Cifuentes, K. Fonseca-Romero, and N. Quesada, "Classical models may be a better explanation of the Jiuzhang 1.0 Gaussian Boson Sampler than its targeted squeezed light model," *Quantum*, vol. 7, p. 1076, 2023.

[177] C. Oh, M. Liu, Y. Alexeev, B. Fefferman, and L. Jiang, "Tensor network algorithm for simulating experimental Gaussian boson sampling," *arXiv preprint arXiv:2306.03709*, 2023.

[178] O. F. Thomas, W. McCutcheon, and D. P. McCutcheon, "A general framework for multimode Gaussian quantum optics and photo-detection: Application to Hong–Ou–Mandel interference with filtered heralded single photon sources," *APL Photonics*, vol. 6, no. 4, 2021.

[179] J. M. Arrazola and T. R. Bromley, "Using Gaussian boson sampling to find dense subgraphs," *Physical review letters*, vol. 121, no. 3, p. 030503, 2018.

[180] S. Sempere-Llagostera, R. Patel, I. Walmsley, and W. Kolthammer, "Experimentally finding dense subgraphs using a time-bin encoded Gaussian boson sampling device," *Physical Review X*, vol. 12, no. 3, p. 031045, 2022.

[181] U. Feige, D. Peleg, and G. Kortsarz, "The dense k-subgraph problem," *Algorithmica*, vol. 29, no. 3, pp. 410–421, 2001.

[182] T. Lanciano, A. Miyauchi, A. Fazzone, and F. Bonchi, "A survey on the densest subgraph problem and its variants," *arXiv preprint arXiv:2303.14467*, 2023.

[183] R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins, "Trawling the web for emerging cyber-communities," *Computer networks*, vol. 31, no. 11-16, pp. 1481–1493, 1999.

[184] J. Chen and Y. Saad, "Dense subgraph extraction with application to community detection," *IEEE Transactions on knowledge and data engineering*, vol. 24, no. 7, pp. 1216–1230, 2010.

[185] A. Angel, N. Koudas, N. Sarkas, D. Srivastava, M. Svendsen, and S. Tirthapura, "Dense subgraph maintenance under streaming edge weight updates for real-time story identification," *The VLDB journal*, vol. 23, no. 2, pp. 175–199, 2014.

[186] A. Beutel, W. Xu, V. Guruswami, C. Palow, and C. Faloutsos, "Copycatch: stopping group attacks by spotting lockstep behavior in social networks," in *Proceedings of the 22nd international conference on World Wide Web*, pp. 119–130, 2013.

[187] S. Arora, B. Barak, M. Brunnermeier, and R. Ge, "Computational complexity and information asymmetry in financial products," in *ICS*, pp. 49–65, 2010.

[188] H. Hu, X. Yan, Y. Huang, J. Han, and X. J. Zhou, "Mining coherent dense subgraphs across massive biological networks for functional discovery," *Bioinformatics*, vol. 21, pp. 213–221, 2005.

[189] F. S. Kuhl, G. M. Crippen, and D. K. Friesen, "A combinatorial algorithm for calculating ligand binding," *Journal of Computational Chemistry*, vol. 5, no. 1, pp. 24–34, 1984.

[190] L. Banchi, M. Fingerhuth, T. Babej, C. Ing, and J. M. Arrazola, "Molecular docking with Gaussian boson sampling," *Science advances*, vol. 6, no. 23, p. eaax1950, 2020.

[191] M. Tang, K. Hwang, and S. H. Kang, "StemP: A fast and deterministic Stem-graph approach for RNA and protein folding prediction," *arXiv preprint arXiv:2201.05724*, 2022.

[192] G. D. Bader and C. W. Hogue, "An automated method for finding molecular complexes in large protein interaction networks," *BMC bioinformatics*, vol. 4, no. 1, pp. 1–27, 2003.

[193] B. K. Shoichet, I. D. Kuntz, and D. L. Bodian, "Molecular docking using shape descriptors," *Journal of computational chemistry*, vol. 13, no. 3, pp. 380–397, 1992.

