UNIVERSITY OF CALGARY

Continuous-Variable Quantum Computation of Oracle Decision Problems

by

MARK R. A. ADCOCK

A THESIS
SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF PHYSICS AND ASTRONOMY
INSTITUTE FOR QUANTUM INFORMATION SCIENCE

CALGARY, ALBERTA
DECEMBER, 2012

Abstract

Quantum information processing is appealing due its ability to solve certain problems quantitatively faster than classical information processing. Most quantum algorithms have been studied in discretely parameterized systems, but many quantum systems are continuously parameterized. The field of quantum optics in particular has sophisticated techniques for manipulating continuously parameterized quantum states of light, but the lack of a code-state formalism has hindered the study of quantum algorithms in these systems. To address this situation, a code-state formalism for the solution of oracle decision problems in continuously-parameterized quantum systems is developed.

In discrete-variable quantum computation, oracle decision problems exploit quantum parallelism through the use of the Hadamard transform. The challenge in continuous-variable quantum computation is to exploit similar quantum parallelism by generalizing the Hadamard transform to the continuous Fourier transform while avoiding non-renormalizable states. This straightforward relationship between the operators in the discrete and continuous settings make oracle decision problems the ideal test-bed. However, as the formalism results in a representation of discrete information strings as proper code states, the approach also allows for the study of a wider range of quantum algorithms in continuously-parameterized quantum systems having both finite- and infinite-dimensional Hilbert spaces.

In the infinite-dimensional case, we study continuous-variable quantum algorithms for the solution of the Deutsch–Jozsa oracle decision problem implemented within a single harmonic-oscillator. Orthogonal states are used as the computational bases, and we show that, contrary to a previous claim in the literature, this implementation of quantum information processing has limitations due to a position-momentum trade-off of the Fourier transform. We further demonstrate that orthogonal encoding bases are not unique, and using the coherent states of the harmonic oscillator as the computational bases, our formalism enables quantifying the relative performances of different choices of the encoding bases.

We extend our formalism to include quantum algorithms in the continuously parameterized yet finite-dimensional Hilbert space of a coherent spin system. We show that the highest-squeezed spin state possible can be approximated by a superposition of two states thus transcending the usual model of using a single basis state as algorithm input. As a particular example, we show that the close Hadamard oracle-decision problem, which is related to the Hadamard codewords of digital communications theory, can be solved quantitatively more efficiently using this computational model than by any known classical algorithm.

Acknowledgements

This thesis is dedicated to my wife Kathryn Lynn Adcock (née Gillespie) for her patience and perseverance through the many hours I have spent on this research and thesis preparation. Without her continual support, this thesis would not have been written.

I would also like to give special thanks to my father Antony Herbert Martin Adcock and my mother Heather Dawn Adcock (née Penrose) for giving me the motivation to undertake this endeavour. It is not clear where the motivation to do a part-time Ph.D. comes from. I think it emerges or is incubated from a young age where one is taught to wonder at the things around us. From watching a chrysalis, to thinking how a radio works, to the joys of algebra, the motivation was instilled in me at a young age.

I would also like to express my gratitude to my supervisors Dr. Barry Sanders and Dr. Peter Høyer. We have enjoyed many discussions over the last 7 years, and they have taught me the importance of perspective in the multi-disciplinary field of quantum information. Both supervisors have also instilled in me an appreciation of the English language and, perhaps more importantly, writing for the reader and not just for myself. I hope that this thesis reflects that as well.

I would also like to thank our children Katie, Keith and Stephen for their understanding and patience over the years. Finally, I would like to express my gratitude to General Dynamics Canada for their support.

Table of Contents

List of Symbols

$\vert\alpha\rangle$	A coherent state of the harmonic oscillator. 30, 31
$\vert\theta, \phi\rangle_s$	A coherent spin state of a spin system having $2s + 1$ spin states. 34
δ	Half-width of position measurement window. 64, 73, 141
$\delta(x - x')$	The Dirac-delta functional also expressed as $\langle x\vert x'\rangle$. 25
f	A Boolean function that takes an n-bit input and gives a single bit output. 42
m	An integer that represents the number of times an oracle is queried or an algorithm is called. 43
n	An integer that represents the number of bits or qubits used as function or algorithm input. 42
N	There are a total of $N = 2^n$ n-bit strings. 42, 63
P	Region of momentum domain encoded with discrete information. 59, 64
ϕ	Spherical coordinate for angle of longitude. vii, 34
$\tilde{\phi}(p)$	Momentum representation of the wave function into which information is encoded. The momentum wave function is the Fourier transform of the position wave function $\phi(x)$. 30
$\phi(x)$	Position representation of the wave function employed as the input state of continuous-variable algorithms. The position wave function is the inverse Fourier transform of the momentum wave function $\tilde{\phi}(p)$. 30
$\vert p\rangle$	Continuous-variable momentum state is an eigenstate of the momentum operator $\hat{p}$ with eigenvalue $p \in \mathbb{R}$. $\vert p\rangle$ is the Fourier transform of $\vert x\rangle$. 25
$\mathrm{Pr}^{\perp}_{\checkmark}$	Single query success probability for the continuous-variable algorithm employing orthogonal states. 74, 81, 137
$\mathrm{Pr}^{\sharp}_{\checkmark}$	Single query success probability for the continuous-variable algorithm employing Gaussian states. 81, 137
$\vert\psi\rangle$	Dirac representation of the position wave function $\psi(x) = \langle x\vert\psi\rangle$. 27
θ	Spherical coordinate for angle of latitude. vii, 34
$\vert x\rangle$	Continuous-variable position state is an eigenstate of the position operator $\hat{x}$ with eigenvalue $x \in \mathbb{R}$. $\vert x\rangle$ is the Fourier transform of $\vert p\rangle$. 25

List of Figures

Chapter 1

INTRODUCTION

1.1 Introduction

Perhaps the most significant aspect of quantum computing is the exponential speed-up observed for certain quantum algorithms over their classical counterparts. For example, it is estimated in [1] that the famous Shor factoring algorithm [2] running on a quantum computer has $O(n)$ computational complexity using $O(n^3)$ qubits to factor an n-digit number. In comparison, the best known classical algorithm, the number field sieve, has $O(e^{(nk \log 2n)^{1/3}})$ [1] computational complexity using computing resources superpolynomial in the number of digits.

The superpolynomial speed-up of the quantum factoring algorithm has significant ramifications if scalable quantum computers can be made given that security protocols are based on the apparent difficulty of factoring large numbers on classical computers. Note that we use asymptotic notation [3, 4, 5] throughout this thesis for the purpose of comparing the performance of quantum algorithms to classical algorithms.

Shor's algorithm, and most other quantum algorithms, have been studied in discretely-parameterized quantum systems. Discrete-variable quantum systems, where experimentalists have successfully demonstrated the potential computational power of quantum systems, include trapped ions [6], liquid nuclear magnetic resonance [7] and quantum dots [8]. The processing of quantum information in discrete-variable systems has the advantage of being amenable to the theory of quantum error correction.

Continuous-variable quantum information is the less studied of the two types of quantum information, but there has been significant progress made in recent years in understanding and controlling quantum optics systems [9, 10, 11, 12, 13, 14, 15, 16, 17]. Processing with continuous-variable systems thus has the advantage of being amenable to many physical preparation and measurement procedures that feature continuous tunability. However, there is lack of a formalism governing the use of proper code-states that has hindered the study of quantum algorithms in these systems.

The absence of a rigorous orthogonal code-state formalism in the continuous-variable model of quantum computing impedes the development of quantum algorithms accompanied by error bounds and full resource analysis. Furthermore the literature on continuous-variable quantum information deleteriously blurs two distinct challenges: finite- vs. infinite-dimensional Hilbert spaces and discrete vs. continuous parameterizations. In addition due to a lack of a code-state formalism, there are examples in the literature where false conclusions of the performance of continuous-variable algorithms have been drawn.

In order to develop our formalism, we first need to define what we mean by continuous-variable quantum information. In our study of continuous-variable quantum information, we consider both finite- and infinite-dimensional Hilbert spaces, which are continuously parameterized. We operationally define continuous-variable quantum information in two ways: continuously parameterized preparation and continuously parameterized measurement. In the case of finite-dimensional Hilbert spaces, we use continuously parameterized preparation, but we use discrete measurement. In the case of infinite-dimensional Hilbert spaces, we use continuously parameterized preparation, and we define continuously parameterized measurements in terms of a dense spectrum in the real numbers.

Braunstein and Pati [18] attempted to map the discrete quantum algorithm for the solution of the Deutsch–Jozsa [19] oracle decision problem directly to the infinite-dimensional, continuous-variable setting. Their claim of infinite speedup of the continuous-variable algorithm over the classical deterministic approach was based on the use of improper, non-renormalizable states and was subsequently proven false [20]. Similarly an attempt to encode a qudit into a single harmonic oscillator [21] has been demonstrated to have serious problems making it non-renormalizable as well [22]. The need for a code-state formalism in the infinite dimension setting is clear.

We begin by dealing with the infinite-dimensional case. Oracle decision problems have been studied in the discrete quantum setting since the early days of quantum information theory [23]. We select oracle decision problems as the test-bed for our formalism because in the discrete case, they exploit quantum parallelism through the application of the Hadamard transform. The challenge of continuous-variable quantum computation is exploiting the same parallelism by generalizing the Hadamard transform to the continuous Fourier transform yet avoiding having non-renormalizable states as computational states in both the canonical position domain and its momentum dual. There is a trade-off, and the lessons from studying oracle decision problems will be valuable elsewhere in the study of continuous-variable quantum information processing.

We select the Deutsch–Jozsa [19] problem as the specific example of an oracle decision problem. The Deutsch–Jozsa [19] algorithm is an important early quantum algorithm that demonstrates exponential speedup over its classical deterministic counterpart. It has also been studied in the continuous-variable setting [18]. The structure of quantum algorithms solving oracle decision problems is quite simple. Typically the input state is a computational basis state. It is transformed into an equal superposition of basis states; the oracle acts on the superposition, which is then transformed back to the computational basis for measurement.

Each of the transformations used in the discrete setting has an analogue in the continuous-variable setting. In the infinite-dimensional, continuous-variable setting, the simplest computational model is to use a single mode of the harmonic oscillator. This requires adaptation of the traditional representation of the quantum algorithms employed in the solution of oracle decision problems.

Traditional quantum algorithms for the solution of oracle decision problems use n control qubits and a single target qubit [5]. The most straightforward mapping onto harmonic oscillators thus requires two oscillators. As we wish to use a single oscillator, our novel approach is first to map the traditional quantum algorithm to one employing only n-qubits by removing the requirement for the target qubit. We then extrapolate this new discrete quantum algorithm to a single-mode of a harmonic oscillator [20].

Another challenging aspect of our formalism is the problem of encoding finite information into the momentum basis of the harmonic oscillator. We achieve this by defining proper, orthogonal code-states with unique mapping between elements in the code space and each of the bits contained in the n-bit string loaded into the oracle. We demonstrate that orthogonal encoding bases are not unique, and our formalism enables quantifying the relative performances of different choices of the encoding bases [24].

The single-mode circuit is also applicable to a continuous parameterization of a collection of spin-1/2 atoms. This novel extension of the formalism to a continuously parameterized system having a finite-dimensional Hilbert space, led to the discovery of a new way of efficiently solving a subset of the bounded-distance decoding problem [25]. We refer to this as the restricted close Hadamard problem.

This research is significant because it identifies flaws with continuous-variable algorithms in the literature and addresses these flaws. This work also yields important insight into how to perform quantum optics experiments of continuous-variable quantum computation. Except for the implementation of the oracle, all other aspects of the algorithm can be implemented with the existing tools of quantum options.

This work also contributes some mathematical techniques that can be adopted for research in other areas. This includes the technique of using the separation between the constant and worst case balanced functions for bounding the single-query success

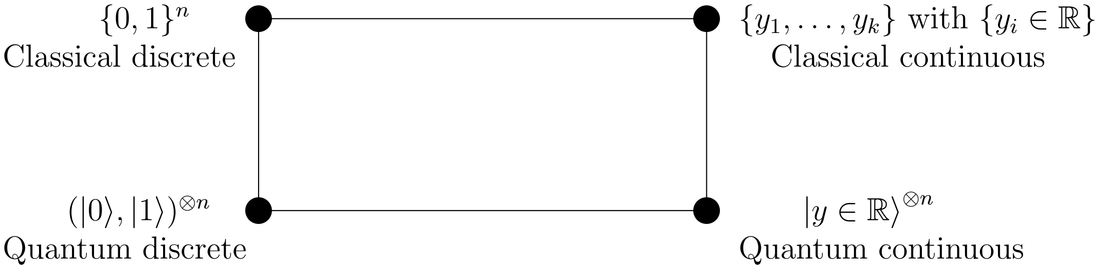
$\{0,1\}^n$
Classical discrete
$\{y_1, \ldots, y_k\}$ with $\{y_i \in \mathbb{R}\}$
Classical continuous
$(|0\rangle, |1\rangle)^{\otimes n}$
Quantum discrete
$|y \in \mathbb{R}\rangle^{\otimes n}$
Quantum continuous

Figure 1.1: The four forms of computation and their respective variables. Here $\{y_1, \ldots, y_k\}$ is a set of real numbers with $k \geq 1$ in order to represent that more than one instance of the real number line can be used.

probability. Since quantum field theories are continuously parameterized, this work could also be employed in the quantum computation associated with these field theories [26].

In order to gain an appreciation of why continuous-variable quantum computation is challenging, we compare it to other forms of computation in the wider classical and quantum settings. In Figure 1.1, we represent the variables used in the four forms of computation, which range from binary digits in the classical discrete case to continuous position states in the quantum continuous case.

In the rest of this introduction, we describe the discrete and continuous versions of both classical and quantum computation. For each of the versions of computation, we describe a relevant model of computation. As there are several models of computation applicable to each version, we select the models that most directly relate to our studies of continuous-variable quantum computation.

For each version of computation, we discuss the conditions of universal computation for the selected model, the types of problems solved by the model, error correction techniques employed, and extending the elements of computation. In Sec. 1.3, we discuss some of the problems associated with computation based on real numbers. For the case of continuous-variable quantum computation, dealing with this problem leads us to the need for a code-state formalism for continuously-parameterized, infinite dimensional systems. We close the introduction with an overview of the thesis organization in Sec. 1.4.

1.2 Discrete computation

Discrete classical computations are carried out on elements of finite sets and discrete quantum computations take place in finite-dimensional Hilbert spaces [27, 28]. In this section, we present a comparison of the two versions of computation highlighting both similarities and differences.

1.2.1 Discrete classical computation

Due to the ubiquity of the digital computer, computation with bits has become synonymous with discrete classical computation. Bits are binary digits usually represented by the set $\{0, 1\}$. Bits may be implemented by means of a two-state device such as, for example, two distinct voltage or current levels allowed by an electronic circuit. Digital computers usually manipulate bits in groups of a fixed size, conventionally named words. The set of strings that can be represented by an n-bit word is written $\{0, 1\}^n$, and for example for $n = 2$,

$$\{0, 1\}^2 = \{00, 01, 10, 11\}. \tag{1.1}$$

As a point of reference, today's personal or server computers have a word size of 32 or 64 bits.

In discrete classical computation theory, many models of computation have been developed. Each model has different capabilities and limitations [29, 3]. Examples of computational models are Boolean circuits [29, 3, 4], the finite-state machine, the random-access machine, the pushdown automaton, and the Turing machine [29]. The Turing machine [30] is a standard model of computation. We select the Boolean circuit model

A	B	$A \vee B$	$A \wedge B$	$A \oplus B$	$\neg A$
0	0	0	0	0	1
0	1	1	0	1	1
1	0	1	0	1	0
1	1	1	1	0	0

Table 1.1: The truth table for the two-input logical gates: OR($\vee$), AND($\wedge$), and XOR ($\oplus$) and the single-input gate NOT ($\neg$).

of computation for discussion here because the Boolean circuit computational model is most appropriate for direct comparison with the circuit model of quantum computation.

A Boolean circuit is a collection of gates having inputs and outputs connected by wires. The wires carry the Boolean values 0 and 1. A logic gate is a device implementing a Boolean function. Logic gates come in two forms: two-wire gates and single-wire gates. The basic two-wire gates are the AND, OR and XOR (exclusive OR) gates, and the single wire gate is the NOT gate. For the inputs $A, B \in \{0, 1\}$, the logical truth tables for these gates are presented in Table 1.1 [4]. Two other important two-wire gates are the NAND gate, which is an AND gate with negated output, and the NOR gate, which is an OR gate with negated output.

A gate, or a set of gates, is considered universal if any Boolean operation can be expressed as a finite sequence of the universal gate or elements from the set [29]. For example, the NAND gate and the NOR gate are both universal gates [29]. The NAND gate is the most ubiquitous in the construction of computers because they are simplest to construct using transistor-to-transistor logic [29]. We note that the sets $\{\mathrm{AND}, \mathrm{NOT}\}$ and $\{\mathrm{OR}, \mathrm{NOT}\}$ are also universal [29, 3, 4]. These sets are mentioned here because they are analogous to similar sets in the circuit model of quantum computation.

Boolean circuits are an important part of the modern digital computer since the “silicon chip”, on which computers are in part based, is a very-large-scale integration of Boolean circuits [3]. The modern digital computer is capable of solving a wide variety of problems, indeed algorithms like the number field sieve [1] for solving the factoring problem execute on modern digital computers. However, the whole notion of how difficult a problem is and what is computable and what is not computable by a digital computer pertains to the study of complexity and computability. These subjects are out of the scope of this thesis. There are many good texts on complexity and the theory of computing for example [3, 4].

When a discrete classical computer exchanges information with another computer or when it reads in information from a physical storage device, there is a chance that data words will be corrupted. Computers deal with this problem through the process of error correction. Classical error-correction techniques include forward error correction using error-resistent codewords [31]. The problem of discriminating between codewords received after transmission over a noisy channel is well known in classical digital coding theory employing linear block codes. Linear block codes include simple repetition codes through the more sophisticated Hadamard and Reed–Solomon codes [32]. Quantum error correcting codes are based upon similar principles to the repetition codes [5].

In discrete classical computation, bits can be generalized to d-dimensional digits called dits. Dits may be implemented by d-state physical devices. We briefly introduce dits here for comparison to their quantum counterparts, which are called qudits. The set of strings comprised of n dits may be represented by the set $\{0, 1, 2, \ldots, d-1\}^n$. Just as the notation $\{0, 1\}^n$ refers to the set of all n-bit strings consisting of zeros and ones, the notation $\{0, 1, 2, \ldots, d-1\}^n$ refers to the set of all n-dit strings consisting of zeros, ones, twos, up to and including the number $d-1$. For example when $d = 3$ we have,

$$\{0, 1, 2\}^2 = \{00, 01, 02, 10, 11, 12, 20, 21, 22\}, \tag{1.2}$$

which has cardinality $3^2 = 9$.

A practical example of dits is the case where $d = 3$, which are referred to as trits or ternary digits [33]. For example, a computer architecture based on a fibre-optic ternary computer using the set $\{-1, 0, 1\}^n$ with dark as 0 and the two orthogonal polarizations of light as 1 and -1 has been proposed [33].

In the next subsection, we discus discrete computation with the quantum analogue of the classical bit. Commonly referred to as the qubit [5], the quantum bit is the basic unit of quantum information.

1.2.2 Discrete quantum computation

The qubit represents the simplest quantum mechanical system. A qubit has a two-dimensional state space. The state space is equipped with an inner product and is referred to as an inner product space. In the finite-dimensional complex vector space employed in discrete quantum computation, a Hilbert space is exactly the same thing as an inner product space [5].

The two computational basis states of the qubit are conventionally written as the vectors (or kets) $|0\rangle$ and $|1\rangle$. In a classical system, a bit would have to be in one state or the other, but quantum mechanics allows the qubit to be in a superposition of both basis states simultaneously [5]. The superposition is expressed as

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{1.3}$$

subject to the constraint $|\alpha|^2 + |\beta|^2 = 1$. The ability to create superpositions is fundamental to quantum computing and is the source of much of the advantage of algorithms implemented in quantum systems. This advantage is sometimes referred to as quantum parallelism, which is the ability of quantum computers to perform computations simultaneously [23].

In vector spaces of higher dimension, the states are expressed in a manner analogous to classical systems. For example, a two-qubit system has the computational basis states expressed as $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$, which is analogous to the representation of the values of two-bits given in Eq. (1.1). These basis vectors are frequently expressed using the Kronecker product notation; thus $|00\rangle = |0\rangle \otimes |0\rangle = |0\rangle^{\otimes 2}$. This notation is extended to define an n-qubit state as

$$|0\rangle|0\rangle \cdots |0\rangle = |00\cdots 0\rangle = |0\rangle^{\otimes n}. \tag{1.4}$$

Any two-level quantum system can be used as a qubit. Multilevel systems can be used as well if two distinct states can be decoupled from the rest [5].

Quantum computation also has models of computation including the circuit model and the one-way quantum computation model [34]. The circuit model of quantum computation is analogous to the Boolean model in the discrete classical setting. The quantum circuit model is among the easiest model to work with and is widely presented in textbooks e.g., [5]. The circuit model consists of unitary operations referred to as quantum gates connected together by quantum wires [35]. Although the wires appear as simple lines on quantum circuits, the quantum wire may not be trivial to implement [35].

The circuit model employs quantum logical gates consisting of single-qubit gates and two-qubit gates. We give an example of a single qubit gate and a two-qubit gate. The quantum mechanical equivalent of the classical NOT gate, shown in Table 1.1, is the single-qubit gate represented by the operator

$$\mathsf{X} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \tag{1.5}$$

which gives $|0\rangle = \mathsf{X}|1\rangle$ and $|1\rangle = \mathsf{X}|0\rangle$. The prototypical two-qubit gate is the controlled-NOT (CNOT) gate whose quantum circuit is presented in Figure 1.2.

The CNOT gate performs an operation similar to the classical XOR gate. The CNOT

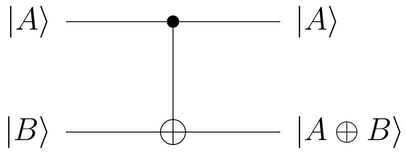

Figure 1.2: Quantum circuit implementing the two-qubit CNOT gate. The upper line represents the 1-qubit 'control' state, and the lower line represents the 1-qubit 'target' state.

gate has the following matrix representation

$$\mathsf{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \tag{1.6}$$

An arbitrary two-level unitary operation on the state space of n qubits may be implemented by single qubit gates and the CNOT gate [5].

An important single qubit gate is called the Hadamard gate and is expressed as

$$\mathsf{H} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{1.7}$$

The Hadamard transform H is widely used to create a uniform superposition of computational basis states [5]. Another common single qubit gate is called the phase gate [5] and is expressed as

$$\mathsf{T} = \begin{pmatrix} 1 & 0 \\ 0 & e^{\mathrm{i}\pi/4} \end{pmatrix}. \tag{1.8}$$

Any single-qubit gate can be simulated to arbitrary precision with a finite sequence of gates from the set $\{\mathsf{CNOT}, \mathsf{H}, \mathsf{T}\}$ [36].

Most discrete quantum algorithms exploiting quantum parallelism solve problems that fall into two broad categories: the hidden subgroup problem and unstructured problems [5]. We single out three quantum algorithms of particular interest in the study of continuous-variable algorithms and discuss their quantum speed-up. Shor's algorithm factors integers in polynomial time on a quantum computer [2] exploiting the algorithm called quantum order finding.

The problem of order finding is an example of the hidden subgroup problem [5], and these problems all share the quality of periodicity for which quantum algorithms can often be used to efficiently determine the period. The Deutsch–Jozsa oracle decision problem is another example of the hidden subgroup problem [5]. The quantum Deutsch–Jozsa algorithm solves the oracle decision problem of whether an unknown string is balanced or constant in a single query, whereas the classical deterministic approach requires an exponential number of queries [19, 37].

The problem of searching a database is an example of what is termed an unstructured problem [5]. On a classical computer, the problem of searching an unstructured database consisting of n items to make a match takes $O(n)$ steps. On a discrete quantum computer, the quantum search algorithm, known as Grover's algorithm, take only $O(\sqrt{n})$ operations [38]. Grovers algorithm has been shown to be optimal [39].

Noise is a problem in both classical and quantum information systems. Quantum computation would not be possible if there were no mechanisms to protect against noise [5]. Quantum error-correcting codes are used to protect quantum information against the effects of noise. Quantum error correction procedures include the use of repetition in a manner analogous to classical bit repetitions. For example, the nine-qubit Shor code [5] protects a single-qubit logical state against arbitrary bit flips and phase flips. Quantum error correction procedures also include the more sophisticated stabilizer codes, which are analogous to classical linear block codes [5].

Analogous to the d-dimensional digit, called the dit, in classical discrete computation, qudits (d-level quantum systems) are an extension of qubits. This an active research area, and it is believed that qudits could speed up certain computing tasks like the simulation of quantum systems. As a particular example, a superconducting phase qudit with $d = 5$ shows promise in the ability to emulate the dynamics of spin systems [40].

The power of quantum computing in the discrete setting is well documented and research in the field continues to develop. However, many quantum systems are naturally parameterized by continuous variables. The ability to manipulate these variables and to perform information related procedures is well developed, but continuous quantum computation is not as well formalized. We discuss the background on continuous computation with real numbers in the next section.

1.3 Continuous computation

The theoretical aspect of continuous classical computation is concerned with the study of computational models based on computing machines that use infinite-precision real numbers [41]. A more practical aspect of continuous classical computation is analogue computation, where one continuously-parameterized system is used to simulate another [42].

Attempts to study continuous-variable quantum computation using a computational basis specified by the real numbers run into difficulty because a precise measurement of a basis state defined by a real number is not possible [20].

1.3.1 Continuous classical computation

The theory of real computation hypothesizes computers that operate using infinite-precision real numbers. The set of real numbers

$$\{y_1, \ldots, y_k\} \text{ with } \{y_i \in \mathbb{R}\} \tag{1.9}$$

having index $k \geq 1$, represents that more than one real number line can be used. However, even for the case where $k = 1$, computation over the real numbers is potentially powerful. The Blum-Shub-Smale machine [41] is a model of computation, which uses a single instance of the real number numbers and is intended to describe computations over the real numbers. Essentially, a Blum-Shub-Smale machine is an extension of the Turing machine [30]. The Blum-Shub-Smale model hypothesizes a computer with registers that can store arbitrary real numbers and that can compute rational functions over the real numbers at unit cost.

The problem with this approach is that infinite-precision real numbers may only be approximated on a computer. For example using a quadruple-precision (128-bit) floating-point representation on a state-of-the-art modern digital computer, gives approximately 34 digits of decimal precision [43]. From this, we see that infinite-precision real numbers would require infinite computer memory. This fundamental limit of real computation has relegated the use of real number computers to theoretical models, and as result, continuous classical computation may only be approximated by discrete classical computation.

An interesting physical aspect of the problem with real numbers is the statement that unlimited precision real numbers in the physical universe are prohibited by the holographic principle and the Bekenstein bound [44]. Essentially the Bekenstein bound is an upper limit on the amount of information that can be contained within a given finite region of space which has a finite amount of energy. Indeed if real computation were physically realizable, one could could compute solutions to problems that are currently not known to be computable [44].

A practical aspect of continuous classical computation is analogue computation. Analogue computers typically use the electrical quantities of inductance and charge as analogues of mass and displacement so that continuously varying voltages may output, for example, the simulated trajectory of a spacecraft [42]. Many other physical phenomena including mechanical or hydraulic quantities are also used to model the problem being solved. Analogue computers are especially well suited to representing systems described by differential equations [42].

Since an electronic circuit can typically operate at higher frequencies than the system being simulated, it is possible for the analogue simulation to run faster than the real time of the simulated system. This approach of simulating one system with another is potentially applicable to the quantum version of continuous-variable computation as well [26]. The problem of infinite-precision real numbers is avoided as long as the desired precision of the simulated system is less than the precision achievable in the measured system [42].

Wireless communications gives another example of how continuous phenomenon may be used in classical computation. The radio frequency portion of the electromagnetic spectrum is modulated with information at the transmitter and the information is then demodulated and recovered at the receiver. The modulation process may by analogue or digital [45]. Thus in the case of digital modulation, the continuously-parameterized radio frequency spectrum serves as a substrate with discrete information embedded in it. We show that this approach of modulating a continuous system with discrete information is important in the quantum version of continuous-variable computation [25, 24]. The limit on the precision of real numbers is applicable to continuous-variable quantum computation as well.

1.3.2 Continuous quantum computation

Continuous-variable studies are often based on quantum optics because of the wide variety of tools that are have been developed to process and measure optical field modes [46]. A number of early successes including the quantum teleportation of optical fields [10] galvanized research into quantum information processing in continuous variables.

Other significant developments in the field include the development of the conditions required for construction of a universal quantum computer over continuous variables [47], error correction for continuous variables [14], continuous-variable quantum cryptography [11] as well as algorithms for continuous variable quantum computation, which we discuss further in the following.

In standard discourse on continuous-variable quantum computation, the set of position states $\{|x\rangle \in \mathbb{R}\}$ serves the same role as the computational basis serves in discrete quantum computation [9, 18]. These states form an uncountably infinite set and are normalized by $\langle x|x'\rangle = \delta(x - x')$. We shall show that this normalization approach is problematic. The dual momentum basis is given by the Fourier transform as

$$|p\rangle = \int \mathrm{d}x \, e^{\mathrm{i}xp} |x\rangle. \tag{1.10}$$

In continuous-variable quantum algorithms solving oracle decision problems, information is encoded into the momentum basis [20, 24].

In continuous-variable quantum computation, an extension of the discrete quantum circuit model is employed, where the continuous Fourier transform given by Eq. (1.10) plays an important role analogous to the role that the Hadamard operator given in Eq. (1.8) plays for qubits. The Fourier transformation creates the continuous-variable equivalent of a superposition of the position eigenstates. This allows the quantum parallelism to be exploited in the continuous-variable setting.

The condition of universal quantum computing in continuous-variable setting requires the capability to construct systems that efficiently approximate any unitary evolution of the system. This can be achieved by having components whose corresponding unitary operators are generated by Hamiltonians of up to and including third order in annihilation and creation operators [47].

There are two examples of continuous-variable quantum algorithms in the literature. The first is the quantum search algorithm [48]. This algorithm demonstrates that the same speedup observed using Grover's quantum search algorithm [38] on a discrete quantum computer can be achieved on a continuous-variable computer. This algorithm uses a modified projection operator having the effect of spreading the projection over a finite interval. This results in avoiding non-renormalizable states as computational states in both the canonical position domain and its momentum dual.

The second is the continuous-variable version of the Deutsch–Jozsa algorithm [18]. In the analysis of this algorithm given in [18], it is claimed that in the idealized continuous-variable case, the speed-up is actually better than the exponential speed-up achieved in the discrete variable case [18]. However, this continuous-variable version of the Deutsch–Jozsa has a fundamental problem because it uses non-renormalizable, infinitesimal states. This example demonstrates the essence of the challenge of providing a code-state formalism for the continuous-variable quantum computation case.

Continuous-variable error correction techniques have been proposed [14] that build on the techniques used in discrete quantum computation. A scheme analogous to the nine-qubit Shor code [5] is the 5-wave-packet code that can correct arbitrary single-wave packet errors. Note that the error syndrome is read with only finite precision, so this error correction is essentially a discrete model that uses continuously-parameterized fields. This notion of approximating continuous-variables through finite-precision measurement is pervasive in the field [9] and is indicative of the need for a code-state formalism.

The challenge of continuous-variable quantum computation is to exploit the parallelism observed in the discrete case by generalizing the Hadamard transform to the continuous Fourier transform while avoiding non-renormalizable states as computational states. We do this by defining proper, orthogonal code-states with unique mapping between elements in the code space and each of the bits contained in the n-bit string processed by the algorithm.

In addition to using continuous-variable systems to implement quantum algorithms to demonstrate quantum speed-up, there is also active research in using continuous-variable quantum systems as analogues of other systems. Of particular interest is the work on quantum field theories [26], which proposes the quantum version of an analogue computer. In the next subsection, we describe overall thesis organization.

1.4 Thesis organization

This thesis is organized as follows. In Chapter 2, we present the background applicable to the proofs given in later sections. We begin with a comparison of how the quantum advantage differs between the discrete-variable and continuous-variable settings. We then discuss the coherent states of the harmonic oscillator, which are continuously-parameterized and infinite dimensional. Continuous-variable studies are typically linked to harmonic oscillators because quantum optics has powerful tools to prepare, process and measure optical field modes [46], which are analogous to harmonic oscillators. We also present continuously parameterized coherent spin states in a finite dimensional Hilbert space.

For both the infinite- and the finite-dimensional cases, we discuss continuous representations of coherent states particularly visualizing them using Wigner functions and Q-functions. These quasi-probability distributions [49] are a useful means of visualizing coherent states since they provide good intuition as to the orientation and the isotropic or anisotropic distribution of uncertainties.

In Sec. 2.4, we define continuous-variable quantum information in terms of the spectrum of a positive-operator valued measure. Infinite-dimensional Hilbert spaces are said to be separable [50], which means that defining the spectrum of measurement in terms of real numbers is problematic. Instead we require that the spectrum of this positive-operator valued measure be dense in the real numbers.

We follow the discussion on the spectrum of measurement with the definition of oracle decision problems. We then formally define the Deutsch–Jozsa [19] oracle decision problem. We finish the background section with a subsection of algorithms for the solution of the Deutsch–Jozsa problem. These algorithms include the classical deterministic and randomized approaches, the traditional quantum approach, which includes the target qubit in the discrete-variable quantum setting, and the two-mode continuous-variable algorithm, which requires a target qubit specified by an infinite-precision real number given in [18]. For each algorithm, we describe the performance in terms of the algorithm query complexity, which we couch in terms of the single-query success probability.

In Chapter 3, we begin with the derivation and performance analysis of the discrete quantum algorithm that does not require the target qubit. We then map this into a continuous-variable algorithm, which uses a single mode of the harmonic oscillator. We encode quantum information into orthogonal states and show that, contrary to a previous claim, this implementation of quantum information processing has limitations due to a position-momentum trade-off of the Fourier transform, analogous to the famous time-bandwidth theorem of signal processing [20].

In Chapter 4, we demonstrate the flexibility of our code-state formalism by showing that the orthogonal encoding bases are not unique. We change the input states of the single-mode algorithm to the coherent states of the harmonic oscillator represented as Gaussian wave functions. The ability to 'tune' the spread of the Gaussian wave function results in a more efficiently encoded momentum wave function leading to improved single-query success probability [24].

In Chapter 5, we introduce the continually parameterized spin-system model having a finite dimensional Hilbert space [25]. We discuss spin squeezing and show how squeezing changes the amplitude distribution of the individual spin states. We show, that for a particular coherent spin state, the limiting squeezed state is asymptotically approximated by a symmetric superposition of two discrete states with constant error independent of the size of the Hilbert space. We use this superposition as the algorithm input state.

We demonstrate that this optimally squeezed state may be approximated well by a superposition of two discrete states, thus idealizing the computation model beyond encoding into squeezed spin states while keeping the spin gates. This approach allows us to discover a new algorithm that can be processed using the circuit model of quantum computation. Our investigation of a continuous-variable spin model of quantum computation has thus inspired us to find a new quantum algorithm.

We summarize our conclusions and contributions in Chapter 6. We also suggest some ideas for future research directions and offer some closing remarks.

Chapter 2

BACKGROUND

2.1 Introduction

In this chapter, we present the background material that we build upon in later chapters. We begin by making a side-by-side comparison of what we refer to as the quantum advantage in the discrete-variable and the continuous-variable settings. We then review continuous-variable systems of coherent states. We give an overview of the coherent states of the continuously-parameterized infinite dimensional system corresponding to the position and momentum representation of quantum optics [46, 54]. We also present the coherent spin states [54] of a spin system that is continuously-parameterized but finite dimensional. In both cases we discuss squeezing and other physically realizable operators. We also present different continuous representations and in particular, the use of Wigner functions and Q-functions for visualizing coherent and squeezed states.

Quantum information theory is well developed for discrete systems, and a good way to gain insight into a new concept is to compare it to an old and established one. In Sec. 2.5, we define oracle-decision problems in general and select the Deutsch–Jozsa as the particular oracle-decision problem used in our formalism. In Sec. 2.6, we present classical algorithms for the solution of the Deutsch–Jozsa problem. We follow this with a brief analysis of the traditional $(n+1)$-qubit discrete-variable quantum algorithm. We close the chapter with an analysis of the two-mode continuous-variable algorithm given in [9] highlighting its problems.

2.2 The quantum advantage

One of the advantages quantum algorithms over classical information processing is the ability to create superpositions of all basis vectors and then operate on the superposition so that all the basis vectors are affected simultaneously. The affected superposition is then transformed back to the computational basis where measurement is performed. In this section, we demonstrate how these superpositions differ between the discrete and continuous settings.

2.2.1 The advantage with discrete variables

In the study of discrete quantum information, we are interested in the complex vector space $\mathbb{C}^n$, where n is a finite integer that represents the dimension of the space. An arbitrary column vector in the space is expressed as the 'ket' $|\psi\rangle$. The norm of a vector is defined by

$$|\psi| = \sqrt{\langle\psi|\psi\rangle}. \tag{2.1}$$

The row vector $\langle\psi|$ is referred to as the bra vector and is said to be dual to the ket $|\psi\rangle$. The quantity $\langle\psi|\psi\rangle$ is the inner product of the vector $|\psi\rangle$ with itself.

A spanning set of vectors $|v_i\rangle$ is such that any vector can be expressed

$$|\psi\rangle = \sum_i a_i|v_i\rangle. \tag{2.2}$$

The complex numbers a_i are the coordinates of the vector $|\psi\rangle$ in the $|v_i\rangle$ basis. We will see that in the continuous case, the coordinate role is assumed by what is referred to as a wave function.

Hilbert space is an inner-product space having the key features of orthogonality and completeness. The orthogonality condition between two basis vectors is expressed as the inner product relation

$$\langle i|j\rangle = \delta_{ij}, \tag{2.3}$$

where $\delta_{ij} = 1$ for $i = j$ and zero otherwise. The completeness relation is

$$\sum_i |i\rangle\langle i| = \mathsf{I}, \tag{2.4}$$

for I the n-dimensional identity operator in this case. These relationships have continuous analogues, which we discuss in the subsection on the advantage with continuous variables.

In the discrete version of quantum information theory, the computational basis is usually used as the reference basis. In $\mathbb{C}^2$, this basis consists of the states $|0\rangle$ and $|1\rangle$, which are represented by the column vectors

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \text{ and } |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{2.5}$$

The real vectors in $\mathbb{C}^2$ having the greatest inner product with these two basis vectors are the diagonal vectors

$$|+\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix}, \text{ and } |-\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ -1 \end{pmatrix}. \tag{2.6}$$

In $\mathbb{C}^2$, the square of the inner product between any diagonal basis vector and any computational basic vector has numerical value of $1/2$. In $\mathbb{C}^2$, the vectors are separated by 45 degrees, and their inner product represents the largest overlap achievable in $\mathbb{C}^2$.

The diagonal vectors $|+\rangle$ and $|-\rangle$ are related to the computational basis vectors by the Hadamard operator

$$\mathsf{H} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{2.7}$$

The diagonal vectors are equal superpositions of the basis vectors since

$$|+\rangle = \mathsf{H}|0\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right), \tag{2.8}$$

and

$$|-\rangle = \mathsf{H}|1\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right). \tag{2.9}$$

The Hadamard operator makes a regular appearance in quantum information theory because it is the simplest way to create these superpositions although many other unitary operators can achieve the same effect.

The results of the above are easily extensible to $\mathbb{C}^n$. For example using the Kronecker product notation we have

$$\mathsf{H}^{\otimes n}|0\rangle^{\otimes n} = \frac{1}{\left(\sqrt{2}\right)^n}|+\rangle^{\otimes n}, \tag{2.10}$$

and

$$\langle 0|+\rangle^{\otimes n} = 2^{-n/2}. \tag{2.11}$$

We now compare how this maximum overlap concept maps over to the continuous-variable setting.

2.2.2 The advantage with continuous variables

We demonstrate the quantum advantage in the continuous-variable setting through a similar expression of maximum overlap between superpositions of basis states and the computational basis states as we did for the discrete case. In the discrete case, the Hadamard transform creates superpositions of the computational basis states. In the infinite-dimensional continuous variable setting, the role of the Hadamard transform is performed by the continuous Fourier transform [20, 24].

In standard continuous-variable discourse [9, 46], the position basis serves as the computational basis. The eigenvectors of the position operator $\hat{x} = \hat{x}^\dagger$, are typically used as an orthonormal basis. Although the definitions we use for the position operator, the position states and their momentum counterparts are problematic [46], they are useful for correctly defining the maximum overlap and the position and momentum wave functions [46] we require for our code-state formalism.

The state $|x\rangle$ is defined to be the eigenstate of $\hat{x}$ with eigenvalue x

$$\hat{x}|x\rangle = x|x\rangle. \tag{2.12}$$

Since the eigenvalues $x \in \mathbb{R}$, the Dirac δ function is used to represent the normalization [9]. The orthonormality condition is now represented as

$$\langle x|x'\rangle = \delta(x - x'). \tag{2.13}$$

The problem with this approach is that $\delta(x - x')$ is not square-integrable, and the definition of the continuous-variable equivalent of the vector norm given by Eq. (2.1) is problematic.

A function $f(x)$ is square-integrable if and only if

$$\int_{-\infty}^{\infty} |f(x)|^2 dx < \infty. \tag{2.14}$$

Complex-valued functions meeting this requirement are said to be in $\mathcal{L}^2(\mathbb{R})$, which is a Hilbert space. By the definition of the δ functional and for $x' = 0$, we have

$$\int_{-\infty}^{\infty} f(x)\delta(x)dx = f(0). \tag{2.15}$$

To see that $\delta(x)$ is not square-integrable, let $f(x) = \delta(x)$ in Eq. 2.15. Then,

$$\int_{-\infty}^{\infty} \delta(x)\delta(x)dx = \delta(0). \tag{2.16}$$

Since $\delta(0)$ is an infinitesimally thin line of unbounded height, we see that $\delta(x)$ does not meet the square-integrable requirement. We will need to recover the square-integrable feature in our formalism, but first we discuss the completeness of the position eigenstates and the relation between the position states $|x\rangle$ and the momentum states $|p\rangle$.

Analogous to the sum of states representation of completeness in the discrete case given by Eq. (2.4), the completeness of the position states is expressed in integral form as

$$\int dx|x\rangle\langle x| = 1. \tag{2.17}$$

This allows us to write the momentum ket as

$$|p\rangle = \int dx\, \langle x|p\rangle|x\rangle. \tag{2.18}$$

We develop an expression for the overlap $\langle x|p\rangle$.

The eigenvectors of the momentum operator $\hat{p}$ are

$$\hat{p}|p\rangle = p|p\rangle \tag{2.19}$$

having completeness

$$\int dp|p\rangle\langle p| = 1. \tag{2.20}$$

The fundamentals of quantum mechanics tells us that the position and momentum operators are related by

$$\hat{p} = -\mathrm{i}\,\hbar\frac{\partial}{\partial \mathrm{x}}, \tag{2.21}$$

which allows us to write

$$\langle x|\hat{p}|p\rangle = p\langle x|p\rangle = -\mathrm{i}\,\hbar\frac{\partial}{\partial x}\langle x|p\rangle. \tag{2.22}$$

This leads to the differential equation

$$\frac{\partial}{\partial x}\langle x|p\rangle = \frac{\mathrm{i}p}{\hbar}\langle x|p\rangle, \tag{2.23}$$

which has the solution

$$\langle x|p\rangle = e^{\mathrm{i}px}, \tag{2.24}$$

where for convenience we have set $\hbar = 1$. We adopt this convention for the remainder of this thesis.

Insertion of the overlap given by Eq. (2.24) into Eq. (2.18) gives

$$|p\rangle = \int dx\, e^{\mathrm{i}px}|x\rangle. \tag{2.25}$$

This expression demonstrates that the position and momentum states are Fourier transform pairs. Eq. (2.25) also highlights the extra degree of freedom we have in continuous variables because it allows for infinitely precise position states and unbounded momentum states and vice versa while maintaining the same magnitude of overlap. However, these states suffer from not being square-integrable. We address this problem by introducing square-integrable wave functions into the formalism.