[194] D. G. Corneil and Y. Perl, "Clustering and domination in perfect graphs," *Discrete Applied Mathematics*, vol. 9, no. 1, pp. 27–39, 1984.

[195] S. Khot, "Ruling out PTAS for graph min-bisection, dense k-subgraph, and bipartite clique," *SIAM Journal on Computing*, vol. 36, no. 4, pp. 1025–1071, 2006.

[196] A. Montanaro, "Quantum speedup of branch-and-bound algorithms," *Physical Review Research*, vol. 2, no. 1, p. 013056, 2020.

[197] R. Sotirov, "On solving the densest k-subgraph problem on large graphs," *Optimization Methods and Software*, vol. 35, no. 6, pp. 1160–1178, 2020.

[198] V. E. Lee, N. Ruan, R. Jin, and C. Aggarwal, "A survey of algorithms for dense subgraph discovery," in *Managing and mining graph data*, pp. 303–336, Springer, 2010.

[199] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi, "Optimization by simulated annealing," *science*, vol. 220, no. 4598, pp. 671–680, 1983.

[200] Y. Asahiro, K. Iwama, H. Tamaki, and T. Tokuyama, "Greedily finding a dense subgraph," *Journal of Algorithms*, vol. 34, no. 2, pp. 203–221, 2000.

[201] M. Charikar, "Greedy approximation algorithms for finding dense components in a graph," in *International workshop on approximation algorithms for combinatorial optimization*, pp. 84–95, Springer, 2000.

[202] K. Brádler, P.-L. Dallaire-Demers, P. Rebentrost, D. Su, and C. Weedbrook, "Gaussian boson sampling for perfect matchings of arbitrary graphs," *Physical Review A*, vol. 98, no. 3, p. 032310, 2018.

[203] J. M. Arrazola, T. R. Bromley, and P. Rebentrost, "Quantum approximate optimization with Gaussian boson sampling," *Physical Review A*, vol. 98, no. 1, p. 012322, 2018.

[204] M. Aaghabali, S. Akbari, S. Friedland, K. Markström, and Z. Tajfirouz, "Upper bounds on the number of perfect matchings and directed 2-factors in graphs with given number of vertices and edges," *European Journal of Combinatorics*, vol. 45, pp. 132–144, 2015.

[205] G. H. Sasaki, *Optimization by simulated annealing: A time-complexity analysis*. PhD thesis, University of Illinois at Urbana-Champaign, 1987.

[206] J. Ernvall and O. Nevalainen, "An algorithm for unbiased random sampling," *The Computer Journal*, vol. 25, no. 1, pp. 45–47, 1982.

[207] O. F. Thomas, *On the Simulation of Gaussian Quantum Optical States*. PhD thesis, University of Bristol, 2023.

[208] S. Yu, Z.-P. Zhong, Y. Fang, R. B. Patel, Q.-P. Li, W. Liu, Z. Li, L. Xu, S. Sagona-Stophel, E. Mer, *et al.*, "A universal programmable Gaussian Boson Sampler for drug discovery," *arXiv preprint arXiv:2210.14877*, 2022.

[209] R. Mezher, A. F. Carvalho, and S. Mansfield, "Solving graph problems with single-photons and linear optics," *arXiv preprint arXiv:2301.09594*, 2023.

[210] Duality Quantum Photonics, "Quantum Gaussian optics toolkit." https://gitlab.com/dualityqp/qgot_public. Accessed 2023-01-11.

[211] W. P. Grice and I. A. Walmsley, "Spectral information and distinguishability in type-II down-conversion with a broadband pump," *Physical Review A*, vol. 56, no. 2, p. 1627, 1997.

[212] W. McCutcheon, "Structure in multimode squeezing: A generalised bloch-messiah reduction," *arXiv preprint arXiv:1809.02544*, 2018.

[213] P. P. Rohde, W. Mauerer, and C. Silberhorn, "Spectral structure and decompositions of optical states, and their applications," *New Journal of Physics*, vol. 9, no. 4, p. 91, 2007.