Using the completeness of the position states given in Eq. (2.17), we express the wave function $\psi(x)$ as the vector $|\psi\rangle$ in the position basis as

$$|\psi\rangle = \int dx\,|x\rangle\langle x|\psi\rangle = \int dx\,\psi(x)|x\rangle, \tag{2.26}$$

where by definition $\psi(x) \equiv \langle x|\psi\rangle$. The wave function $\psi(x)$ thus represents the coordinates of the state $|\psi\rangle$ in the $|x\rangle$ basis. Similarly, the position wave function $\psi(x)$ may be expressed in terms of the momentum wave function $\phi(p)$ using the completeness of the momentum states given in Eq. (2.20)as

$$\begin{aligned}\psi(x) = \langle x|\psi\rangle = \langle x| \left(\int dp\, |p\rangle\langle p|\right) |\psi\rangle \\ = \int dp\, \langle x|p\rangle\langle p|\phi\rangle \\ = \int dp\, e^{\mathrm{i}px}\phi(p), \end{aligned} \tag{2.27}$$

where $\phi(p) \equiv \langle p|\psi\rangle$.

We impose the square-integrable requirement on the wave functions $\psi(x)$ and $\phi(p)$ such that

$$\int dx\, |\psi(x)|^2 = 1, \tag{2.28}$$

and

$$\int dp\, |\phi(p)|^2 = 1. \tag{2.29}$$

Wave functions meeting these criteria and the Fourier transform requirement given in Eq. (2.27) form the fabric of our code-state formalism in the infinite-dimensional case. In the next section we discuss coherent states, which give us wave functions based on physical systems that meet these requirements.

2.3 Coherent states

The method of generalized coherent states has been successfully used to describe a number of diverse physical phenomena including quantum optics, atom-light interactions, and superfluidity [49]. In this section, we present the coherent states of the harmonic oscillator and coherent spin states. These systems of coherent states are important in our study of continuous-variable quantum algorithms because they are good examples of infinite- and finite-dimensional, continuously-parameterized quantum systems. In both cases, the coherent wave functions have physical manifestations [46, 51], and they meet the square integrable conditions given in Eqs. (2.28) and (2.29).

2.3.1 Coherent states of the harmonic oscillator

The harmonic oscillator is central to the description of quantized light fields. We give a brief overview of this important topic so that we can introduce the operators and transforms required in our study of continuous-variable quantum algorithms. This system also enable us to draw analogies in our description of coherent spin states.

The harmonic oscillator is described by the Hamiltonian

$$\mathcal{H} = \frac{\hat{p}^2}{2m} + \frac{1}{2}\omega^2 m\hat{x}^2, \tag{2.30}$$

where m is the mass and ω of the oscillator. The operators $\hat{x}$ and $\hat{p}$ are the position and momentum operators of the harmonic oscillator respectively [46], and have commutation relation

$$[\hat{x}, \hat{p}] = \mathrm{i}. \tag{2.31}$$

Note that different systems of coherent states are often described by different operator commutation relations.

The energy eigenstates of the oscillator are given by solutions to the Schrödinger equation

$$\left(\hat{a}\hat{a}^\dagger + \hat{a}^\dagger\hat{a}\right)\psi = \epsilon\psi, \tag{2.32}$$

where ψ is the wave function. The annihilation operator $\hat{a}$ represents the quantized amplitude with which the oscillator is excited and is expressed as

$$\hat{a} = \frac{1}{\sqrt{2}}\left(\hat{x} + \mathrm{i}\hat{p}\right), \tag{2.33}$$

and the corresponding creation operator is

$$\hat{a}^\dagger = \frac{1}{\sqrt{2}}\left(\hat{x} - \mathrm{i}\hat{p}\right). \tag{2.34}$$

The annihilation and creation operators commutations relation is $[\hat{a}, \hat{a}^\dagger] = 1$.

With the ground state represented by the ket $|0\rangle$, the action of the annihilation operator is required to be

$$\hat{a}|0\rangle = 0. \tag{2.35}$$

Substituting back into the Schrödinger Eq. (2.32) and using the quadrature decomposition given in Eq. (2.33), gives the wave function for the vacuum in the position representation as

$$\phi_0(x) = \langle x|0\rangle = \frac{e^{-\frac{x^2}{2}}}{\sqrt[4]{\pi}}. \tag{2.36}$$

The Fourier transform of this expression gives the corresponding momentum representation

$$\tilde{\phi}_0(p) = \langle p|0\rangle = \frac{e^{-\frac{p^2}{2}}}{\sqrt[4]{\pi}}. \tag{2.37}$$

Even though the vacuum state is a state with zero photons, the quadratures still fluctuate with noise variances

$$\Delta^2 x = \langle \phi|x^2|\phi\rangle = \frac{1}{2}, \tag{2.38}$$

and

$$\Delta^2 p = \langle \tilde{\phi}|p^2|\tilde{\phi}\rangle = \frac{1}{2}. \tag{2.39}$$

Note that throughout this thesis, we will use $\phi(x)$ and $\tilde{\phi}(p)$ with appropriate subscripts as the position and momentum representations of the wave functions used in our analysis of continuous-variable algorithms.

The coherent states of the harmonic oscillator are the eigenstates of the annihilation operator

$$\hat{a}|\alpha\rangle = \alpha|\alpha\rangle \tag{2.40}$$

and are usually expressed as the ket $|\alpha\rangle$ with $\alpha = x_0 + \mathrm{i}p_0$. This state can also be created by the displacement operator [46]

$$\hat{D}(\alpha) = \exp\left(\alpha\hat{a}^\dagger - \alpha^*\hat{a}\right), \tag{2.41}$$

which gives

$$\hat{D}(\alpha)|0\rangle = |\alpha\rangle. \tag{2.42}$$

An interpretation of the displacement operator is that it creates an oscillator at α and destroys one at $\alpha = 0$. Coherent states are sometimes referred to as displaced vacuums, but the only thing they share with the vacuum are the noise properties given in Eqs. (2.38) and (2.39).

In the position representation, the displaced vacuum $|\alpha\rangle$ may be expressed as

$$\phi(x;\alpha) = \langle x|\alpha\rangle = \pi^{-1/4}\exp\left[-\frac{(x-x_0)^2}{2} + \mathrm{i}p_0x - \frac{\mathrm{i}p_0x_0}{2}\right], \tag{2.43}$$

where $x_0 = p_0 = 0$ corresponds to the vacuum state. Coherent states may also be expressed in terms of photon number statistics

$$|\alpha\rangle = e^{-\frac{1}{2}|\alpha|^2}\sum_{n=0}^{\infty}\frac{\alpha^n}{\sqrt{n!}}|n\rangle, \tag{2.44}$$

where $|n\rangle$ are the photon number states also called Fock states [46]. Coherent states are often referred to in the literature as Glauber states [46].

Note that vacuum state given by Eq. (2.36) and the displaced state given by Eq. (2.43) have the identical quantum noise properties given by Eq. (2.38). This is a key feature of coherent states and is sometimes described using the ball and stick model [52]. In this description, the ball symbolizes all the quantum noise effects and the stick portion is a purely classical description. Coherent states are thus the most classical of quantum states [53].

The coherent states of the harmonic oscillator are not orthogonal since

$$|\langle\alpha'|\alpha\rangle|^2 = e^{-|\alpha-\alpha'|^2}. \tag{2.45}$$

The coherent states are also complete [46] and can form a coherent-state basis. However, because of the overlap between states given in Eq. (2.45), it is possible to span the vector space with less than the complete set of basis states. For this reason, the coherent states are said to be over complete [49].

Quantum optics has many tools that allow for the manipulation of light. Of further interest in continuous-variable quantum computation is light squeezing, where quantum uncertainties are redistributed altering the shape of the distribution. The squeezing operator is given in [46] as

$$\hat{S}(\zeta) = \exp\left(\frac{\zeta}{2}\left(\hat{a}^2 - \hat{a}^{\dagger 2}\right)\right), \tag{2.46}$$

where the quantity ζ is referred to as the squeezing parameter [46].

In Dirac notation, the squeezed vacuum state may be expressed as

$$|\phi_0\rangle = \hat{S}(\zeta)|0\rangle, \tag{2.47}$$

and in the position representation, a squeezed coherent state is expressed as

$$\phi_s(x;\alpha,\zeta) = \left\langle x \left| \hat{S}(\zeta) \right| \alpha \right\rangle = \pi^{-1/4} e^{\zeta/2} \exp\left[-e^{2\zeta}\frac{(x-x_0)^2}{2} + \mathrm{i}p_0 x - \frac{\mathrm{i}p_0 x_0}{2}\right]. \tag{2.48}$$

In our analysis of continuous-variable algorithms, we use the standard deviation $\sigma = e^{-\zeta}$ to represent the effect of the squeezing operator. We thus re-write Eq. (2.48) giving

$$\phi_\sigma(x;\alpha,\sigma) = \pi^{-1/4}\sigma^{-1/2} \exp\left[-\frac{(x-x_0)^2}{2\sigma^2} + \mathrm{i}p_0 x - \frac{\mathrm{i}p_0 x_0}{2}\right]. \tag{2.49}$$

Note that $\phi_\sigma(x;\alpha,\sigma)$ is in $\mathcal{L}^2(\mathbb{R})$ since

$$\int_{-\infty}^{\infty} |\phi_\sigma(x;\alpha,\sigma)|^2 \, dx = 1. \tag{2.50}$$

We will employ squeezed states in our analysis of quantum algorithms.

Squeezing has the effect of enhancing the variance in one quadrature while reducing the variance in the other while maintaining the overall uncertainty product. Thus we have

$$V_+ = \Delta^2 x = \langle \phi_\sigma | x^2 | \phi_\sigma \rangle = \frac{\sigma^2}{2}, \tag{2.51}$$

and

$$V_- = \Delta^2 p = \langle \tilde{\phi}_\sigma | p^2 | \tilde{\phi}_\sigma \rangle = \frac{1}{2\sigma^2}, \tag{2.52}$$

which maintains the minimum uncertainty product $\Delta x \Delta p = \frac{1}{2}$. Note that the coherent states can be squeezed to an arbitrary degree.

Quantum optics has other important tools that we briefly summarize here. High quality lasers generate light fields that are coherent [46]. The continuous Fourier transform is straightforward to implement as a phase delay with respect to a reference phase [46]. Quantum homodyne detection uses a combination of beam splitters and photodetectors to measure the degree of quadrature squeezing [46]. All these tools would be deployed in a harmonic oscillator implementation of a continuous-variable quantum algorithm solving oracle decision problems. The exception is the application of the oracle.

2.3.2 Coherent spin states

Coherent spin states [54, 53] are analogous to the coherent states of the harmonic oscillator. Whereas the harmonic-oscillator coherent states are *translations* of the oscillator ground state [46], the coherent spin states are *rotations* of the spin-system ground state [51, 53, 49]. Individual spin states are often referred to in the literature [53, 49] as Dicke states analogous to photon number states, and the coherent spin states are referred to as Bloch states analogous to Glauber states.

The spin system dynamics are determined by the Hamiltonian, which is expressed as a polynomial of su(2) algebraic elements. These algebraic elements are Pauli spin operators in the spin-1/2 single-particle case. For higher even dimensions, we use notation similar to [51] with operators $\hat{\mathsf{S}}_i$, $\hat{\mathsf{S}}_j$ and $\hat{\mathsf{S}}_k$ and i, j, k denoting the components of any three orthogonal directions, such that

$$\left[\hat{\mathsf{S}}_i, \hat{\mathsf{S}}_j\right] = \mathrm{i}\hat{\mathsf{S}}_k, \tag{2.53}$$

and

$$\Delta\hat{\mathsf{S}}_i^2 \Delta\hat{\mathsf{S}}_j^2 \geq \frac{1}{4}\left\langle \hat{\mathsf{S}}_k \right\rangle^2, \tag{2.54}$$

and cyclic permutations.

The spin system is oriented in the usual way. With

$$m \in \{-s, -s+1, -s+2, \ldots, s\}, \tag{2.55}$$

the spin kets $|m\rangle_s$ are eigenstates of $\hat{\mathsf{S}}_z$ and S^2 satisfying

$$\hat{\mathsf{S}}_z |m\rangle_s = m |m\rangle_s, \tag{2.56}$$

and

$$\mathsf{S}^2 |m\rangle_s = s(s+1) |m\rangle_s, \tag{2.57}$$

where $\mathsf{S}^2 = \hat{\mathsf{S}}_x^2 + \hat{\mathsf{S}}_y^2 + \hat{\mathsf{S}}_z^2$. The ladder operators are $\hat{\mathsf{S}}_\pm = \hat{\mathsf{S}}_x \pm \mathrm{i}\hat{\mathsf{S}}_y$, and the action of the lowering operator on the ground state is

$$\hat{\mathsf{S}}_- |-s\rangle_s = 0. \tag{2.58}$$

We use the discrete spin states to construct the continuously parameterized coherent spin states.

The coherent spin state, $|\theta, \phi\rangle_s$ with $\theta, \phi \in \mathbb{R}$, is [51]:

$$\begin{aligned}|\theta,\phi\rangle_s &= \mathsf{R}_{\theta,\phi}\,|{-s}\rangle_s \\ &= \left(1+\tan^2\frac{\theta}{2}\right)^{-s} \times \\ &\sum_{k=0}^{2s}\binom{2s}{k}^{\frac{1}{2}}\left(e^{\mathrm{i}\phi}\tan\frac{\theta}{2}\right)^{k}|s-k\rangle_s\,. \end{aligned} \tag{2.59}$$

Note that the rotation operator $\mathsf{R}_{\theta,\phi}$ performs the analogous operation in the spin setting to the displacement operator given by Eq. (2.41) in the harmonic oscillator setting.

Coherent spin states are in $\mathcal{L}^2(\mathbb{R})$ as

$$\int d\phi\, d\theta \sin\theta\, \langle\theta,\phi|\theta,\phi\rangle_s = c, \tag{2.60}$$

with c a constant [54]. The coherent spin state that will be of interest in our study of quantum algorithms in the spin setting is the 'equitorial' spin state.

$$|\pi/2, 0\rangle_s = 2^{-s}\sum_{k=0}^{2s}\binom{2s}{k}^{\frac{1}{2}}|s-k\rangle_s\,. \tag{2.61}$$

This coherent spin state has a Dicke-state amplitude spectrum whose squared magnitude is the binomial probability distribution with $p = q = 1/2$.

Just as the coherent states of the harmonic oscillator can be squeezed, coherent spin states can also be squeezed [51]. Whereas the Glauber states can be squeezed to an arbitrary degree, spin states can only be squeezed to the Heisenberg limit of 1/2 [51]. We employ two-axis counter-twisting [51] to define the squeezing operator

$$\mathsf{S}_\mu = e^{\mathrm{i}\frac{\pi}{4}\hat{\mathsf{S}}_\mathsf{x}} e^{\mathrm{i}\mu\left(\hat{\mathsf{S}}_\mathsf{z}^2-\hat{\mathsf{S}}_\mathsf{y}^2\right)}, \tag{2.62}$$

where μ is the squeezing parameter [51]. The operator $e^{\mathrm{i}\frac{\pi}{4}\hat{\mathsf{S}}_\mathsf{x}}$ orients the resulting anisotropic uncertainty distribution in the y, z directions.

In a manner similar to harmonic oscillator squeezed states, spin squeezing reduces the variance in one direction and enhances the variance in the orthogonal direction while maintaining the minimum uncertainty product. Applying the operator S_μ to

$$|\Psi\rangle = |\pi/2, 0\rangle_s \tag{2.63}$$

allows us to reduce the variance $\Delta\hat{\mathsf{S}}_z^2$ at the expense of enhancing the variance $\Delta\hat{\mathsf{S}}_y^2$. The reduced variance may be expressed as

$$V_- = \langle\hat{\mathsf{S}}_z^2\rangle = \langle\Psi|\, \mathsf{S}_\mu^\dagger \hat{\mathsf{S}}_z^2 \mathsf{S}_\mu \,|\Psi\rangle \tag{2.64}$$

since the first moment $\langle\hat{\mathsf{S}}_z\rangle = 0$.

It is helpful to visualize the effects of translation and squeezing on the coherent states of the harmonic oscillator and rotation and spin squeezing on coherent spin states. We turn to what are referred to as quasi-probability distribution as visualization aids.

2.3.3 Continuous representations

Visualization of coherent and squeezed states is a helpful way to develop intuition into the behaviour of continuous-variable quantum algorithms. We employ two different continuous representations: the Wigner functions to visualize the coherent and squeezed states of the harmonic oscillator and spherical Q-functions to visual coherent and squeezed spin states.

The Wigner function for the arbitrary wave function $\phi(x)$ may be written

$$W(x,p) = \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{\mathrm{i}pz}\phi\left(x + z/2\right)\phi^*\left(x - z/2\right)dz, \tag{2.65}$$

where z is a variable of integration. The Wigner function is a quasi-probability distribution. For some wave functions it takes on negative values, and Gaussian wave functions are the only pure states with non-negative Wigner functions [46]. The two marginal distributions achieved by integrating $W(x,p)$ over each of the variables form proper probability distributions. The Wigner function has many applications, and it has a rich

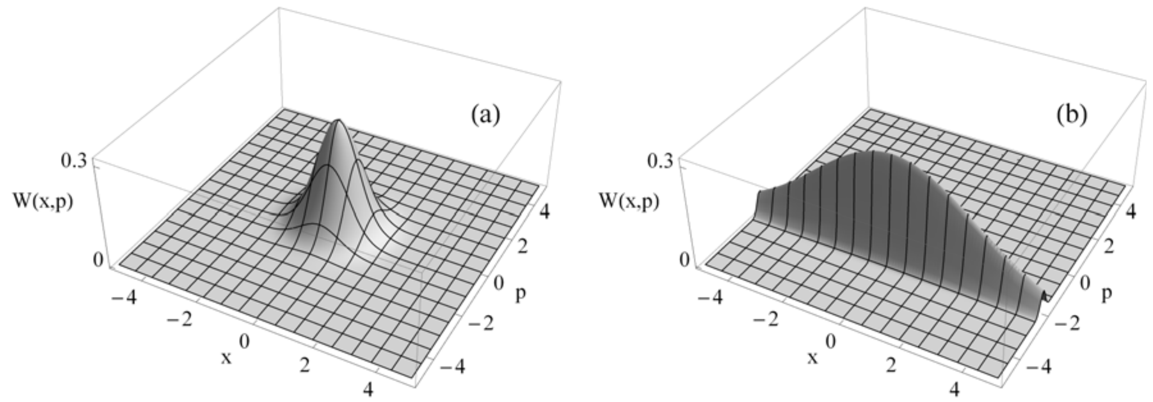

Figure 2.1: (a) The Wigner function of the vacuum demonstrates isotropic fluctuations and (b) The Wigner function for the squeezed coherent state given by Eq. (2.49) with $\alpha = -2\mathrm{i}$ and $\sigma = 3.0$.

history, but we use it as a visualization aid only. We choose some simple wave functions and create their Wigner functions.

Wigner functions for coherent states and squeezed coherent states have simple expressions. For example, the Wigner function for the vacuum given by Eq. (2.36) is

$$W_0(x,p) = \frac{1}{\pi} e^{-p^2-x^2}, \tag{2.66}$$

and for the squeezed coherent state given by Eq. (2.49), the Wigner function is

$$W_s(x,p) = \frac{1}{\pi} e^{-\sigma^2(p-p_0)^2-\frac{(x-x_0)^2}{\sigma^2}}. \tag{2.67}$$

Note how neatly the Wigner function simultaneously captures the quadrature variances given by Eqs. (2.51) and (2.52) in Eq. (2.67). The Wigner function for the vacuum is presented in Figure 2.1(a) and for a squeezed coherent state in Figure 2.1(b). For $\sigma = 3.0$, the enhanced variance in the x direction and the reduced variance in the p direction is clear in Figure 2.1(b). Note that the allowed amount of squeezing of the coherent states of the harmonic oscillator is unbounded.

We give an example of a Wigner function that takes on negative values. The sinc function is an orthogonal wave function, which is in $\mathcal{L}^2(\mathbb{R})$ and is of interest in our study of continuous-variable quantum algorithms. The sinc function in the position representation is

$$\phi(x) = \frac{\sin(Px)}{\sqrt{P\pi}x}, \tag{2.68}$$

and its Fourier transform is the momentum top-hat function

$$\tilde{\phi}(p) = \left(\frac{1}{\sqrt{2P}}\right)\begin{cases} 1, & \text{if } p \in [-P, P] \\ 0, & \text{if } p \notin [-P, P], \end{cases} \tag{2.69}$$

having finite extent of $2P$ in the momentum domain. The Wigner function for the sinc function cannot be simply expressed, so we resort to numerical means to calculate it.

In Figure 2.2, we plot the calculated Wigner function for $P = 2$. In Figure 2.2 (a), we see that quantum interference effects attenuate slowly in the position dimension, whereas they are constrained to $P = \pm 2$ in the momentum direction. In Figure 2.2 (b), we demonstrate that this Wigner function takes on negative values.

The Wigner function is an ideal tool for studying the coherent states of the harmonic oscillator because it is dependent on the cartesian coordinates x and p it naturally captures the effects of translations and squeezing in the $x - p$ plane. The study of coherent spin states requires visualization of rotations and squeezing, and for this we use spherical Q-functions.