[214] Y. Liu, C. Wu, X. Gu, Y. Kong, X. Yu, R. Ge, X. Cai, X. Qiang, J. Wu, X. Yang, *et al.*, "High-spectral-purity photon generation from a dual-interferometer-coupled silicon microring," *Optics Letters*, vol. 45, no. 1, pp. 73–76, 2020.

[215] A. Uvarov and D. Vinichenko, "On randomized estimators of the hafnian of a nonnegative matrix," *arXiv preprint arXiv:2312.10143*, 2023.

[216] L. Bernick, "Modeling human networks using random graphs," *Lecture notes, MIT*, 2018.

[217] J. Spall, X. Guo, T. D. Barrett, and A. Lvovsky, "Fully reconfigurable coherent optical vector–matrix multiplication," *Optics Letters*, vol. 45, no. 20, pp. 5752–5755, 2020.

[218] A. Miyauchi and A. Takeda, "Robust densest subgraph discovery," in *2018 IEEE International Conference on Data Mining (ICDM)*, pp. 1188–1193, IEEE, 2018.

[219] C. E. Tsourakakis, T. Chen, N. Kakimura, and J. Pachocki, "Novel dense subgraph discovery primitives: Risk aversion and exclusion queries," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 378–394, Springer, 2019.

[220] A. Fazzone, T. Lanciano, R. Denni, C. E. Tsourakakis, and F. Bonchi, "Discovering polarization niches via dense subgraphs with attractors and repulsers," *Proceedings of the VLDB Endowment*, vol. 15, no. 13, pp. 3883–3896, 2022.

[221] A. Gheorghiu, T. Kapourniotis, and E. Kashefi, "Verification of quantum computation: An overview of existing approaches," *Theory of computing systems*, vol. 63, pp. 715–808, 2019.

[222] J. Eisert, D. Hangleiter, N. Walk, I. Roth, D. Markham, R. Parekh, U. Chabaud, and E. Kashefi, "Quantum certification and benchmarking," *Nature Reviews Physics*, vol. 2, no. 7, pp. 382–390, 2020.

[223] T. Yamakawa and M. Zhandry, "Verifiable quantum advantage without structure," in *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 69–74, IEEE, 2022.

[224] S. Aaronson, "Summer 2022 Quantum Supremacy Updates." `https://scottaaronson.blog/?p=6645`. Accessed: 2024-01-30.

[225] B. Villalonga, M. Y. Niu, L. Li, H. Neven, J. C. Platt, V. N. Smelyanskiy, and S. Boixo, "Efficient approximation of experimental Gaussian boson sampling," *arXiv preprint arXiv:2109.11525*, 2021.

[226] S. Aaronson, "Gaussian BosonSampling, higher-order correlations, and spoofing: An update." `https://scottaaronson.blog/?p=5868`. Accessed: 2024-01-30.

[227] M. C. Tichy, K. Mayer, A. Buchleitner, and K. Mølmer, "Stringent and efficient assessment of boson-sampling devices," *Physical review letters*, vol. 113, no. 2, p. 020502, 2014.

[228] D. Phillips, M. Walschaers, J. Renema, I. Walmsley, N. Treps, and J. Sperling, "Benchmarking of Gaussian boson sampling using two-point correlators," *Physical Review A*, vol. 99, no. 2, p. 023836, 2019.

[229] U. Chabaud, F. Grosshans, E. Kashefi, and D. Markham, "Efficient verification of boson sampling," *Quantum*, vol. 5, p. 578, 2021.

[230] M. Bentivegna, N. Spagnolo, C. Vitelli, D. J. Brod, A. Crespi, F. Flamini, R. Ramponi, P. Mataloni, R. Osellame, E. F. Galvão, *et al.*, "Bayesian approach to boson sampling validation," *International Journal of Quantum Information*, vol. 12, no. 07n08, p. 1560028, 2015.

[231] Editors of Encyclopaedia of Mathematics, "Bayes formula." `http://encyclopediaofmath.org/index.php?title=Bayes_formula&oldid=45997`, 2024.

[232] S. Aaronson and L. Chen, "Complexity-theoretic foundations of quantum supremacy experiments," *arXiv preprint arXiv:1612.05903*, 2016.