For the arbitrary coherent spin state represented as $|\Psi\rangle = \sum_{k=0}^{2s} \alpha_k |k\rangle$, we express the spherical Q-function [55] as

$$Q(\theta, \phi) = \sum_{k=0}^{2s} \binom{2s}{k}^{\frac{1}{2}} \sin(\theta/2)^k \cos(\theta/2)^{2s-k} \alpha_k e^{ik\phi}. \tag{2.70}$$

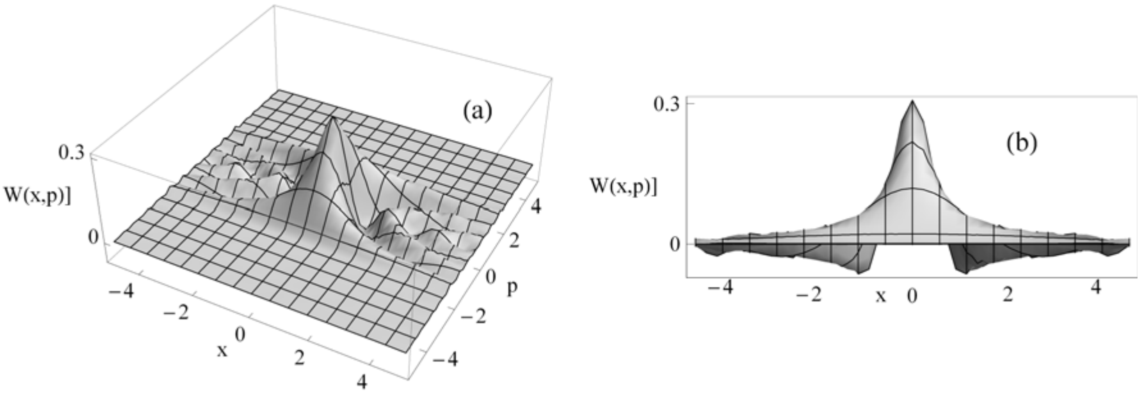

Figure 2.2: (a) The Wigner function the sinc function given by Eq. (2.68). (b) Front elevation view demonstrates negative values.

For plotting, we map the cartesian coordinates to the spherical angles as

$$\begin{aligned} x &\mapsto \sin(\theta)\cos(\phi)\left(1+|Q(\theta,\phi)|^2\right) \\ y &\mapsto \sin(\theta)\sin(\phi)\left(1+|Q(\theta,\phi)|^2\right) \\ z &\mapsto \cos(\theta)\left(1+|Q(\theta,\phi)|^2\right) . \end{aligned} \tag{2.71}$$

This mapping allows us to represent the quasi-probability distributions on the surface of the unit sphere, which enables an intuitive appreciation for the limits on squeezing in the spin setting.

In Figure 5.2(a), we plot the spherical Q-function of continuously-parameterized spin state given by Eq. (5.14). Note that this coherent spin state appears as an 'equatorial' state with isotropic uncertainty distribution when represented this way. In Figure 5.2(b), we plot the effect of the squeezing operator given by Eq. (2.62), where μ is such that the Q-function wraps around the equator and 'touches' itself. This visually demonstrates why the amount of squeezing is bounded in the spin system case.

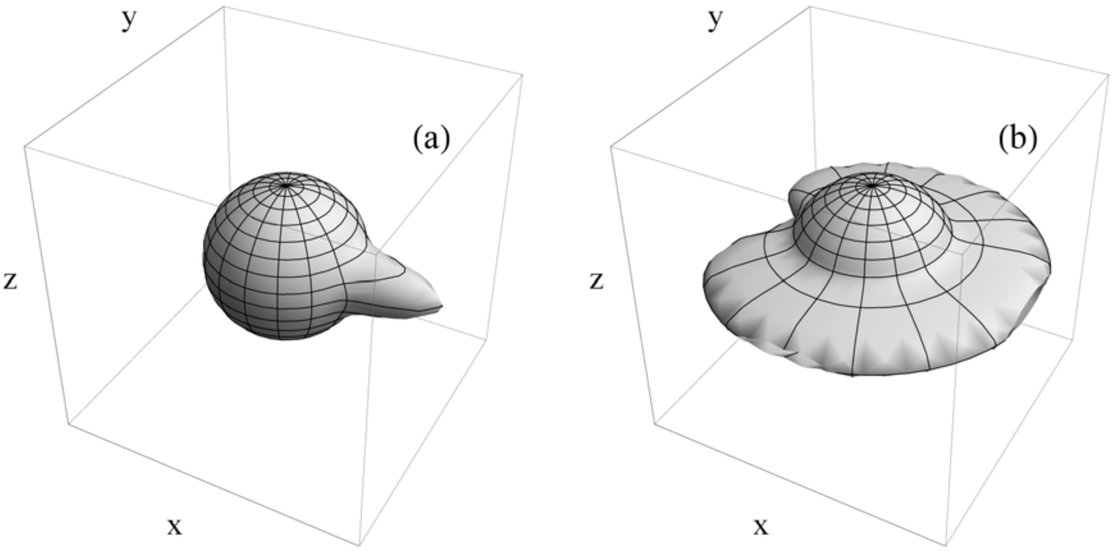

Figure 2.3: (a) Spherical Q-function of the state given by Eq. (5.14) for $s = \frac{63}{2}$, and (b) Squeezed coherent spin state with $\mu = 0.3$.

In this section, we have demonstrated that coherent states can be defined for both finite- and infinite dimensional Hilbert spaces having continuous parameterizations. We have shown that coherent states are square-integrable and that they describe physical states, which can be prepared in the laboratory. For these reasons, we will use coherent states in our code-state formalism.

2.4 Spectrum of measurement

We operationally define continuous-variable quantum information in terms of continuously parameterized preparation and continuously parameterized measurement, and we define continuous-variable continuously parameterized measurement in terms of the spectrum of an observable. The most general observable is a positive operator-valued measure (POVM) [5]. Projective measurements are employed in this thesis, and projective measurement augmented by unitary operations are equivalent to POVMs [5]. We thus present our discussion on spectrum of measurement in terms of POVMs.

A naïve definition of continuous-variable quantum information might be to require the spectrum of the observable in a continuously-parameterized, infinite dimensional system used to process quantum information to consist of real numbers. However, the separability of Hilbert space [50] allows the uncountable basis $\{|x\rangle \in \mathbb{R}\}$ to be replaced by a dense countable basis, for example $\{|q\rangle \in \mathbb{Q}\}$ with $\mathbb{Q}$ the set of rational numbers.

Suppose we define the position POVM

$$\mathsf{P}_x = (|x\rangle\langle x|,\, x \in \mathbb{R})\,. \tag{2.72}$$

The Hilbert space is separable, so there always exists a countable basis. The rational numbers are dense in the reals since every neighbourhood of every real number contains a least one rational number. As a result, we can always define an alternative position POVM

$$\mathsf{P}_q = (|q\rangle\langle q|,\, q \in \mathbb{Q})\,, \tag{2.73}$$

which allows us to construct the countably infinite orthonormal basis $\langle q'|q\rangle = \delta_{qq'}$. The operator P_q is said to be dense in the real numbers. We therefore modify our definition of the measurement spectrum corresponding to continuous-variable quantum information to require that the spectrum of the observable be dense in the reals.

We look at how this modified definition of the spectrum of an observable pertains to a continuously-parameterized, finite dimensional system used to process quantum information. Consider the continuously-parameterized phase state [57] of a spin system having dimension $2s+1$

$$|\theta\rangle_s = \frac{1}{\sqrt{2s+1}} \sum_{m=-s}^{m=s} e^{\mathrm{i}m\theta} |m\rangle_s. \tag{2.74}$$

For this finite-dimensional Hilbert space, there is no way to construct an observable that is dense in the real numbers.

For the interval $I \subset \mathbb{R}$, with $I = [-\pi/2, \pi/2)$, we can construct the POVM

$$\mathsf{P}_\theta = \left(|\theta\rangle_s\langle\theta|,\, \theta \in I \right). \tag{2.75}$$

However, we can't have a countably infinite orthonormal basis set for a finite-dimensional Hilbert space, so we can't construct a POVM whose spectrum is dense in the spectrum of the POVM defined by Eq. (2.75).

We thus conclude that a finite-dimensional system is not a continuous-variable quantum information system from the perspective of measurement, but we can still refer to the finite-dimensional case as "continuous variable" from the perspective of preparation. We can use continuous-variable techniques in the analysis of finite-dimensional systems. For example, we can describe a finite-dimensional spin system in terms of continuously-parameterized spin states [25] and use continuous-variable techniques such as squeezing as we demonstrate in Chapter 5. Indeed this approach of using continuous-variable techniques in finite-dimensional systems is beneficial because it allows us to discover a new quantum algorithm. We continue this background chapter with a discussion of oracle-decision problems and algorithms for their solution.

2.5 Oracle-decision Problems

The challenge of oracle decision problems is to identify which of two mutually disjoint sets contains a unique N-bit string by making the fewest possible queries to an oracle.

2.5.1 Definition

The oracle decision problem is typically couched in terms of a function f that maps n-bit strings to a single bit

$$f : \{0,1\}^n \mapsto \{0,1\}. \tag{2.76}$$

Any Boolean function on n bits can also be represented by a string z of $N = 2^n$ bits, in which the i^{th} bit z_i is the value of the function on the i^{th} bit string, taken in lexicographical order.

We use the definition of an oracle decision problem given in [25] as follows.

Definition 1. *An oracle decision problem is specified by two non-empty, disjoint subsets $A, B \subset \{0,1\}^N$. Given a string $z \in A \cup B = C$, the oracle-decision problem is to determine whether $z \in A$ or $z \in B$ with the fewest queries to the oracle possible.*

We are interested in finding an efficient strategy to identify the property of whether f belongs to set A or to set B without necessarily determining f itself.

2.5.2 The Deutsch–Jozsa oracle decision problem

The Deutsch–Jozsa is a specific example of an oracle-decision problem. The problem is to determine a specific characteristic of an unknown function $f : \{0,1\}^n \to \{0,1\}$ of an n-bit input with as few queries as possible. The special characteristic of interest is that the function is guaranteed to be either *balanced* or *constant*. We give the definition of the Deutsch–Jozsa problem given in [25] as follows.

Problem 1. *Given the set of balanced strings $A \subset \{0,1\}^N$, where exactly $N/2$ elements take on the value 0 and the set of constant strings $B \subset \{0,1\}^N$, where all N elements take on the same value everywhere, and a string z randomly selected with uniform distribution μ such that $z \in_\mu C = A \cup B$, the Deutsch–Jozsa Problem is to determine if $z \in A$ or $z \in B$ with the fewest oracle queries.*

Note that the output of a function that is constant has either all 1's or all 0's; whereas the output of a function that is balanced has equal number of 1's and 0's. For example, for $n = 2$ the constant functions are $\{0000, 1111\}$ and the balanced functions are

$\{0011, 0101, 0110, 1001, 1010, 1100\}$. There are 2^N (with $N = 2^n$) functions in total of which 2 are constant and $\binom{N}{N/2}$ are balanced.

2.6 Algorithms for solving the Deutsch–Jozsa problem

In this section, we present two classical and two quantum approaches to solving the Deutsch–Jozsa problem. In the classical case we, we summarize deterministic and probabilistic algorithms. In the quantum case, we present the traditional discrete quantum algorithm and a two-mode continuous-variable algorithm.

Throughout this thesis, we use the integer m to represent the number of queries made to an oracle or the number of times an algorithm is invoked. The best case is where $m = 1$, and the problem is solved in a single query. If the query complexity is $O(m)$, the solution requires a linear number of queries, and if $m = O(N)$ then an exponential number of queries is required since $N = 2^n$.

2.6.1 The classical version of the Deutsch–Jozsa algorithm

On a classical Turing machine, Problem 1 can be solved deterministically using $m = N/2 + 1$ queries to determine whether the given function is balanced or constant, with certainty. The problem cannot be solved with certainty with fewer than $N/2 + 1$ queries because $N/2$ queries of any balanced function could result in the same value and would thus appear to be a constant function. However, the problem can be solved with much fewer queries by allowing for an incorrect answer with some (small) probability.

A probabilistic algorithm achieves an exponentially small error of 2^{-m} with a number of queries that is only linear in m. To understand how a probabilistic algorithm can help, consider that, although a single query with a random input provides no information, two queries with two random inputs can be highly informative. If the output from the second query differs from the first output, then the function is proved not to be constant and

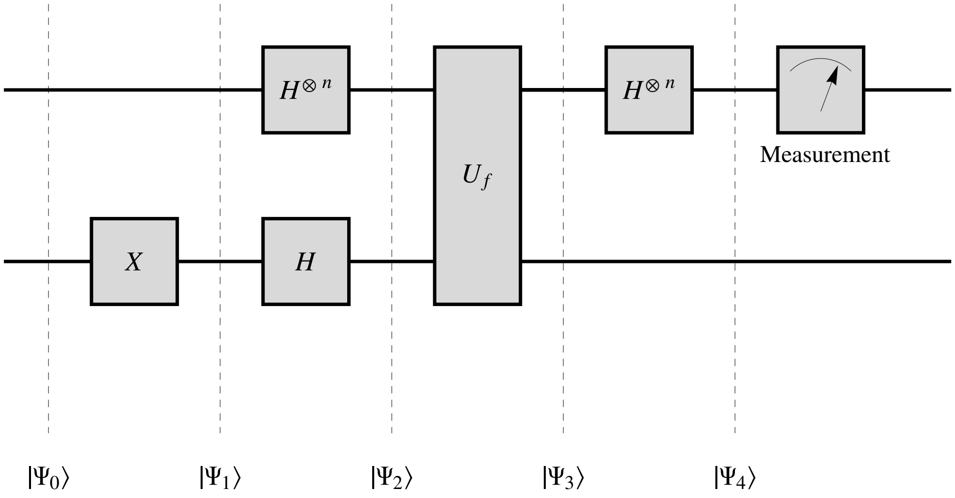

Figure 2.4: Traditional $(n + 1)$-qubit discrete-variable circuit implementation of the Deutsch–Jozsa algorithm. The upper line represents the n-qubit 'control' state, and the lower line represents the 1-qubit 'target' state.

therefore must be balanced. If, on the other hand, the second output is the same as the first, then the outcome is not certain, but the more times the outputs are the same, the more confident one can be about the function being constant.

We calculate the success probability of determining whether a given function f_z is balanced or constant as

$$\Pr_{\checkmark} = 1 - \prod_{j=1}^{m} \frac{N/2 - (j-1)}{N - (j-1)} \geq 1 - \left(\frac{1}{2}\right)^{m}. \tag{2.77}$$

Here the equality is calculated assuming randomized input and sampling without replacement and shows dependency on N, whereas the inequality is based on assuming randomized input and sampling with replacement and is independent of N. We note that the failure probability declines exponentially with the number of queries.

2.6.2 The traditional $(n + 1)$ qubit Deutsch–Jozsa algorithm

The quantum Deutsch–Jozsa algorithm is an important early quantum algorithm, which demonstrates exponential speed-up over its classical counterpart. The quantum Deutsch–

Jozsa algorithm has been shown to solve Problem 1 in a single query [19]. We present a standard circuit version [37] in Figure 2.4. Note that the inclusion of the operator X to create the target state $|1\rangle$ is not the original representation but was given by [37] as a means of having the input in the state $|\Psi_0\rangle = \left|0^{\otimes(n+1)}\right\rangle$. We give a brief and informal analysis of this algorithm here in order to emphasize its salient features.

With reference to Figure 2.4, the first step is to modify the input states $|0\cdots0\rangle|0\rangle \xrightarrow{\mathsf{U}_0} |\Psi_1\rangle = |0\cdots0\rangle|1\rangle$ through the action of the unitary operator $\mathsf{U}_0 = \mathsf{I}^{\otimes n}\otimes\mathsf{X}$. Note that we will suppress the Kronecker symbol $\otimes$ when convenient. The next step is to place this state into an equal superposition of all computational basis states

$$|\Psi_1\rangle \xrightarrow{\mathsf{U}_1} |\Psi_2\rangle = \left(\frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle\right)|-\rangle \tag{2.78}$$

through the action of the operator $\mathsf{U}_1 = \mathsf{H}^{\otimes n}\otimes\mathsf{H}$. The placing of the target qubit into the state $|-\rangle$ is critical since as we shall see, this clever arrangement serves to 'kick back' the phase of the oracle function to the control state superposition.

The crucial step, is to apply the oracle function to transform $|\Psi_2\rangle \xrightarrow{\mathsf{U}_f} |\Psi_3\rangle$ The oracle construct originally proposed by Deutsch–Jozsa is expressed, for $x\in\{0,1\}^n$ and $y\in\{0,1\}$, as

$$U_f : |x\rangle|y\rangle \mapsto |x\rangle|y\oplus f(x)\rangle. \tag{2.79}$$

With this definition of the oracle, the state $|\Psi_3\rangle$ may be represented as follows

$$\begin{aligned}|\Psi_3\rangle =& \frac{1}{\sqrt{2^n}}\left[|0\cdots0\rangle\left(\frac{|0\oplus f(0\cdots0)\rangle - |1\oplus f(0\cdots0)\rangle}{\sqrt{2}}\right)\right.\\ &+|0\cdots1\rangle\left(\frac{|0\oplus f(0\cdots1)\rangle - |1\oplus f(0\cdots1)\rangle}{\sqrt{2}}\right)\\ &+\cdots+|1\cdots0\rangle\left(\frac{|0\oplus f(1\cdots0)\rangle - |1\oplus f(1\cdots0)\rangle}{\sqrt{2}}\right)\\ &\left.+|1\cdots1\rangle\left(\frac{|0\oplus f(1\cdots1)\rangle - |1\oplus f(1\cdots1)\rangle}{\sqrt{2}}\right)\right].\end{aligned} \tag{2.80}$$

The function $f(x)$ can only take on the values 0 and 1. When $f(x) = 0$, the phase of the target qubit $|-\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right)$ in the above expression remains unchanged, but when $f(x) = 1$, the phase of $|-\rangle$ is flipped.

Since flipping the phase is equivalent to multiplying the target state $|-\rangle$ by -1, we can rewrite Eq. (2.80) as follows

$$\begin{aligned} |\Psi_3\rangle &= \frac{1}{\sqrt{2^n}}\big((-1)^{f(0\ldots0)}|0\ldots0\rangle + (-1)^{f(0\ldots1)}|0\ldots1\rangle + \\ &\quad \ldots + (-1)^{f(1\ldots0)}|1\ldots0\rangle + (-1)^{f(1\ldots1)}|1\ldots1\rangle\big) \\ &\quad \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right) \\ &= \left(\frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}(-1)^{f(x)}|x\rangle\right)|-\rangle. \end{aligned} \tag{2.81}$$

Note that the appearance of the term $(-1)^{f(x)}$ multiplying each component of the control state superposition is what we term phase ‘kick back’. The result expressed in Eq. (2.81) means that the oracle definition given in Eq. (2.79) may be equivalently expressed as

$$U_f : |x\rangle|y\rangle \mapsto (-1)^{f(x)}|x\rangle|y\rangle, \tag{2.82}$$

which is a convenient representation for comparison to the oracle implemented in an n-qubit algorithm. This is a demonstration of how the quantum advantage is achieved by exploiting the superposition of all computational basis states.

In the penultimate step, we focus on the control portion only as the target remains unchanged. We express the control portion of the final state as the action of $\mathsf{H}^{\otimes n}$ on $|\Psi_{4\mathrm{c}}\rangle$ as follows

$$|\Psi_{4\mathrm{c}}\rangle = \frac{1}{2^n}\begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & -1 & \cdots & -1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & -1 & \cdots & 1 \end{pmatrix}\begin{pmatrix} (-1)^{f(0\cdots0)} \\ (-1)^{f(0\cdots1)} \\ \vdots \\ (-1)^{f(1\cdots0)} \end{pmatrix}. \tag{2.83}$$

For the two constant cases, we have

$$\left|\Psi_{4\mathrm{c}}^{++}\right\rangle = \pm\frac{1}{2^n}\begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & -1 & \cdots & -1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & -1 & \cdots & 1 \end{pmatrix}\begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \pm\begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \tag{2.84}$$

where the symbol ++ denotes a constant string. The last step in Eq. (2.84) is true since all the rows of $\mathsf{H}^{\otimes n}$ have an equal number of plus 1's and minus 1's except for the first, which results in all amplitudes of the components of the final control state being zero except for the first.

We now consider one of the balanced cases as a point of illustration. For the case where $f = 0101\cdots01$, we can simply express the result as follows

$$\left|\Psi_{4\mathrm{c}}^{+-}\right\rangle = \frac{1}{2^n}\begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & -1 & \cdots & -1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & -1 & \cdots & 1 \end{pmatrix}\begin{pmatrix} 1 \\ -1 \\ \vdots \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}, \tag{2.85}$$

where the symbol +− denotes a balanced string.

The last step in Eq. (2.85) is true since only the second row of $\mathsf{H}^{\otimes n}$ has an alternating number of plus 1's and minus 1's, which results in all amplitudes of the components of final control state being zero except for the second. Since all balanced functions result in a state $\left|\Psi_{4\mathrm{c}}^{+-}\right\rangle$ having and an equal number of plus 1's and minus 1's, the first component of the amplitude is always zero. The non-zero value will land on the component that corresponds to the row in $\mathsf{H}^{\otimes n}$ that matches the phase of the balanced function.

The final step is to make a projective measurement [5] on the first qubit. If the measurement returns a one, the unknown function is constant, and if the measurement returns a zero, the unknown function is balanced. This analysis demonstrates that the

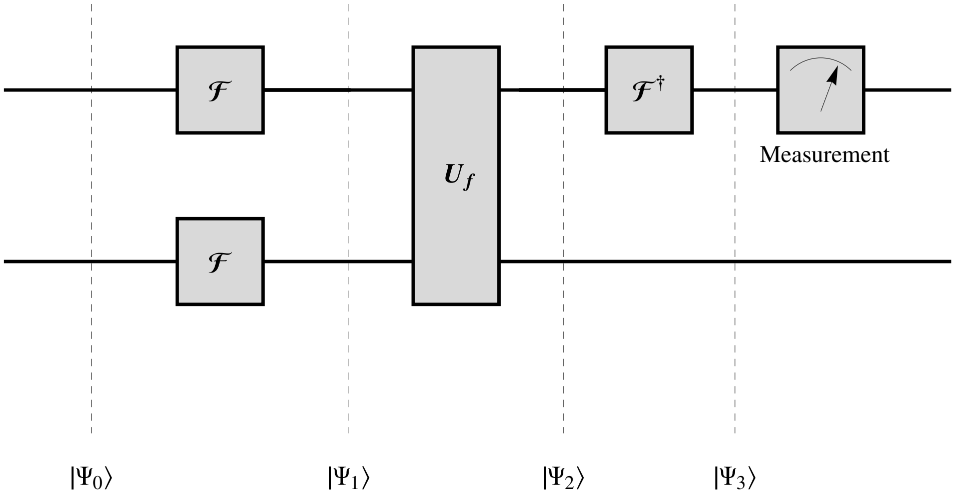

Figure 2.5: Two-mode continuous-variable circuit implementation of the Deutsch–Jozsa algorithm uses two oscillator modes [18]. The upper line represents the 'control' position state, and the lower line represents the 'target' position state.

traditional discrete-variable Deutsch–Jozsa algorithm solves this problem in a single query by relying on an oracle that kicks-back the phase of the function thus exploiting the quantum advantage. In the next, and final subsection of this background chapter, we study an attempt to directly map this version of the discrete quantum algorithm to the continuous-variable setting.

2.6.3 The two-mode continuous-variable Deutsch–Jozsa algorithm

The circuit shown in Figure 2.5 depicts the continuous-variable Deutsch–Jozsa algorithm given in [18]. This algorithm uses two separate modes of the harmonic oscillator, which are represented by the horizontal lines in the figure. The top line is the position eigenstate $|x_0\rangle$, which corresponds to the n-qubit control state in the traditional discrete algorithm, and the bottom line is the position eigenstate $|\pi\rangle$, which corresponds to the 1-qubit target state in the traditional discrete algorithm shown in Figure 2.4. Both the control and the target states are defined to meet the Dirac orthogonality condition $\langle x|x'\rangle = \delta(x - x')$

given in Eq. (2.15).

The steps of the algorithm are analogous to the steps of the discrete algorithm. The input state $|x_0\rangle|\pi\rangle$ is transformed into a superposition by applying the continuous Fourier transform. The superposition is expressed as

$$\mathcal{F}|x_0\rangle\mathcal{F}|\pi\rangle = \frac{1}{\sqrt{2\pi}}\int dx\, dy\ e^{\mathrm{i}x_0x+\mathrm{i}\pi y}|x\rangle|y\rangle. \tag{2.86}$$

Note that we have changed the definition of the Fourier transform given in [18] to the standard transform [56] given in Eq. (2.27) for consistency with [20, 24]. With this definition, the target state is $|\pi\rangle$ instead of $|\pi/2\rangle$ used in [18].

The oracle mapping is the continuous-variable analogue of the discrete mapping given in Eq. (2.79)

$$U_f : |x\rangle|y\rangle \mapsto |x\rangle|y + f(x)\rangle, \tag{2.87}$$

where the + sign is the continuous-variable equivalent of the XOR function [9]. The action of the oracle on the superposition state given by Eq. (2.86) is

$$\begin{aligned}\mathcal{U}_f\left(\mathcal{F}|x_0\rangle\mathcal{F}|\pi\rangle\right) =& \frac{1}{\sqrt{2\pi}}\int dx\, dy\, e^{\mathrm{i}x_0x+\mathrm{i}\pi y}|x\rangle|y + f(x)\rangle \\ =& \frac{1}{\sqrt{2\pi}}\int dx\, dy\, e^{\mathrm{i}x_0x}e^{\mathrm{i}\pi y+\mathrm{i}\pi f(x)}|x\rangle|y\rangle \\ =& \left(\frac{1}{\sqrt{2\pi}}\int dx\, e^{\mathrm{i}x_0x}(-1)^{f(x)}|x\rangle\right)\mathcal{F}|\pi\rangle.\end{aligned} \tag{2.88}$$

We have been careful in the preceding to show that the algorithm was designed to mimic the phase kick-back employed in the traditional discrete quantum algorithm. In particular setting the target position state to $|\pi\rangle$ achieves the desired effect since $e^{\mathrm{i}\pi f(x)} = (-1)^{f(x)}$.

The state after the inverse Fourier transform is given by

$$\mathcal{F}^\dagger\left[\mathcal{U}_f\left(\mathcal{F}|x_0\rangle\mathcal{F}|\pi\rangle\right)\right] = \int dxdx'\, e^{\mathrm{i}\pi(x_0-x')}(-1)^{f(x)}|x'\rangle\mathcal{F}|\pi\rangle. \tag{2.89}$$

Measurement is achieved by projecting this resultant state onto the original state using the idealized continuous-variable projection operator [18].

$$P_{\Delta x_0} = \int_{x_0 - \Delta x_0/2}^{x_0 + \Delta x_0/2} dy|y\rangle\langle y|. \tag{2.90}$$

Note that this measurement has finite spread Δx_0. The balanced and constant argument is then made. The essence of the argument is that if the function is constant, the state given by Eq. (2.89) is simply $\pm|x_0\rangle|\pi\rangle$ and the measurement returns one. If the function is balanced then the output is orthogonal to the constant case and measurement returns zero [18].

The result of this analysis is the claim [18] that the reduction in the number of query calls is from infinity to one, but there are fundamental issues that make this a false conclusion. These issues all stem from the fact that infinitesimal position states are employed as the computational basis, and these states do not meet the square-integrable condition given by Eq. (2.28). Encoding the oracle information into proper square-integrable basis states changes this conclusion [20].

2.7 Summary

In this chapter, we have presented a comparison of the quantum advantage in both the discrete and continuous quantum settings. The maximum state overlap is analogous in the two settings, but the continuous form allows for the potential for ill-defined position states. Our code-state formalism requires square-integrable functions in $\mathcal{L}^2(\mathbb{R})$. Any functions that meet the square-integrable and Fourier-transform pair requirements can be used. Orthogonal wave functions have features that can be beneficially exploited.

Coherent states of the harmonic oscillator and coherent spin states also both meet the square-integrable requirements. Coherent states are over-complete and therefore are said to be non-orthogonal. Coherent states are particularly attractive because they can be created in the laboratory, and there is a collection of tools that can realize most of the algorithm components. Implementation of the oracle remains the biggest challenge.

Oracle decision problems are defined, and algorithms for the solution of the Deutsch–Jozsa oracle decision problem are presented. The traditional quantum algorithm exploits quantum parallelism and solves the problem in a single query. An attempt to map the traditional quantum algorithm to a continuous-variable setting using two modes of the oscillator draws false conclusions because of the use of improper, non-renormalizable states. In the next chapter we introduce a code-state formalism employing orthogonal wave functions in $\mathcal{L}^2(\mathbb{R})$, which enables the correct analysis of algorithm performance.

Chapter 3

CONTINUOUS-VARIABLE QUANTUM COMPUTATION WITH ORTHOGONAL STATES

3.1 Introduction

In continuous-variable quantum information procedures, the input state is typically a Gaussian wave function over the canonical position representation with the physical system being a harmonic oscillator (equivalently a single-mode optical field [46]). However, the unbounded extent of these wave functions does not naturally fit with the finite length of the information strings being processed. To deal with the problem of finite-length information strings, we first develop our code-state formalism using orthogonal wave functions [20].

In this chapter, we focus on a continuous-variable generalization of the Deutsch–Jozsa algorithm to the case of a single mode of light. To deal with the single mode of light, we first introduce a variation of the traditional discrete-variable Deutsch–Jozsa algorithm that employs only n qubits. We then map this algorithm into the continuous-variable setting using a single mode of light and a Fourier transform performing the equivalent of the Hadamard transformation.

There are three advantages in working with this approach. One is that Braunstein and Pati [18] introduced a two-mode continuous-variable Deutsch–Jozsa algorithm, which we analyzed in Sec. 2.6.3. We can learn from their approach of requiring infinitesimal position states. The second advantage is that the single mode continuous-variable Deutsch–Jozsa algorithm employs standard quantum optics techniques. The third advantage is that we can demonstrate how the Fourier transform is itself responsible for the limitations of continuous-variable schemes.

This chapter is organized as follows. In Sec. 3.2, we introduce a recasting of the traditional Deutsch–Jozsa algorithm from an $(n+1)$-qubit algorithm to an n-qubit algorithm, which does not employ the target qubit. In Sec. 3.3, we motivate the selection of the continuous-variable model and introduce techniques for bounding the success probability. In Sec. 3.4, we calculate the single-query success probability $\Pr_{\checkmark}^{\perp}$ and bound the query complexity after m repetitions. We summarize in Sec. 3.5.

3.2 An n-qubit quantum Deutsch–Jozsa algorithm

The quantum Deutsch–Jozsa algorithm has been shown to solve Problem 1 in a single query [19]. We presented an analysis of the traditional algorithm in Sec. 2.6.2 using the standard circuit version [37] presented in Figure 2.4. This standard circuit employs $n+1$ qubits. The extra qubit is referred to as the *target* qubit and is represented by the lower line in Figure 2.4.

In order for easier adaption of this circuit to the continuous-variable setting, we choose an alternative, and equivalent, circuit formulation — one without the target state. We take this approach because the n-qubit model can be implemented in single mode of the harmonic oscillator rather than the two-mode model employed in [9]. The unitary operator associated with the oracle function changes slightly in this alternative circuit. This simpler algorithm without the target qubit is given in Figure 3.1.

In Chapter 2, we demonstrated that the oracle construct originally proposed by Deutsch–Jozsa and given in Eq. (2.79), may be equivalently expressed as

$$U_f : |x\rangle|y\rangle \mapsto (-1)^{f(x)}|x\rangle|y\rangle, \tag{3.1}$$

which is given in Eq. (2.82) and is re-expressed here for ease of reference. This construction yields a matrix representation for U_f as a permutation matrix, hence always

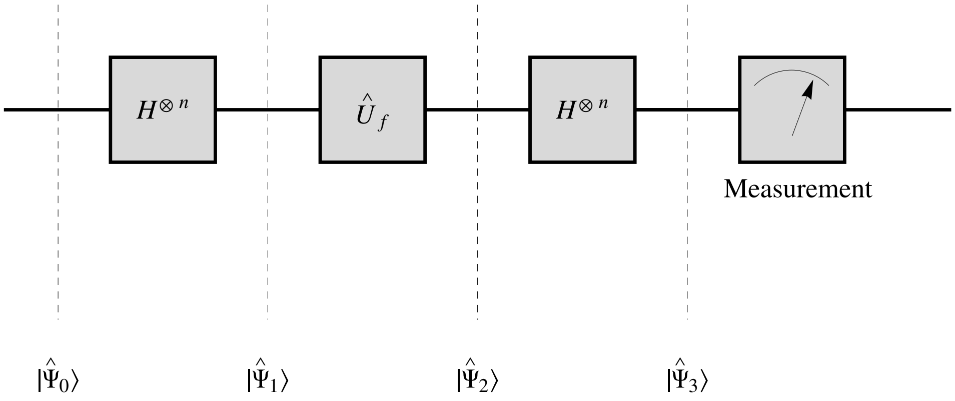

Figure 3.1: Alternative n-qubit discrete-variable circuit implementation of the Deutsch–Jozsa algorithm. Note the absence of the target qubit and the use of the operator $\hat{\mathsf{U}}_f$ defined in Eq. (3.3)

unitary [5]. The representation of the oracle given in Eq. (3.1) demonstrates that the target qubit remains unchanged and that it only provides a mechanism to encode the j^{th} bit of the unknown oracle string into the phase of the j^{th} computational basis state comprising the control state.

With respect to the ordered basis

$$B = \{|0\cdots 0\rangle|0\rangle, |0\cdots 0\rangle|1\rangle, \ldots, |1\cdots 1\rangle|0\rangle, |1\cdots 1\rangle|1\rangle\},$$

the unitary matrix U_f can be expressed in the following insightful form

$$\mathsf{U}_f = \begin{pmatrix} \mathsf{X}^{f(0\cdots 0)} & 0 & \cdots & 0 \\ 0 & \mathsf{X}^{f(0\cdots 1)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathsf{X}^{f(1\cdots 1)} \end{pmatrix}, \tag{3.2}$$

with X the 2×2 NOT operator in this case.

The operator U_f can also be expressed in the alternative ordered basis with $|+\rangle = \frac{1}{\sqrt{2}}(|1\rangle + |0\rangle)$, $|-\rangle = \frac{1}{\sqrt{2}}(|1\rangle - |0\rangle)$ and

$$B' = \{|0\cdots 0\rangle|-\rangle, |0\cdots 1\rangle|-\rangle, \ldots, |1\cdots 0\rangle|+\rangle, |1\cdots 1\rangle|+\rangle\},$$

as

$$\mathsf{U}_f = \begin{pmatrix} \hat{\mathsf{U}}_f & 0 \\ 0 & \mathbb{1} \end{pmatrix},$$

with $\mathbb{1}$ the $2^n \times 2^n$ identity operator. Furthermore the operator $\hat{\mathsf{U}}_f$ is expressed as the $2^n \times 2^n$ matrix

$$\hat{\mathsf{U}}_f = \begin{pmatrix} (-1)^{f(0\cdots 0)} & 0 & \cdots & 0 \\ 0 & (-1)^{f(0\cdots 1)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & (-1)^{f(1\cdots 1)} \end{pmatrix}, \tag{3.3}$$

and thus provides a reduced representation for U_f. It is clear from the matrix representation of $\hat{\mathsf{U}}_f$ that the oracle construct for $\hat{U}_f$ is simply

$$\hat{U}_f : |x\rangle \mapsto (-1)^{f(x)}|x\rangle, \tag{3.4}$$

where we note the absence of the target qubit that appears in Eq. (3.4).

It is apparent that the operator $\hat{\mathsf{U}}_f$ acts non-trivially only on a 2^n-dimensional subspace of the 2^{n+1}-dimensional space. This correspondence is clear in the following identity relation

$$\mathsf{U}_f = \Big(\hat{\mathsf{U}}_f \otimes |-\rangle\langle -|\Big) \oplus \Big(\mathbb{1}^{\otimes n} \otimes |+\rangle\langle +|\Big).$$

Thus, there is an equivalence between employing the oracle U_f, or its counterpart $\hat{\mathsf{U}}_f$. Oracle equivalence can also be understood because one of the oracles can be used to simulate the other [37]. We conclude that the construction employing both control and target qubits is not strictly necessary.

We now present a step-by-step analysis of the alternative circuit presented in Figure 3.1. We shall analyze the continuous-variable circuit in the same steps for cross reference and comparison. We use the tilde notation $|\tilde{\Psi}_j\rangle$ in order to emphasize that this analysis is of the algorithm presented in Figure 3.1. The n qubit input state of this circuit is a string of qubits prepared in $|\tilde{\Psi}_0\rangle = |0\cdots 0\rangle$. We place this state into an equal superposition of all computational basis states

$$\mathsf{H}^{\otimes n}|\tilde{\Psi}_0\rangle \mapsto |\tilde{\Psi}_1\rangle = 2^{-n/2} \sum_{x\in\{0,1\}^n} |x\rangle$$

for H the single qubit Hadamard operator.

Given the definition of the reduced operator $\hat{\mathsf{U}}_f$ defined in Eq. (3.3), its effect on the equal superposition of basis states expressed in the state $|\tilde{\Psi}_1\rangle$ is to encode the N-bit string z unitarily into the state $|\tilde{\Psi}_2\rangle$. We express this as

$$\hat{\mathsf{U}}_f|\tilde{\Psi}_1\rangle \mapsto |\tilde{\Psi}_2\rangle = 2^{-n/2} \begin{pmatrix} (-1)^{f(0\cdots 0)} \\ (-1)^{f(0\cdots 1)} \\ \vdots \\ (-1)^{f(1\cdots 1)} \end{pmatrix}, \tag{3.5}$$

which is a convenient representation. We shall show that this representation naturally extends to the continuous-variable setting.

Measurement proceeds by first undoing the superposition created during the state preparation step. The resultant state may be expressed as follows

$$|\tilde{\Psi}_3\rangle = \frac{1}{2^n} \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & -1 & \cdots & -1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & -1 & \cdots & 1 \end{pmatrix} \begin{pmatrix} (-1)^{f(0\cdots 0)} \\ (-1)^{f(0\cdots 1)} \\ \vdots \\ (-1)^{f(1\cdots 1)} \end{pmatrix}. \tag{3.6}$$

Eq. (3.6) allows us to see that all of the rows (and columns) of the operator $\mathsf{H}^{\otimes n}$ have an equal number of positive and negative ones except for the first row, which consists entirely of plus ones. It is this feature that permits the constant and balanced functions to be distinguished in a single measurement.

For the two constant cases, Eq. (3.6) results in the state $|\tilde{\Psi}_{3\mathrm{C}}\rangle = \pm(10\cdots0)'$ as only the first row does not result in amplitude cancellation of the 2^n constant amplitude components of the state $|\tilde{\Psi}_2\rangle$. Each of the balanced functions result in the amplitudes of the state $|\tilde{\Psi}_2\rangle$ having an equal number of positive and negative ones. This feature results in the first component of the state $|\tilde{\Psi}_3\rangle$ having zero amplitude for all the balanced functions. We express this result as $|\tilde{\Psi}_{3\mathrm{B}}\rangle = \pm(0\,x\cdots x)'$ where we use the symbol x to represent the non-zero value(s) that land on the other $N-1$ components depending on which of the $\binom{N}{N/2}$ balanced functions the oracle is set to.

For the final measurement step, we employ the projection operator [5] defined for $m \in \{0,1\}^N$ as follows

$$\mathsf{M}_m = |m\rangle\langle m|. \tag{3.7}$$

We are only concerned with the first component as discussed above, so for the constant cases we have

$$\Pr[m = (0\cdots0)] = \left\langle \tilde{\Psi}_{3\mathrm{C}} \middle| \mathsf{M}_{(0\cdots0)} \middle| \tilde{\Psi}_{3\mathrm{C}} \right\rangle = 1, \tag{3.8}$$

and for all balanced cases we have

$$\Pr[m = (0\cdots0)] = \left\langle \tilde{\Psi}_{3\mathrm{B}} \middle| \mathsf{M}_{(0\cdots0)} \middle| \tilde{\Psi}_{3\mathrm{B}} \right\rangle = 0 \tag{3.9}$$

as required.

We have established that the n-qubit discrete Deutsch–Jozsa algorithm gives the same result as the standard implementation. In the next section we formulate how to adapt the n-qubit model to the continuous-variable setting.

3.3 Single mode continuous-variable algorithm with orthogonal states

There are many potential models representing continuous-variable Deutsch–Jozsa algorithms. Both single-mode and multiple-mode models can be analyzed; each model can be studied using input states represented by different $\mathcal{L}^2(\mathbb{R})$ functions; also different oracle encoding schemes can be envisaged. In this section, we present the arguments for studying a particular one-mode model that offers insight into the key features of all continuous-variable models.

3.3.1 Continuous-variable model selection

Our strategy is to create a model of the continuous-variable Deutsch–Jozsa algorithm that naturally relates to the discrete case, employs reasonable resources, is generally applicable and is independent of particular basis functions. Discrete quantum information algorithms solving oracle decision problems follow a pattern: prepare the input state in one of n computational basis states $\in \mathbb{C}^n$; apply the Hadamard transformation to place the input state in an equal superposition of basis states; apply a unitary operator representing the oracle; undo the superposition by again applying the Hadamard operator, and perform a projective measurement of a particular basis state. The probability of the desired outcome is the square of the magnitude of the amplitude of the particular basis state.

In order for our continuous-variable model to relate naturally to the discrete case, we mimic this pattern using the position and momentum states of a single particle. We choose a single-mode continuous-variable model for the following reasons. First, we can apply the necessary algorithm features with minimal resources using the tools of quantum optics. Second, this single mode has infinite dimension, which is intrinsically more than sufficient to solve the problem. If it cannot be solved in a single-mode model, it is highly desirable to gain insight into the fundamental reasons why not. Third, Braunstein and Pati's techniques [18] employ a similar model that although it is a two-mode model employing a continuous-variable target state, the results of our single-mode model are directly applicable to it.

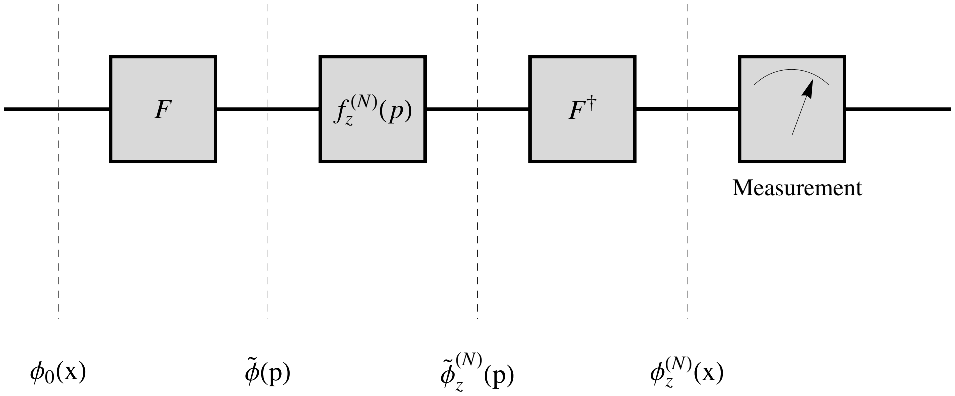

Figure 3.2: Single mode continuous-variable quantum circuit implementation of the Deutsch–Jozsa algorithm without the use of the target state. F is the continuous Fourier transform given by Eq. (3.12) and $f_z^{(N)}(p)$ is the oracle encoding function given by Eq. (3.20).

Our algorithm is the continuous-variable analogue of the alternative formulation of the discrete Deutsch–Jozsa algorithm presented in Figure 3.1. We present the continuous-variable version of this algorithm in Figure 3.2. In our model, we also choose to encode the strings $z \in \{0,1\}^N$ into an orthogonal basis of square-integrable complex-valued functions over the reals, namely $\mathcal{L}^2(\mathbb{R})$. Since information is of finite length and can be represented by N-bits, it is desirable that the encoded functions form a *compact* basis with $p \in [-P, P]$. We address the problem of encoding finite information into the non-orthogonal and non-compact Gaussian states in Chapter 4..

To deal with the problem of finite-length information strings, we encode the information into a finite region of the momentum domain. Information is represented by finite-length bit strings $z \in \{0,1\}^N$ with N the number of bits. These strings are encoded into a region of the momentum domain p extending from -P to P. A regular lattice of N discrete values of p are embedded in this domain such that the i^{th} bit of z, is assigned to p as follows: $p_{z_i} := (1 + 2i - N \pm 1)P/N$. The lattice is thus $\{p_{z_i}\}$ with spacing $2P/N$.