[233] S. Boixo, S. V. Isakov, V. N. Smelyanskiy, R. Babbush, N. Ding, Z. Jiang, M. J. Bremner, J. M. Martinis, and H. Neven, "Characterizing quantum supremacy in near-term devices," *Nature Physics*, vol. 14, no. 6, pp. 595–600, 2018.

[234] C. Oh, L. Jiang, and B. Fefferman, "Spoofing cross-entropy measure in boson sampling," *Physical Review Letters*, vol. 131, no. 1, p. 010401, 2023.

[235] T. Giordani, V. Mannucci, N. Spagnolo, M. Fumero, A. Rampini, E. Rodolà, and F. Sciarrino, "Certification of Gaussian boson sampling via graph theory," *arXiv preprint arXiv:2202.07711*, 2022.

[236] L. Devroye, "Nonuniform random variate generation," *Handbooks in operations research and management science*, vol. 13, pp. 83–121, 2006.

[237] G. Casella, C. P. Robert, and M. T. Wells, "Generalized accept-reject sampling schemes," *Lecture Notes-Monograph Series*, pp. 342–347, 2004.

[238] A. Neville, C. Sparrow, R. Clifford, E. Johnston, P. M. Birchall, A. Montanaro, and A. Laing, "Classical boson sampling algorithms with superior performance to near-term experiments," *Nature Physics*, vol. 13, no. 12, pp. 1153–1157, 2017.

[239] J. S. Liu, "Metropolized independent sampling with comparisons to rejection sampling and importance sampling," *Statistics and computing*, vol. 6, pp. 113–119, 1996.

[240] J. Huh, G. G. Guerreschi, B. Peropadre, J. R. McClean, and A. Aspuru-Guzik, "Boson sampling for molecular vibronic spectra," *Nature Photonics*, vol. 9, no. 9, pp. 615–620, 2015.

[241] I. Bezáková, A. Galanis, L. A. Goldberg, and D. Štefankovič, "The complexity of approximating the matching polynomial in the complex plane," *ACM Transactions on Computation Theory (TOCT)*, vol. 13, no. 2, pp. 1–37, 2021.

[242] U. Chabaud and M. Walschaers, "Resources for bosonic quantum computational advantage," *Physical Review Letters*, vol. 130, no. 9, p. 090602, 2023.

[243] A. Ehrenberg, J. T. Iosue, A. Deshpande, D. Hangleiter, and A. V. Gorshkov, "Transition of anticoncentration in Gaussian boson sampling," *arXiv preprint arXiv:2312.08433*, 2023.

[244] T. C. Ralph, A. Gilchrist, G. J. Milburn, W. J. Munro, and S. Glancy, "Quantum computation with optical coherent states," *Physical Review A*, vol. 68, no. 4, p. 042319, 2003.

[245] M. Bayati, D. Gamarnik, D. Katz, C. Nair, and P. Tetali, "Simple deterministic approximation algorithms for counting matchings," in *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pp. 122–127, 2007.

[246] V. Patel and G. Regts, "Deterministic polynomial-time approximation algorithms for partition functions and graph polynomials," *SIAM Journal on Computing*, vol. 46, no. 6, pp. 1893–1919, 2017.

[247] R. Kotecký and D. Preiss, "Cluster expansion for abstract polymer models," *Communications in Mathematical Physics*, vol. 103, pp. 491–498, 1986.

[248] A. Barvinok, "Computing permanents of complex diagonally dominant matrices and tensors," *Israel Journal of Mathematics*, vol. 232, no. 2, pp. 931–945, 2019.

[249] R. L. Mann and R. M. Minko, "Algorithmic cluster expansions for quantum problems," *PRX Quantum*, vol. 5, no. 1, p. 010305, 2024.

[250] R. W. Johnson, "An introduction to the bootstrap," *Teaching statistics*, vol. 23, no. 2, pp. 49–54, 2001.

[251] R. Lipton, "New directions in testing," *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, vol. 2, pp. 191–202, 1991.