An orthogonal basis of top-hat functions is formed from the p_{z_i}, and the encoded momentum wave function is expressed as $\frac{1}{\sqrt{2P}} \sum_{i=0}^{N-1} (-1)^{z_i} \left|p_{z_i}\right\rangle$. The kets $\left|p_{z_i}\right\rangle$ are phase-modulated by their corresponding bit values, and with this phase modulation, each of the possible 2^N strings is uniquely represented. Note that the constant wave function is the top-hat function extending from $-P$ to P.

There is a translational invariance between computational basis states because each of the top-hat basis functions occupies an identically-sized region of momentum space and because the string z' is obtained from the string z by the translation $z' = z \oplus (z \oplus z')$. We can regard this finite basis as an infinite basis modulated by a top-hat function extending from $-P$ to P, which has the effect of truncating the allowed strings from an infinite domain to being from 0 to $N-1$.

We select $x, p \in \mathbb{R}$ and use the particle's position wave function, $\phi(x)$ to describe both the input and output states. The information representing the oracle is encoded into the momentum wave function $\tilde{\phi}(p)$. The position and momentum wave functions are Fourier transform pairs, and the relationship between the particle's position and its momentum is governed by Heisenberg's uncertainty principle. We selected x and p for illustrative purposes and use it throughout for convenience. We note however that it is unimportant in principle whether information is encoded in x or p — or indeed anything in between.

Regarding the use of the continuous Fourier transform we note that in the oracle setting, any mathematically-allowed transformation is indeed permitted for the oracle, but the non-oracle operations should be described in an implementable way. Thus, we choose a transformation that is amenable to current or planned technology. It is germane to our scheme that the canonical position and momentum bases are dual to each other in the sense that the Fourier transform is the analogue of the Hadamard transformation for qubit-based quantum computation. We thus exclude the use of the discrete Fourier or Hadamard transformations in this model.

Our approach to the creation of a single-mode continuous-variable Deutsch–Jozsa algorithm differs from that of Braunstein and Pati [18] in two key areas. First, our approach relies only on states that are elements in the Hilbert space congruent to $\mathcal{L}^2(\mathbb{R})$. Second, our approach does not require the use of the target state. Although their model employs the target state, it only serves to 'kick back' the phase of the oracle function and the conclusions from both models should be identical.

The basic algorithm elements required to implement the single-mode continuous-variable model are input state preparation; Fourier transform; oracle application and measurement. Our model requires that the momentum wave function be compact and the encoding be unambiguous. We define the following function, along with its Fourier dual, to help us achieve this end. For $P > 0$, the 'top hat' function

$$\begin{aligned}\sqcap(p; P, P_0) &= \langle p | \sqcap (P, P_0)\rangle \\ &= \frac{1}{\sqrt{2P}} \begin{cases} 1, & \text{if} \quad p \in [P_0 - P, P_0 + P] \\ 0, & \text{if} \quad p \notin [P_0 - P, P_0 + P] \end{cases} ,\end{aligned} \tag{3.10}$$

has the compact feature. We also note that

$$\lim_{P \to 0} |\sqcap (p; P, P_0)|^2 = \delta(p - P_0), \tag{3.11}$$

so the state $|\sqcap\rangle$ is, in some sense, a momentum eigenstate $|p = P_0\rangle$ in the limit $P \to 0$.

The state given by Eq. (3.10) is the Fourier transform of the desired input state. Here the Fourier transform acts as the continuous version of the discrete Hadamard transform (extending the Hadamard transformation to the continuous-variable case is not unique [21, 57]). For x the canonical position and p the canonical momentum, the Fourier transform maps a function $\phi(x)$ to its dual $\tilde{\phi}(p)$ according to [56]

$$F : \phi(x) \mapsto \tilde{\phi}(p), \tag{3.12}$$

such that

$$\tilde{\phi}(p) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \mathrm{d}x \, \mathrm{e}^{\mathrm{i}px} \phi(x),$$

and

$$\phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \mathrm{d}p \, \mathrm{e}^{-\mathrm{i}px} \tilde{\phi}(p).$$

Note that $\phi(x) = \langle x|\phi\rangle$; the momentum state $|p\rangle$ is the Fourier transform of $|x\rangle$, and $\tilde{\phi}(p) = \langle p|\phi\rangle$.

The inverse Fourier transform of the function $\mathrm{e}^{\mathrm{i}x_0} \sqcap (p; P, P_0)$ gives the input state, which is described by the sinc function

$$\begin{aligned} \phi(x) =& \langle x|\phi\rangle \\ =& \frac{\mathrm{e}^{\mathrm{i}P_0} \sin\left(P(x - x_0)\right)}{\sqrt{P\pi}(x - x_0)} \end{aligned} \tag{3.13}$$

centred at x_0. The limit of $|\phi(x)|^2$ as P goes to ∞ yields $\delta(x - x_0)$. The position eigenstate $|x = x_0\rangle$ is likewise formed in the limit $P \to \infty$.

Our model of the oracle encodes the N-bit string z into the compact domain $[-P, P]$. Since each of the N-bits comprising the string is represented unambiguously, the contiguous momentum pulses within $[-P, P]$ have width $\delta_p = 2P/N$, and for $j \in \{0, N-1\}$, the j^{th} momentum pulse is centred at position $-P + (j + 1/2)\delta_p$ and takes on value $(-1)^{z_j}$. The oracle can be repressed in Dirac notation as

$$\mathsf{U}_{\text{CV}} : |p_j\rangle \mapsto (-1)^{z_j} |p_j\rangle, \tag{3.14}$$

which is equivalent to the oracle construct for $\hat{\mathsf{U}}_f$ given in Eq. (3.4). We illustrate this oracle behaviour in Figure 3.3 for a particular $N = 4$ balanced case. Note that each of the unique balanced and constant strings form a set of orthogonal functions $\in \mathcal{L}^2(\mathbb{R})$ when encoded in this manner.

The final quantum element of our algorithm is measurement. We measure finite intervals of the probability density function $|\phi_i(x)|^2$, with the index i representing the

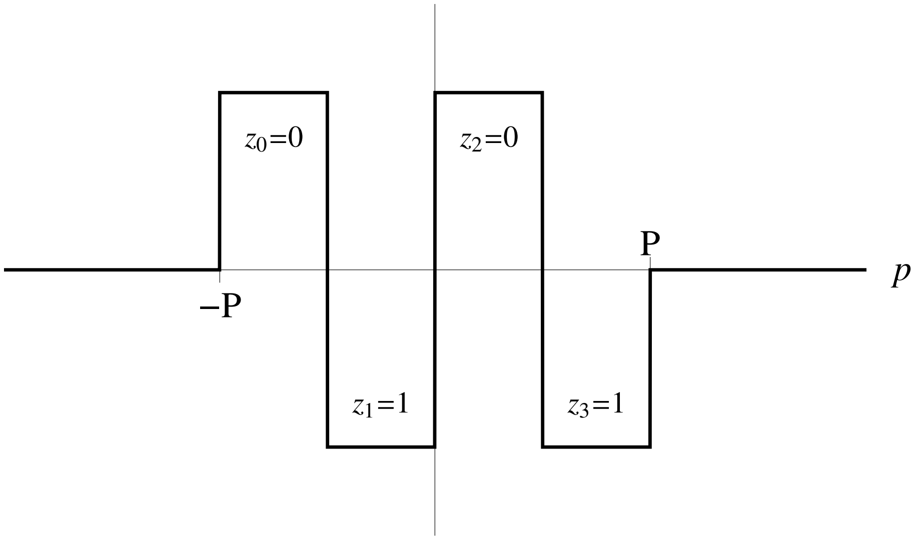

Figure 3.3: Illustration of the concept for encoding an N-bit string in a region of momentum extending from $-P$ to $+P$ using the $N = 4$, $z = 0101$ example. Note that each of the bits z_j are uniquely represented by individual pulses of height $\pm\frac{1}{\sqrt{2P}}$, width $\delta_p = 2P/N$ centred at $-P + (j + 1/2)\delta_p$.

encoded function. This is preferable to making precise measurements of real numbers. The continuous-variable analog of the projection operator given in Eq. (3.7) is defined as

$$E_a^b = \int_b^a |x\rangle\langle x|\mathrm{d}x, \tag{3.15}$$

and thus the probability of detecting the wave function $\phi_i(x)$ in the interval $[a, b]$ is

$$\Pr[\phi_i] = \int_b^a |\phi_i(x)|^2 dx. \tag{3.16}$$

With these concepts in order, we illustrate the four stages of our algorithm in Figure 3.4.

We begin with the position wave function centered at $x_0 = 0$ and illustrated in Figure 3.4(a). In Figure 3.4(b), we present the momentum wave function, which acts as the ‘substrate’ into which the N-bit strings are encoded. In Figure 3.4(c), the compact pulse function is encoded with the particular N-bit string $z = \{0 \cdots 01 \cdots 1\}$. Finally, the inverse Fourier transform of this ‘square wave’ is presented in Figure 3.4(d). Since

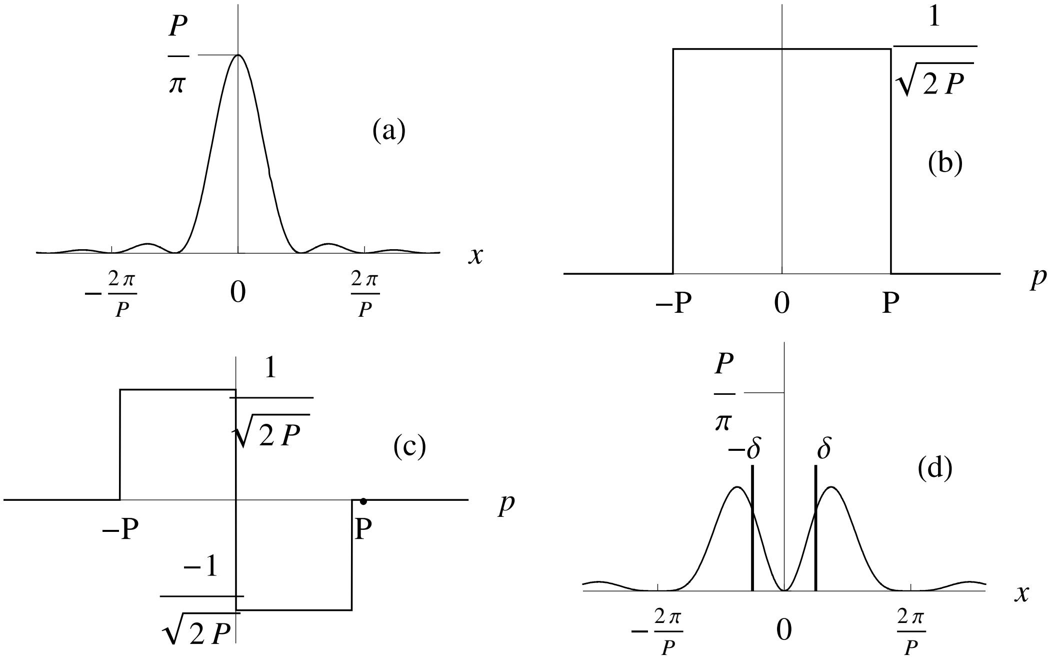

Figure 3.4: Overview of the four stages of our continuous-variable Deutsch–Jozsa algorithm: (a) The probability distribution of the input state. (b) The Fourier transform of the position state acts as the encoding ‘substrate’. (c) The N-bit string $z = \{0 \cdots 01 \cdots 1\}$ modulates this ‘substrate’. (d) The inverse Fourier transform of the encoded ‘square wave’ necessitates an optimal measurement ‘window’ parameterized by $\pm\delta$.

the inverse Fourier transforms of finite pulses in the momentum domain have infinite extent in the position domain, we need to limit the extent of our measurement to $\pm\delta$.

We briefly summarize the key features of our single-mode model of the Deutsch–Jozsa algorithm. Our continuous-variable Deutsch–Jozsa algorithm requires the parameters N, which represents the size of the problem, the momentum ‘length’ P over which the information is encoded, and the extent of the measurement window, δ. Since the Fourier transform maps compact states into non-compact states, we expect to determine relationships between these parameters.

3.3.2 Encoding Information into Orthogonal States

In this subsection, we analyze the single-mode continuous-variable Deutsch–Jozsa algorithm presented in Figure 3.2. This algorithm employs position states represented by sinc functions; employs the continuous Fourier transform; has oracle-encoded states represented by modulated top-hat functions, and employs the continuous-variable projection operator. We show that this model results in an uncertainty principle between the momentum 'length' P over which the N-bit string is encoded and the position 'length' δ corresponding to the optimum measurement window. This uncertainty implies that this model results in an algorithm that is necessarily probabilistic.

We argued previously that we need the Fourier transform of the input state to be the top hat function defined in Eq. (3.10). We add several conditions that do not take away from the generality of the solution. First, we want the top hat to have zero phase, which gives $x_0 = 0$ and to be centred at $P_0 = 0$. Second, we want the pulse to have extent $\pm P$. This gives us the simplest form of the sinc function for the initial state

$$\phi_0(x) = \frac{\sin(Px)}{\sqrt{\pi P}\, x}. \tag{3.17}$$

With reference to Figure 3.2, the next step in the algorithm is to perform the Fourier transform, which yields the top hat function with extent $\pm P$

$$\tilde{\phi}(p) = \frac{1}{\sqrt{2P}} \begin{cases} 1, & \text{if } p \in [-P, P] \\ 0, & \text{if } p \notin [-P, P]. \end{cases} \tag{3.18}$$

This function forms the raw substrate, which is 'modulated' by the individual N-bit strings z.

We perform encoding by partitioning the real numbers representing momentum into non-overlapping, contiguous and equal-sized bins. In this digital-to-analogue strategy, the width of each p-bin is $2P/N$, and

$$\sqcap_i^{(N)}(p) = \begin{cases} 1, & \frac{p}{P} \in \left[-\left(1 - 2\frac{N-1-i}{N}\right), -\left(1 - 2\frac{N-i}{N}\right)\right] \\ 0, & \text{otherwise.} \end{cases} \tag{3.19}$$

The oracle encodes the index z into the function f_z as follows:

$$f_z^{(N)}(p) = \sum_{i=0}^{N-1} (-1)^{z_i} \sqcap_i^{(N)}(p), \tag{3.20}$$

where the factor $(-1)^{z_i}$ serves to modulate the phase of the top hat function according to the bit value.

Example 1. *Consider the case $n = 2$; hence $N = 2^2 = 4$. As one case, the function corresponding to the four-bit string* 0011 *is*

$$f_{0011}^{(N)}(p) = \sqcap_0^{(N)}(p) + \sqcap_1^{(N)}(p) - \sqcap_2^{(N)}(p) - \sqcap_3^{(N)}(p). \tag{3.21}$$

The only two four-bit strings yielding constant functions would be 0000*, for which the function is identically unity over the whole domain $[-P, P]$, and* 1111*, for which the function is identically -1 over $[-P, P]$. Four cases are presented in Figure 3.5.*

In the limit that $N \to \infty$ with P fixed, $|\sqcap_i(p)|^2 \mapsto \delta(p - p_i)$ for p_i the midpoint of the i^{th} bin. The limit $N \to \infty$ thus gives a prescription for approaching a continuous variable representation where the z index seems to approach a continuum; however this limit yields a countable, rather than uncountable, set $\{z\}$, and the finite domain $[-P, P]$ has important ramifications on the nature of the functions corresponding to Fourier transforms of $\sqcap_i(p)$. We express the state after encoding as

$$\tilde{\phi}_z^{(N)}(p) = f_z^{(N)}(p)\tilde{\phi}(p), \tag{3.22}$$

where we observe the 'modulating' effect of the encoded string f_z on the momentum 'substrate' $\tilde{\phi}(p)$.

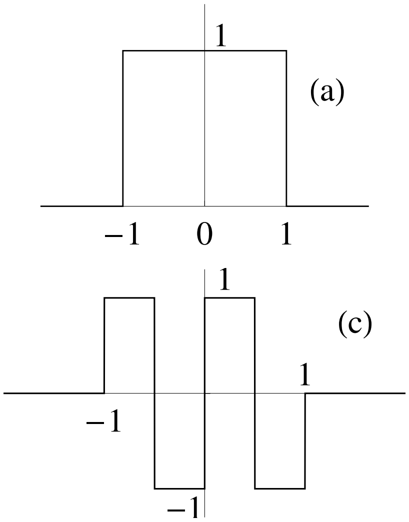

Figure 3.5: Encoded functions $f_z^{(4)}(p)$ (a) $z = 0000$, (b) $z = 0011$,(c) $z = 0101$, and (d) $z = 0110$.

We have the strings $z \in \{0, 1\}^N$ encoded into the momentum state (3.22). The next step is to take the inverse Fourier transform of this pulse train. For z_j the j^{th} bit of z, this is expressed as

$$\phi_z^{(N)}(x) = \frac{\sin(Px/N)}{\sqrt{P\pi}\, x} \sum_{j=1}^{N} (-1)^{z_j} \mathrm{e}^{\mathrm{i}\varphi_j(x)}, \tag{3.23}$$

where we have set $\varphi_j(x) = \left(\frac{(2j-1)-N}{N}\right) Px$. We see that the magnitude of an individual generalized sinc function, $\phi_z^{(N)}(x)$, is determined by a vector sum of N phasors, which is modulated by a particular N-bit string z.

The phasors, $\mathrm{e}^{\mathrm{i}\varphi_j(x)}$, are equiangular divisions of the angular interval

$$[-(N-1)Px/N, (N-1)Px/N],$$

and they exhibit the pairwise complex conjugate property $\varphi_j(x) = -\varphi_{N+1-j}(x)$. In Figure 3.6, we present the phasors for $N = 8$ with $Px = \pi/2$ and $Px = \pi/4$ to illustrate these features.

We note that the only functions with $\phi_z^{(N)}(0) \neq 0$ are the two constant sinc functions.

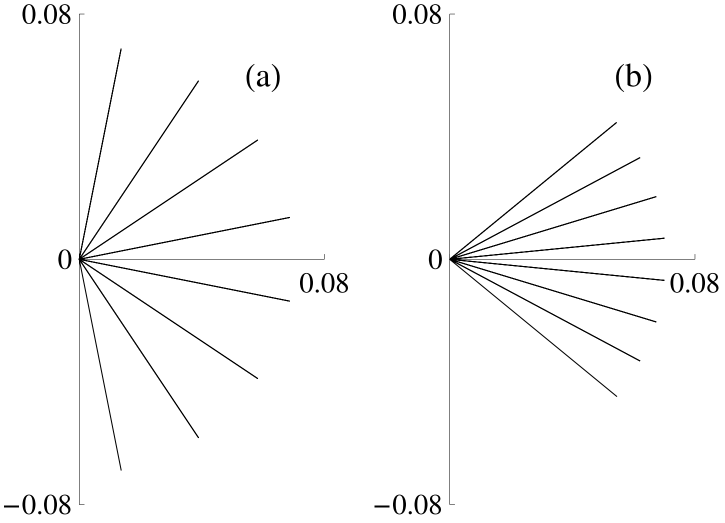

Figure 3.6: The phasors $\mathrm{e}^{\mathrm{i}\varphi_j x}$ defined in Eq. (3.23) for $N = 8$ (a) $Px = \pi/2$ phasors range between $\pm 7\pi/16$ in steps of $\pi/8$, and(b) $Px = \pi/4$ phasors range between $\pm 7\pi/32$ in steps of $\pi/16$.

This is clear given that $\sum_{j=1}^{N}(-1)^{z_j} = \pm N$ for the two constant cases, and $\sum_{j=1}^{N}(-1)^{z_j} = 0$ for all balanced cases. This feature of the set of $\binom{N}{N/2} + 2$ sinc functions represented by $\phi_z^{(N)}(x)$ implies that the strategy for measurement that distinguishes between the constant and balanced cases is to measure about $x_0 = 0$.

We measure the probability distribution in a small band around the position $x_0 = 0$. Due to the symmetry of the sinc functions about x_0, we employ the continuous-variable projection operator given in Eq. (3.15) and set $a = \delta$ and $b = -\delta$. The probability of detecting a particular wave function in the interval $\pm\delta$ is thus expressed as

$$\Pr\left[\phi_z^{(N)}(x)\right] = \int_{-\delta}^{\delta} \left|\phi_z^{(N)}(x)\right|^2 \mathrm{d}x. \tag{3.24}$$

We now need to determine the optimal value of measurement window δ that maximizes our ability to distinguish between the constant and balanced cases.

3.4 Bounding the success probability

We first determine which of the balanced functions dominates the region $\pm\delta$ and set bounds on the value of δ. We hypothesize that the dominant balanced functions are the balanced functions having the lowest 'frequency' content. This occurs when the first $N/2$ bits and the last $N/2$ bits have opposite values. For this pair of balanced functions, which we call the antisymmetric balanced ($\mathcal{AB}$) functions, we have

$$\mathcal{AB} \in \left\{ \underbrace{0\cdots0}_{N/2}\underbrace{1\cdots1}_{N/2}, \underbrace{1\cdots1}_{N/2}\underbrace{0\cdots0}_{N/2} \right\}. \tag{3.25}$$

Note that of the $\binom{N}{N/2}$ balanced functions, there are many that are also antisymmetric about the midpoint. However, we reserve the term $\mathcal{AB}$ for these two *lowest-order* antisymmetric balanced functions. Before proving the antisymmetric balanced function dominates all other balanced functions in the region $\pm\delta$, we illustrate the concept of frequency in the following example.

Example 2. *Again consider the case $P = 1$, $n = 2$; hence $N = 2^2 = 4$. As one case, the function corresponding to the four-bit string* 0011 *is*

$$\begin{aligned}\phi^{(4)}_{0011}(x) &= \frac{\sin(x/4)}{\sqrt{\pi}x}\left(e^{-\mathrm{i}\frac{3x}{4}} + e^{-\mathrm{i}\frac{x}{4}} - e^{\mathrm{i}\frac{x}{4}} - e^{\mathrm{i}\frac{3x}{4}}\right) \\ &= \frac{\mathrm{i}\,(\cos x - 1)}{\sqrt{\pi}x}\end{aligned} \tag{3.26}$$

This function corresponds to the $N = 4$ antisymmetric balanced function. The probability distributions for the four distinct $N = 4$ cases are presented in Figure 3.7. We clearly see that of the three balanced cases, the $N = 4$ antisymmetric balanced function has probability peaks closest to $x_0 = 0$ and thus has the lowest frequency content.

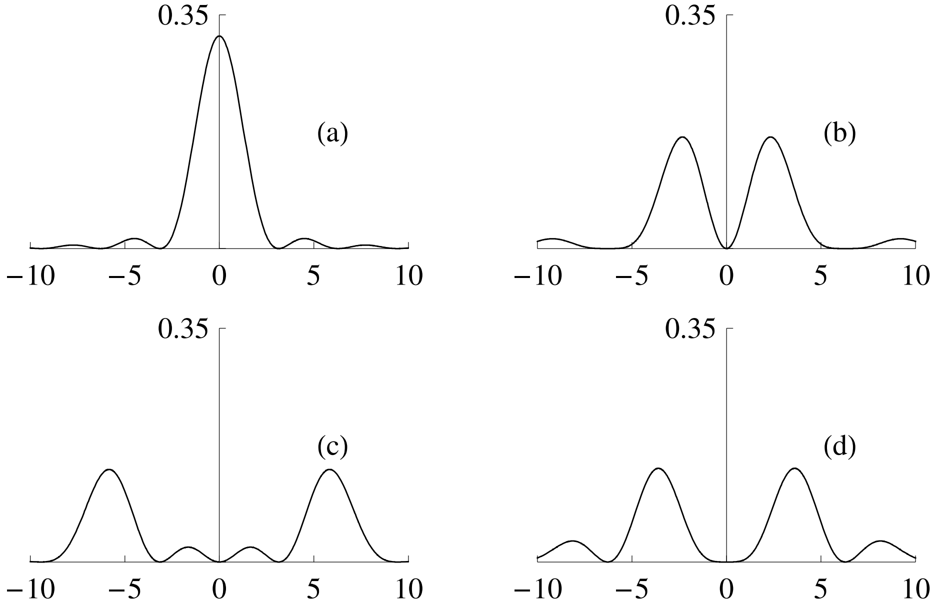

Figure 3.7: The probability distributions $|\phi_z^{(4)}(x)|^2$ for (a) $z = 0000$, (b) $z = 0011$, (c) $z = 0101$, and (d) $z = 0110$. We clearly see that of the three balanced cases (b) through (d), the $\mathcal{AB}$ function (b) has probability peaks closest to $x_0 = 0$.

Before stating and proving a Lemma on the dominance of the antisymmetric balanced, we set $m = N/2$ and define the quantity

$$S = \sum_{j=1}^{2m} g(j) \mathrm{e}^{\mathrm{i}\varphi_j}, \tag{3.27}$$

where $g : [2m] \mapsto \pm 1$ subject to the balanced condition $\sum_j g(j) = 0$. The sum defined in Eq. (3.27) is a convenient re-expression of the vector sum portion of Eq. (3.23). Here we have incorporated x into the definition of $\varphi_j = \left(\frac{(2j-1)-N}{N}\right) Px$. We seek to prove that, for all N, $|S|$ is maximized by the $\mathcal{AB}$ balanced function given in Eq. (3.25).

3.4.1 Proof of dominance of antisymmetric balanced function

In this sub-section we state and prove a lemma that the balanced function that dominates the measurement window is the antisymmetric balanced function.

Lemma 3.1. *For the region $|Px| < \pi$, Max $|S|$ occurs under the specific balanced conditions*

$$g(j) = \begin{cases} 1 & \textit{if } 1 \leq j \leq m \\ -1 & \textit{if } m+1 \leq j \leq 2m, \end{cases} \tag{3.28}$$

and

$$g(j) = \begin{cases} -1 & \textit{if } 1 \leq j \leq m \\ 1 & \textit{if } m+1 \leq j \leq 2m, \end{cases} \tag{3.29}$$

which we refer to as the asymmetric balanced functions ($\mathcal{AB}$) defined in Eq. (3.25).

Proof. Proof is done by induction on N. We require two base cases for $N = 2$ and $N = 4$. The base case for $N = 2$ is trivial since the only balanced cases are the two $\mathcal{AB}$ cases represented by the strings $\{01, 10\}$. The base case for $N = 4$ is a little more involved. We begin by labelling the angles and phasors as shown in Figure 3.8. There are $\binom{4}{2} = 6$ balanced cases, and we have to consider the strings $\{0011, 0101, 0110, 1100, 1010, 1001\}$. Since the latter three are complements of the first three, we have to consider only three vector sums.

With reference to Figure 3.8, we have $S_{\{0011\}} = \mathrm{e}^{-\mathrm{i}\varphi_1} + \mathrm{e}^{-\mathrm{i}\varphi_2} - \mathrm{e}^{\mathrm{i}\varphi_1} - \mathrm{e}^{\mathrm{i}\varphi_2}$. We simplify and express the result along with the three other cases as

$$\begin{aligned} S_1 &= S_{\{\{0011\},\{1100\}\}} = \pm 2\mathrm{i}\left(\sin\varphi_1 + \sin\varphi_2\right) \\ S_2 &= S_{\{\{0110\},\{1001\}\}} = \pm 2\left(\cos\varphi_1 - \cos\varphi_2\right) \\ S_3 &= S_{\{\{0101\},\{1010\}\}} = \pm 2\mathrm{i}\left(\sin\varphi_1 - \sin\varphi_2\right), \end{aligned}$$

where $\varphi_1 = -\frac{3xP}{4}$ and $\varphi_2 = -\frac{xP}{4}$. Clearly $|S_1| > |S_3|$. We use the trigonometric identities,

$$\begin{aligned} |S_1| &= 2\sin\left(\frac{\varphi_1 + \varphi_2}{2}\right)\cos\left(\frac{\varphi_1 - \varphi_2}{2}\right) \\ |S_2| &= 2\sin\left(\frac{\varphi_1 + \varphi_2}{2}\right)\sin\left(\frac{\varphi_1 - \varphi_2}{2}\right), \end{aligned} \tag{3.30}$$

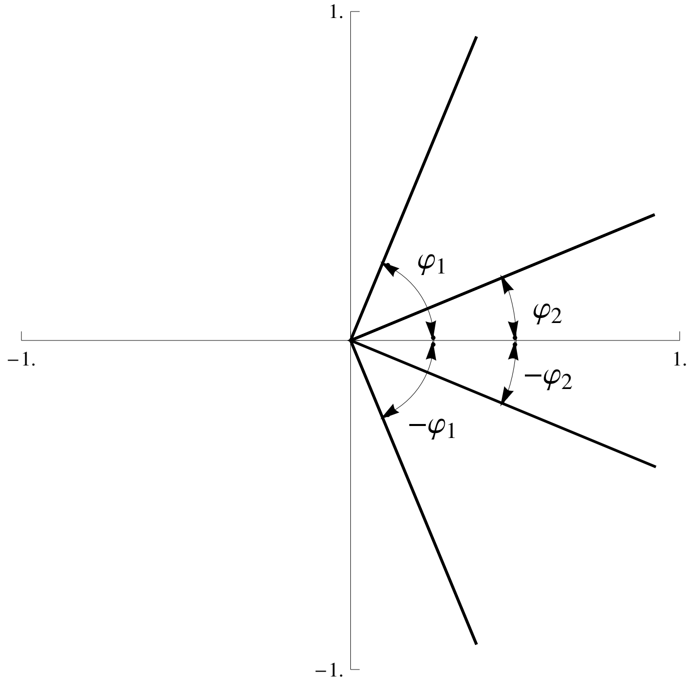

Figure 3.8: Definition of the phasor angles for the $N = 4$ base case for $P = 1$. Note that the effect of varying x over $[-\pi/2, \pi/2]$ simply focuses or expands the double angles $2\varphi_1$ and $2\varphi_2$ proportionally.

to establish the relationship between $|S_1|$ and $|S_2|$. For $0 \leq Px < \pi$, $\cos\left(\frac{Px}{4}\right) > \sin\left(\frac{Px}{4}\right)$, and therefore $|S_1| > |S_2|$. This proves that the theorem is true for the $N = 4$ base case.

For the inductive step for $N > 4$, we assume that $|S|$ is maximized for both the $N - 2$ and the $N - 4$ balanced strings. For an arbitrary balanced string of size N, we consider two cases. Case (i) assumes every pair is antisymmetric. By this we mean that $g(j) = -g(2m + 1 - j)$. If we remove any antisymmetric pair, we recover the $N - 2$ base case only if the string itself is the $\mathcal{AB}$ string. This part of the inductive step is expressed as $|S| \leq |S(\{l, 2m + 1 - l\})| + |S_2(w)|$, where $|S_2(w)|$ is maximized per the $N - 2$ base case.

Case (ii) assumes that Case (i) is false and therefore the arbitrary string must have two symmetric pairs for which $g(l) = g(2m+1-l) = +1$ and $g(k) = g(2m+1-k) = -1$. If we remove any two symmetric pairs, we recover the $N - 4$ base case only if the string itself is the $\mathcal{AB}$ string. This part of the inductive step is expressed as $|S| \leq |S(\{l, 2m+1-l, k, 2m+1-k\})| + |S_4(w)|$ where $S_4(w)$ is maximized per the $N = 4$ base case. Only when $|S(\{l, 2m+1-l, k, 2m+1-k\})|$ itself is maximized is equality achieved and the total sum maximized. This occurs for the $\mathcal{AB}$ strings. □

We have established that the antisymmetric-balanced functions dominate all other balanced function in the region $|Px| < \pi$. We use this result to determine the optimum value of δ.

3.4.2 Derivation of optimum measurement window

The ability to effectively distinguish between two random events is proportional to the separation of the individual probabilities of occurrence. Thus the optimum measurement window δ corresponds to the position where the difference between the constant and the antisymmetric-balanced probability distributions is maximized. We define the constant and antisymmetric-balanced probability distributions.

The two constant functions are defined by the strings

$$\mathcal{C} \in \left\{ \underbrace{0 \cdots 0}_{N}, \underbrace{1 \cdots 1}_{N} \right\}.$$

Taking the square of the magnitude of Eq. (3.23) given the constant functions defined above gives the constant probability distribution

$$\mathcal{P}_{\text{C}}(x) = \left| \phi^{(N)}_{z \in \mathcal{C}}(x) \right|^2 = \frac{\sin^2(Px)}{P\pi x^2}. \tag{3.31}$$

Similarly for the antisymmetric-balanced functions defined by Eq. (3.25), the probability distribution is expressed as

$$\mathcal{P}_{\text{AB}}(x) = \left| \phi^{(N)}_{z \in \mathcal{AB}}(x) \right|^2 = \frac{(\cos(Px) - 1)^2}{P\pi x^2}. \tag{3.32}$$

We select δ such that we get as much separation between the constant distribution and the antisymmetric-balanced distribution as possible.

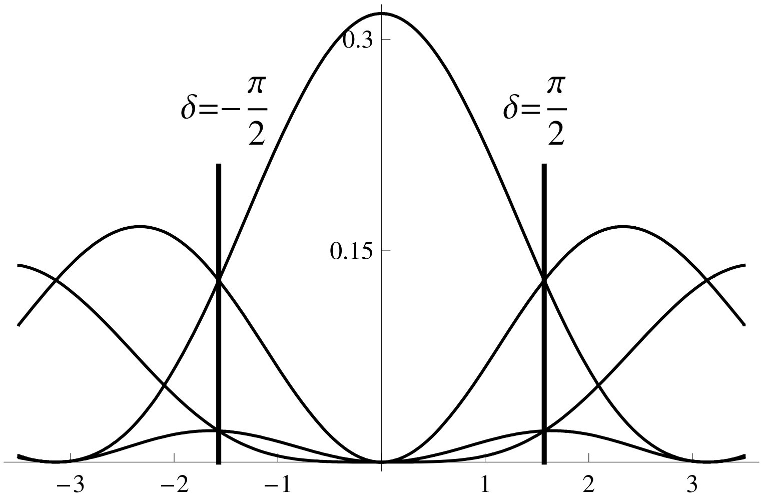

Figure 3.9: For $P = 1$, the optimal value of $\delta = \frac{\pi}{2}$. This graph shows that only the Constant and the Antisymmetric Balanced Functions significantly contribute to probability between $\pm\delta$.

When we make a measurement we are distinguishing between two events, the probabilities for which we define as follows

$$\mathrm{Pr}_{\mathrm{C}}(\delta) = \mathrm{Pr}\left[\left|\phi_z^{(N)}\right|^2 = \mathcal{P}_{\mathrm{C}}(x)\right] = E_{-\delta}^{\delta}\left(\mathcal{P}_{\mathrm{C}}(x)\right), \tag{3.33}$$

and

$$\mathrm{Pr}_{\mathrm{AB}}(\delta) = \mathrm{Pr}\left[\left|\phi_z^{(N)}\right|^2 = \mathcal{P}_{\mathrm{AB}}(x)\right] = E_{-\delta}^{\delta}\left(\mathcal{P}_{\mathrm{AB}}(x)\right). \tag{3.34}$$

The single-query success probability for the single-mode algorithm employing orthogonal states $\mathrm{Pr}_{\checkmark}^{\perp}$ is expressed in these terms as

$$\mathrm{Pr}_{\checkmark}^{\perp} = \max\left|\mathrm{Pr}_{\mathrm{C}}(\delta) - \mathrm{Pr}_{\mathrm{AB}}(\delta)\right|, \tag{3.35}$$

where δ is the value that maximizes the success probability.

We determine the optimum value of δ by maximizing the expression $|\mathrm{Pr}_{\mathrm{C}}(\delta) - \mathrm{Pr}_{\mathrm{AB}}(\delta)|$. It suffices to find the value of δ for which $\frac{\mathrm{d}}{\mathrm{d}\delta}|\mathrm{Pr}_{\mathrm{C}}(\delta) - \mathrm{Pr}_{\mathrm{AB}}(\delta)| = 0$, which may be expressed as

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}\delta}\left|\Pr_{\mathrm{C}}(\delta) - \Pr_{\mathrm{AB}}(\delta)\right| \\
&= \frac{\mathrm{d}}{\mathrm{d}\delta}\left|\int_{-\delta}^{\delta}\left(\frac{\sin^2(Px)}{P\pi x^2} - \frac{(\cos(Px)-1)^2}{P\pi x^2}\mathrm{d}x\right)\right| \\
&= \frac{\sin^2(P\delta)}{P\pi\delta^2} - \frac{(\cos(P\delta)-1)^2}{P\pi\delta^2} = 0.
\end{aligned} \tag{3.36}$$

This occurs where $\cos(P\delta) = \cos^2(P\delta)$ for $\delta \neq 0$, which gives a global maximum at $\delta = \frac{\pi}{2P}$. It is interesting to think of this result as an uncertainty relationship

$$P\delta = \frac{\pi}{2}. \tag{3.37}$$

For $P = 1$ and $N = 4$, the optimal measurement window of $\delta = \frac{\pi}{2}$ is plotted along with the four distinct probability distributions in Figure 3.9.

The probability distributions $\mathcal{P}_{\mathrm{C}}(x)$ and $\mathcal{P}_{\mathrm{AB}}(x)$ defined by Eqs. (3.31) and (3.32) respectively, are in $\mathcal{H}_2$, the Hilbert space of $\mathcal{L}^2(\mathbb{R})$ functions over the interval $(-\infty, \infty)$. This implies that since we are measuring over a finite interval, the continuous-variable Deutsch–Jozsa algorithm is necessarily probabilistic. Furthermore, we noted that P and δ are related by the uncertainty relation given in Eq. (3.37). This leads to the conclusion that even in the limit of the improper delta function $\delta(x - x_0)$, the continuous-variable Deutsch–Jozsa algorithm remains probabilistic.

This conclusion contradicts Braunstein and Pati's speed-up claim [18]. Their algorithmic improvement relies on unboundedness of the canonical momentum to provide perfect resolution for canonical position. As both domains must be bounded, our quantum-information Fourier limit applies, and their claimed speed-up is forbidden by the probabilistic nature of the protocol due to the position-momentum (equivalently time-bandwidth) tradeoff.

3.4.3 Numerical value of single-query success probability

We now determine numerical values of the probabilities determined in Eqs. (3.33) and (3.34). We can readily calculate the probability of detecting if the function is constant

$$\begin{aligned}\mathrm{Pr}_{\mathrm{C}} &= \int_{-\delta}^{\delta} \frac{\sin^2(Px)}{P\pi x^2}\mathrm{d}x \\ &= \frac{\cos(2\delta P) + 2\delta P\mathrm{Si}(2\delta P) - 1}{\delta P\pi}, \end{aligned} \tag{3.38}$$

where the sine integral is given by

$$\mathrm{Si}(z) = \int_0^z \frac{\sin t}{t}\mathrm{d}t. \tag{3.39}$$

If the function is the antisymmetric-balanced function, we have

$$\begin{aligned}\mathrm{Pr}_{\mathrm{AB}} &= \int_{-\delta}^{\delta} \frac{(\cos(Px) - 1)^2}{P\pi x^2}\mathrm{d}x \\ &= \frac{-8\sin^4\left(\frac{\delta P}{2}\right) + 4\delta P\mathrm{Si}(\delta P) - 2\delta P\mathrm{Si}(2\delta P)}{\delta P\pi}. \end{aligned} \tag{3.40}$$

Note that these probabilities depend only on the product $P\delta$.

For $P\delta = \pi/2$, the numerical values of these two probabilities are

$$\mathrm{Pr}_{\mathrm{C}} = \frac{2\left(\pi \,\mathrm{Si}\,(\pi) - 2\right)}{\pi^2} \approx 0.77, \tag{3.41}$$

and

$$\mathrm{Pr}_{\mathrm{AB}} = \frac{4\pi \,\mathrm{Si}\,(\pi/2) - 2\pi \,\mathrm{Si}\,(\pi) - 4}{\pi^2} \approx 0.16. \tag{3.42}$$

Thus the single-query success probability for the single-mode algorithm employing orthogonal states is

$$\mathrm{Pr}_{\checkmark}^{\perp} \approx 0.61. \tag{3.43}$$

Given this probabilistic nature of the continuous-variable Deutsch–Jozsa algorithm, we need to develop a strategy to bound the error probability. We employ the technique sometimes called probability amplification [20].

3.4.4 Success probability after multiple repetitions

Our strategy is to make m repetitions of the continuous-variable Deutsch–Jozsa algorithm, where we assume that the oracle is set to the same function for each of the repetitions. Each repetition ends with a measurement. From this sequence of measurements we want to determine whether the unknown function is balanced or constant with high probability.

Lemma 3.2. *An error of $O(e^{-m})$ can be achieved by making $O(m)$ repetitions of the* CV *DJ algorithm.*

Proof. We adopt the convention that when we make a query to the continuous-variable Deutsch–Jozsa algorithm, we either detect something (algorithm returns a 1), or we do not (algorithm returns a 0). We can thus treat multiple queries as a sequence of Bernoulli trials [58]. We assume that we have set our measurement limits to the optimal $\pm\delta$. The two events we are trying to uncover are the constant cases where, for ease of calculation we set the probability of detecting something is $\mathrm{Pr}_{\mathrm{C}} \geq 3/4$, and the balanced cases where the probability detecting something is $\mathrm{Pr}_{\mathrm{B}} \leq 1/4$. Note that we have set the probabilities to these rational numbers for illustrative purposes and to simplify the calculation. We can make this arbitrary setting, and we obtain the same result as long as the probabilities are bound from 1/2 by a constant.

If each measurement is based on an independent preparation of the state $\phi_0(x)$, then each of the queries are independent. After a series of m queries, we can use the Chernoff bounds of the binomial distribution to amplify the success probability [58, 59]. The simplest (but somewhat weak) Chernoff bound on the lower tail is given by [59] as

$$\mathrm{Pr}[X < (1-\epsilon)\mu)] < \mathrm{e}^{-\frac{\mu\epsilon^2}{2}}, \tag{3.44}$$

and on the upper tail as

$$\Pr[X > (1+\epsilon)\mu)] < \mathrm{e}^{-\frac{\mu\epsilon^2}{4}}, \tag{3.45}$$

where μ is the expected mean of the resulting binomial distributions after m queries, and ϵ is the relative distance from the respective means.

First, we bound the lower tail corresponding to the distribution of the constant case for which we have $\mu = m\mathcal{P}_{\mathrm{C}}$. Here we set $\epsilon = \frac{1}{3}$, which expresses the probability for the value being less than halfway between the two means as $\Pr[X < (m/2)] < \mathrm{e}^{-\frac{m}{24}}$. Similarly, we bound the upper tail for the balanced case for which we have $\mu = m\mathcal{P}_{\mathrm{B}}$. Here we set $\epsilon = 1$, which expresses the probability for the value being greater than half way between the two means as $\Pr[X > (m/2)] < \mathrm{e}^{-\frac{m}{16}}$. Clearly the success is worse for the lower tail allowing us to bound the success probability of the continuous-variable Deutsch–Jozsa algorithm after m queries as

$$\Pr[\text{Success}] \geq 1 - \mathrm{e}^{-\frac{m}{24}}. \tag{3.46}$$

This gives an error probability that is $O\left(\mathrm{e}^{-m}\right)$ as required. □

We note that this is of the same order as the exponentially good success probability we have for the classical probabilistic approach given by Eq. (2.77). Also note that this query complexity is independent of the value of N. We have made no attempt to obtain a tighter bound preferring to show only that we can achieve the same exponentially small error probability using the continuous-variable Deutsch–Jozsa algorithm as the classical probabilistic approach to solving the Deutsch–Jozsa problem using a number of queries that is of the same order as the classical probabilistic approach. The continuous-variable Deutsch–Jozsa algorithm is in this sense no worse than the classical probabilistic approach.

3.5 Summary

In this chapter, we have presented a single-mode continuous-variable model of the Deutsch–Jozsa algorithm. Our model employs logical states that are in the Hilbert space congruent to the space of $\mathcal{L}^2(\mathbb{R})$ functions. Our proof that the antisymmetric-balanced function given by expression (3.25) is the dominant balanced function in the measurement window enables us to neatly bound the single-query success probability $\text{Pr}_{\checkmark}^{\perp} \approx 0.61$. This demonstrates that the algorithm is necessarily probabilistic, and cannot provide the exponential speed-up of its discrete quantum counterpart. This is contrary to results published in the literature.