[252] J.-Y. Cai, X. Chen, and P. Lu, "Graph homomorphisms with complex values: A dichotomy theorem," *SIAM Journal on Computing*, vol. 42, no. 3, pp. 924–1029, 2013.

[253] B. Ghosh, "Probability inequalities related to Markov's theorem," *The American Statistician*, vol. 56, no. 3, pp. 186–190, 2002.

[254] J. Williamson, "On the algebraic problem concerning the normal forms of linear dynamical systems," *American journal of mathematics*, vol. 58, no. 1, pp. 141–163, 1936.

[255] M. A. De Gosson, *Symplectic geometry and quantum mechanics*, vol. 166. Springer Science & Business Media, Basel, 2006.

[256] M. Yusofsani, "Symplectic geometry and Williamson's theorem lecture notes," 2018.

[257] R. Myers, *Data management and statistical analysis techniques*. Scientific e-Resources, New Delhi, 2019.

[258] J. Krempasky, "CEP equation exact to the fourth order," *Navigation*, vol. 50, no. 3, pp. 143–149, 2003.

[259] P. P. Rohde and T. C. Ralph, "Error tolerance of the boson-sampling model for linear optics quantum computing," *Physical Review A—Atomic, Molecular, and Optical Physics*, vol. 85, no. 2, p. 022332, 2012.

[260] S. Aaronson, "The equivalence of sampling and searching," *Theory of Computing Systems*, vol. 55, no. 2, pp. 281–298, 2014.

[261] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2012.

Index

additive error, 46, 66
adjacency matrix, 36
advantage, quantum, 26
algorithm, 6
ancilla, 25
anti-concentration, 47
approximation ratio, 73

Baker-Campbell-Hausdorff, 54
bipartite, 74
bipartite graph, 36
Bloch-Messiah decomposition, 80
Bogoliubov transformations, 31
Boolean, 5
Born rule, 16
boson, 19
boson sampling, 40
branch-and-bound, 72

characteristic equation, 53
Church-Turing thesis, 5
    extended, 7
classical, 4, 14
clique, 39, 71
coherence, 14
coherent state, 54
collapse, 16
collision-free, 74
commutator, 22
complete, 8
complexity, 6
computable, 5
counting problem, 10
covariance matrix, 163

decidable, 9
decision problem, 8
decoherence, 14
delta function identity, 52
densest k-subgraph (DkS), 71
density, 37
Dirac notation, 20
displacement operator, 53, 54
docking, 71
double slit experiment, 15

efficient, 7
entanglement, 19, 20

feedforward, 35
fermion, 19
Fock state, 30

gate, 24
Gaussian, 56
GKP state, 58
graph, 36
    colouring, 12
greedy algorithm, 73
ground truth, 45, 102

hafnian, 38, 61, 73
halting problem, 9
hard, 8
harmonic oscillator, 29

Heisenberg picture, 21
Hermitian, 21
heuristic, 6, 73
hiding, 47
Hilbert space, 20

joint spectral amplitude, 83

KLM, 35

marginal, 51, 86
matching, 37
    perfect, 38
matching polynomial, 130
Maxwell's equations, 27
measurement-based quantum computing, 35
mode, 28
monomer-dimer model, 64
multiplicative error, 47, 66

natural units, 50
normalisation, 20

oracle, 8
orthogonal, 20

passive, 31
permanent, 37
polynomial time approximation scheme, 72
postselection, 11
proportional sampling, 78

quadrature, 48
quasiprobability, 50
qubit, 21

random sampling, 77
reduction, 8

scattershot boson sampling, 59
Schrödinger's equation, 17
Schrödinger picture, 21
simulated annealing, 73, 78
simulation, 4
spin, 19
squeezing, 58
stochastic, 77
subgraph, 36
superposition, 16, 20
supremacy, quantum, 26
symplectic, 49

threshold theorem, 26

ultraviolet catastrophe, 14
uncertainty, 22, 23
unitary, 21
universal, 21

verification, 102

wave-particle duality, 15
wavefunction, 15
Wigner function, 51
Williamson decomposition, 80