The probabilistic nature of this single-mode continuous-variable quantum algorithm derives from the nature of the continuous Fourier transform. The lack of speed-up results from an uncertainty principle between the ability to encode perfectly in a continuous representation and the subsequent inability to measure perfectly in the Fourier-dual representation. This uncertainty relationship is manifest in Eq. (3.37), which relates P, the encoding extent, to δ, the measurement extent.

Our result is directly analogous to the time-bandwidth uncertainty theorem of signal processing. Furthermore our result does not have to be limited to the Deutsch–Jozsa algorithm. We argue that our model is generally applicable and that the algorithm relies only on the orthogonality of basis functions, compactness of encoding requirements, and the time-bandwidth uncertainty relation.

In our analysis so far, we intentionally have not employed the Gaussian states so readily created in quantum optics experiments. We chose to select the sinc-pulse Fourier transform pair for our model because of our interest in unambiguous encoding of the unknown function. Since the Fourier transform of a Gaussian state is itself a Gaussian state and both span infinite domains, the encoding strategy may be different, but we can see that the result is also probabilistic. This is again due to the necessity of finite measurement extent. We study encoding finite information in Gaussian states in the next chapter.

Chapter 4

CONTINUOUS-VARIABLE QUANTUM COMPUTATION WITH COHERENT STATES

4.1 Introduction

The code-state formalism developed in Chapter 3 and demonstrated in [20] enables information to be embedded in the momentum domain such that each bit z_i of the N-bit string z is represented by a unique code state, and each code-state occupies an identically-sized region. This approach of using a finite region of the momentum domain having extent $2P$ to act as an encoding substrate leads to the algorithm being probabilistic having single-query success probability $\mathrm{Pr}_{\checkmark}^{\perp} < 1.0$. The probabilistic nature of this procedure is because an orthogonal wave function that is bounded in the momentum domain is unbounded in its Fourier dual position domain [20], and since measurement must be performed over a finite window, the measurement is not exact. In this Chapter and demonstrated in [24], we extend our code-state formalism to include encoding in a Gaussian wave function defined over a finite domain.

The appeal of using the Gaussian wave functions as the next step in our code-state formalism is due to the ability to create coherent states in the laboratory. The optical field modes created in the laboratory are analogous to harmonic oscillators and as demonstrated in Sec 2.3.1, the coherent wave functions of the harmonic oscillator may be represented by Gaussian wave functions. The unbounded extent of the Gaussian wave functions does not naturally fit with the finite length of the information strings being processed, and we deal with this problem by defining the encoding wave function over a finite domain similar to the approach taken in [60].

We modify the approach of embedding information in momentum states having identically sized-momentum regions representing bits by shaping the overall top-hat function with a Gaussian having width set by its standard deviation σ. The Gaussian is truncated for $|p| > P$, which has the effect that the computational basis states, although still orthogonal, are no longer translations of each other but are more complicated Gaussian-modulated basis states. At first glance, this latter feature would appear to provide a disadvantage, but using the mathematical properties of the error function, we prove that the single-query success probability for the Gaussian case $\mathrm{Pr}^{\sharp}_{\checkmark} > \mathrm{Pr}^{\perp}_{\checkmark}$ is greater than the single-query success probability for the orthogonal case.

We demonstrate that this result is enabled by the extra degree of freedom manifest in the spread of the Gaussian wave function. Tuning the available parameters of encoding width, the spread of the Gaussian wave function and the width of the measurement window results in a more efficiently encoded momentum wave function leading to the improved single-query success probability. Note that the use of the top-hat basis to encode information into a single harmonic oscillator is different than the approach used in [21], where information is encoded into a collection of harmonic oscillators.

In the analysis presented in [60], the authors use Gaussian states defined on a finite domain as input states to a single-mode harmonic oscillator implementing an algorithm for metrology parameter estimation. For a fixed parameter, their procedure reduces to the Deutsch-Jozsa algorithm, which is demonstrated to be a probabilistic procedure although a specific success probability is not given. Our approach of using normalized, orthogonal code-states embedded in a single Gaussian wave function is complementary to their approach. We show that similar probabilistic properties hold for our work and also calculate a success probability that is demonstrably better than orthogonal encoding.

The chapter is organized as follows. In Sec. 4.2, we define the coherent input states to the algorithm and give a theorem stating that the single-query success probability improves for coherent states over orthogonal states. In Sec. 4.3, we prove this theorem extending the techniques that we developed for the algorithm employing orthogonal states.

4.2 Single mode continuous-variable algorithm with Gaussian states

The sinc function employed as algorithm input to the orthogonal version of the algorithm presented in Chapter 3 and in [20] cannot be readily created in the laboratory. Here we are inspired by the ability to create and manipulate physical states of light in the laboratory using the tools of quantum optics. In particular, we employ coherent states of the harmonic oscillator discussed in Sec. 2.3.1.

With reference to continuous-variable quantum circuit given in Figure 3.2 in Chapter 3, we define the wave functions $\phi_0(x)$, $\tilde{\phi}(p)$, $\tilde{\phi}_z^{(N)}(p)$ and $\phi_z^{(N)}(x)$ for each of the steps of the algorithm based on coherent states. In each case, x and p are the continuous position and momentum variables, and z is the N-bit string encoded into the oracle.

In the position representation, the coherent state of laser light given in Eq. (2.43) and repeated here is

$$\phi(x;\alpha) = \langle x|\alpha\rangle = \pi^{-1/4} \exp\left[-\frac{(x-x_0)^2}{2} + \mathrm{i}p_0 x - \frac{\mathrm{i}p_0 x_0}{2}\right], \tag{4.1}$$

where $x_0 = p_0 = 0$ corresponds to the vacuum state. From the perspective of our quantum algorithm employing coherent states, the displaced vacuum behaves no differently than the vacuum itself. The constants x_0 and p_0 shift the distributions and change the region of the momentum domain where information is encoded and the region of the position domain where measurement is performed. However, they have no effect on the important algorithm parameters of the encoding length P and the width of the measurement window δ. We therefore chose to set $x_0 = 0$ and $p_0 =$ and use the position representation of the vacuum,

$$\langle x|0\rangle = \frac{e^{-\frac{x^2}{2}}}{\sqrt[4]{\pi}}, \tag{4.2}$$

as the starting-point state for defining the input state.

We employ the squeezed vacuum as the input state because of our interest in exploring the degree of squeezing as a parameter in finding the optimal single-query success probability. In section 2.3.1, we discussed the coherent states of the harmonic oscillator and presented the squeezing operator $\hat{S}(\zeta)$ in Eq. (2.47) terms of the squeezing parameter $\sigma = e^{-\zeta}$. We employ the squeezed vacuum as the input state to our algorithm, which we describe in the position representation as

$$\phi_0(x;\sigma) = \langle x|\hat{S}(\zeta)|0\rangle = \frac{e^{-\frac{x^2}{2\sigma^2}}}{\sqrt[4]{\pi}\sqrt{\sigma}}. \tag{4.3}$$

The subscript zero identifies this state as the algorithm input state represented by the leftmost vertical line in Figure 3.2. We now state a theorem that this algorithm with Gaussian input state has improved single-query success probability over the algorithm employing orthogonal state input.

Theorem 1. *Using the single-mode quantum circuit given in Figure 3.2 with the coherent state given by Eq. (4.3) as algorithm input and employing sharp information cutoff, the single-query success probability* $\mathrm{Pr}^{\sharp}_{\checkmark} > \mathrm{Pr}^{\perp}_{\checkmark}$ *is greater than the single-query success probability* $Pr^{\perp}_{\checkmark}$ *obtained using the orthogonal state given by Eq. (3.17) as algorithm input.*

In the following subsection, we set up the conditions for proving this theorem by establishing equations for the encoded position wave function $\phi_z^{(N)}(x)$.

4.2.1 Encoding information into Gaussian states

With reference to Figure 3.2, the first step of the algorithm is to take the Fourier transform of the input state $\phi_0(x;\sigma)$ giving

$$\tilde{\phi}_1(p\,;\sigma) = \frac{e^{-\frac{1}{2}p^2\sigma^2}\sqrt{\sigma}}{\sqrt[4]{\pi}}. \tag{4.4}$$

The next step has the oracle U_f modulate the momentum Gaussian with the pulse train that represents the encoding of the N-bit string z.

The modulated momentum wave function is

$$\tilde{\phi}_2(p) = \tilde{\phi}_z^{(N)}(p\,;\sigma,P) = \eta f_z^{(N)}(p;P)\tilde{\phi}_1(p\,;\sigma), \tag{4.5}$$

where we have labelled the state with all relevant parameters. Descriptions of the elements of this equation follow. The modulating square-wave encoded with the N-bit string z is

$$f_z^{(N)}(p;P) = \sum_{i=0}^{N-1}(-1)^{z_i}\sqcap_i^{(N)}(p;P), \tag{4.6}$$

where the definition of the momentum bins given in [20] is repeated here as

$$\sqcap_i^{(N)}(p;P) = \begin{cases} 1, & \frac{p}{P} \in \left[-\left(1-2\frac{N-1-i}{N}\right), -\left(1-2\frac{N-i}{N}\right)\right] \\ 0, & \text{otherwise.} \end{cases} \tag{4.7}$$

Note that the modulating function has the effect of chopping off the tails of the momentum Gaussian outside $\pm P$ thus truncating the Hilbert space.

The normalization factor, η, of the chopped distribution is calculated as

$$\int_{-\infty}^{\infty}\left|\tilde{\phi}_z^{(N)}(p;\sigma,P)\right|^2 \mathrm{d}p = \mathrm{erf}(P\sigma), \tag{4.8}$$

where the error function is $\mathrm{erf}(w) = \frac{2}{\sqrt{\pi}}\int_0^w e^{-t^2}dt$, and

$$\eta = 1/\sqrt{\mathrm{erf}(P\sigma)}. \tag{4.9}$$

The penultimate step is to take the inverse Fourier transform of this encoded momentum state.

The encoded position state is thus expressed as

$$\phi_3(x) = \phi_z^{(N)}(x;\sigma,P) = \frac{\eta\,\mathrm{e}^{-\frac{x^2}{2\sigma^2}}}{2\sqrt[4]{\pi}\sqrt{\sigma}}\,M_z^{(N)}(x;\sigma,P). \tag{4.10}$$

The effect of the encoded information is completely captured in the position modulating term

$$M_z^{(N)}(x;\sigma,P) = \sum_{j=1}^{N}(-1)^{z_j}\left[\operatorname{erf}\left(\frac{\vartheta_j\sigma^2+\mathrm{i}x}{\sqrt{2}\sigma}\right)-\operatorname{erf}\left(\frac{\vartheta_{j-1}\sigma^2+\mathrm{i}x}{\sqrt{2}\sigma}\right)\right], \tag{4.11}$$

where

$$\vartheta_j = P\left(\frac{2j-N}{N}\right). \tag{4.12}$$

Our approach is to determine which balanced functions dominate all other balanced functions in the measurement window. In the orthogonal case we had to define only one balanced function. Due to the symmetric shaping of the Gaussian envelope, we must take into consideration the potential for information at the extremes of the encoding region becoming attenuated.

We define three pairs of N-bit strings for consideration: the antisymmetric balanced $\mathcal{AB}$ strings, the symmetric balanced $\mathcal{SB}$ strings and the constant $\mathcal{C}$ strings as

$$\mathcal{AB} = \left\{\underbrace{0\cdots0}_{N/2}\underbrace{1\cdots1}_{N/2},\underbrace{1\cdots1}_{N/2}\underbrace{0\cdots0}_{N/2}\right\}, \tag{4.13}$$

$$\mathcal{SB} = \left\{\underbrace{0\cdots0}_{N/4}\underbrace{1\cdots1}_{N/2}\underbrace{0\cdots0}_{N/4},\underbrace{1\cdots1}_{N/4}\underbrace{0\cdots0}_{N/2}\underbrace{1\cdots1}_{N/4}\right\}, \tag{4.14}$$

$$\mathcal{C} = \left\{\underbrace{0\cdots0}_{N},\underbrace{1\cdots1}_{N}\right\}. \tag{4.15}$$

Note that the constant strings have zero bit transitions, the antisymmetric balanced strings have one bit transition, and the symmetric balanced strings have two transitions. All other balanced strings have two or greater transitions.

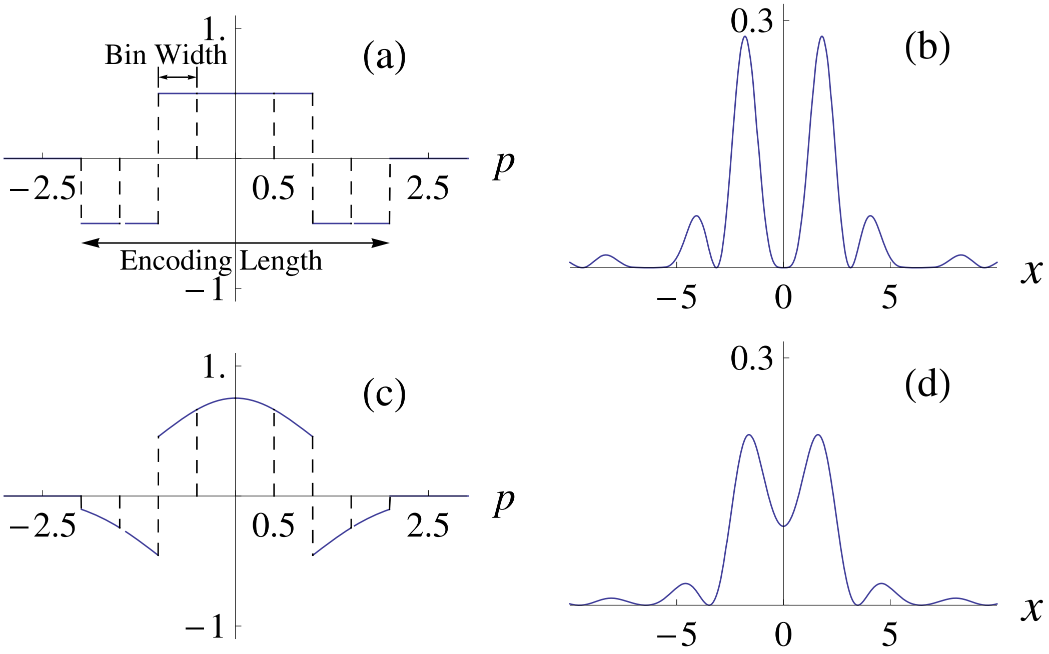

Figure 4.1: For $N = 8$, the symmetric balanced string 11000011 modulates the momentum Gaussian in (a) and (c), and the magnitude of the corresponding position wave functions are presented in (b) and (d). If the momentum Gaussian is too narrow with respect to the length of the encoded information as is the case depicted in (c), information is lost.

The leftmost and rightmost bits of the string z are the most likely to be affected by the Gaussian shaping function. In the symmetric balanced case given in Eq. (4.14), the attenuation of the outermost bits results in loss of the balanced character of the function. In Figure 4.1, we demonstrate that the failure to match the Gaussian width to the encoded information is analogous to being communications bandwidth limited. For the case where the momentum Gaussian is very wide, the situation approaches the orthogonal case depicted in Figure 4.1(a) and Figure 4.1(b). However, if the momentum Gaussian is too narrow, information at the extremes becomes attenuated and information is lost as depicted in Figure 4.1(c) and Figure 4.1(d).

It is insightful to analyze the modulating term given by Eq. (4.11) for $x = 0$, which we express as

$$M_z^{(N)}(0;\sigma,P) = \sum_{j=1}^{N}(-1)^{z_j}\left[\text{erf}\left(\frac{\vartheta_j\sigma}{\sqrt{2}}\right) - \text{erf}\left(\frac{\vartheta_{j-1}\sigma}{\sqrt{2}}\right)\right]. \tag{4.16}$$

We use the anti-symmetric property of the error function $\text{erf}(a) = -\text{erf}(-a)$, and the property that $\text{erf}(0) = 0$. We also use the facts that $\vartheta_N = P$, $\vartheta_N/2 = 0$ and $\vartheta_j = -\vartheta_{N-j}$ for $j = 0, 1, \ldots, N$ in determining the following results.

For the constant case, all the terms cancel except the first and last, and we have

$$M_{z\in\mathcal{C}}^{(N)}(0;\sigma,P) = \pm\left[\text{erf}\left(\frac{\vartheta_N\sigma}{\sqrt{2}}\right) - \text{erf}\left(\frac{\vartheta_0\sigma}{\sqrt{2}}\right)\right] = \pm 2\,\text{erf}\left(\frac{P\sigma}{\sqrt{2}}\right). \tag{4.17}$$

For the antisymmetric balanced case, all the terms cancel and we have

$$M_{z\in\mathcal{AB}}^{(N)}(0;\sigma,P) = \pm\left[\text{erf}\left(\frac{\vartheta_N\sigma}{\sqrt{2}}\right) + \text{erf}\left(\frac{\vartheta_0\sigma}{\sqrt{2}}\right)\right] = 0. \tag{4.18}$$

For the symmetric case we have

$$M_{z\in\mathcal{SB}}^{(N)}(0;\sigma,P) = \pm\left[2\,\text{erf}\left(\frac{P\sigma}{\sqrt{2}}\right) - 4\,\text{erf}\left(\frac{P\sigma}{2\sqrt{2}}\right)\right]. \tag{4.19}$$

Here the sum is non-zero except for in the limiting case where $P\sigma \to 0$.

We see from the results of Eqs. (4.18) and (4.19) that there are different classes of balanced functions since some balanced functions are only non-zero at $x = 0$ in the limit as σ goes to zero, and some balanced functions are zero at $x = 0$ for all values of σ. We will exploit this feature in bounding the single-query success probability. In the next section we first determine the specific balanced functions that dominate the measurement region for different values of σ. We then use these functions to bound the single query success probability.

4.3 Bounding the success probability with Gaussian States

For the algorithm using orthogonal states, we bound the single-query success probability by calculating the difference between the probability of detecting a constant function and the probability of detecting the balanced function that dominates the measurement window. In the orthogonal case, the dominant function is the antisymmetric balanced function. For the algorithm using Gaussian states, the extra degree of freedom offered by the standard deviation σ results in a second balanced function dominating the window for some values of σ. In this section, we prove that the dominant balanced function is either the antisymmetric balanced function given by Eq. (4.13) or the symmetric balanced function given by Eq. (4.14) depending on the value of σ.

4.3.1 Proof of dominance of symmetric and antisymmetric functions

We state and prove in three Lemmas that the symmetric and the antisymmetric balanced functions maximize the magnitude of $\left|M_z^{(N)}(x;\sigma,1)\right|$ given by Eq. (4.10) subject to the balanced condition $\sum_{i=1}^{N}(-1)^{z_i} = 0$. In Lemma 4.1, we demonstrate that in the limit $\sigma \to 0$, the encoded Gaussian position wave function given by Eq. 4.10 reduces to the encoded synch position wave function given in Eq. (3.23). In Lemma 4.2, we demonstrate that for $x = 0$ and $\sigma > 0$, the symmetric balanced function is dominant. In Lemma 4.3, we demonstrate for $0 < x \leq \pi$ and $\sigma > 0$ that either the symmetric balanced function or the symmetric balanced function are dominant and derive an expression for the cross-over point.

Lemma 4.1. *For the region $|x| \leq \pi$ with $\sigma = 0$ and subject to the balanced condition $\sum_{j=1}^{N}(-1)^{z_j} = 0$, $\max\left|\phi_z^{(N)}(x;\sigma,P)\right|$ occurs for $z \in \mathcal{AB}$.*

Proof. We prove this Lemma by showing that, in the limiting case where $\sigma \to 0$, the encoded position wave function given in Eq. (4.10) becomes the same as the encoded orthogonal wave function analyzed in [20]. In the orthogonal case, we proved that the antisymmetric balanced function dominates all other balanced wave functions in the region of interest.

We begin by defining the quantity

$$A_k(x,\sigma) = \mathrm{erf}\left(\frac{\frac{2Pk}{N}\sigma^2 - \mathrm{i}x}{\sqrt{2}\sigma}\right) - \mathrm{erf}\left(\frac{\frac{2P(k-1)}{N}\sigma^2 - \mathrm{i}x}{\sqrt{2}\sigma}\right) \tag{4.20}$$

for $k = 1, ..., N/2$. We use this term here and in later Lemmas. For ease of understanding the antisymmetric features of this term, we have elected to change the counting variable in the term ϑ_j given in Eq. (4.12) from j to k, where $j = k + N/2$. We express the k^{th} term of the encoded position wave function given by Eq. (4.10) in terms of this quantity as

$$\phi_z^{(k)}(x;\sigma,P) = \pm\frac{\eta\,\mathrm{e}^{-\frac{x^2}{2\sigma^2}}}{2\sqrt[4]{\pi}\sqrt{\sigma}}A_k(x,\sigma) \tag{4.21}$$

where the ± represents the effect of the bit $(-1)^{z_k}$. We represent this quantity as the phasor

$$\phi_z^{(k)}(x;\sigma,P) = \pm R_k(x;\sigma,P)\exp\left[\mathrm{i}\varphi_k(x;\sigma,P)\right], \tag{4.22}$$

to align with the description of the orthogonal case. The phasor magnitude is expressed

$$R_k(x;\sigma,P) = \left|\frac{\eta\,\mathrm{e}^{-\frac{x^2}{2\sigma^2}}}{2\sqrt[4]{\pi}\sqrt{\sigma}}A_k\right|, \tag{4.23}$$

and the argument is

$$\varphi_k(x;\sigma,P) = \arctan\left(\frac{\mathrm{Im}\left[A_k\right]}{\mathrm{Re}\left[A_k\right]}\right), \tag{4.24}$$

where we have suppressed the arguments of A_k for the sake of brevity.

In Figure 4.2, we construct a polar plot of the magnitude and argument of the phasors given by Eqs. (4.23) and (4.24). This plot contrasts the orthogonal case given by Eq. (3.23) and plotted in Figure 3.6. The orthogonal phasors have constant magnitude and equiangular spacing, whereas the Gaussian phasors have magnitude and angular spacing dependent on the value of σ. As the value of $\sigma \to 0$, the Gaussian case depicted

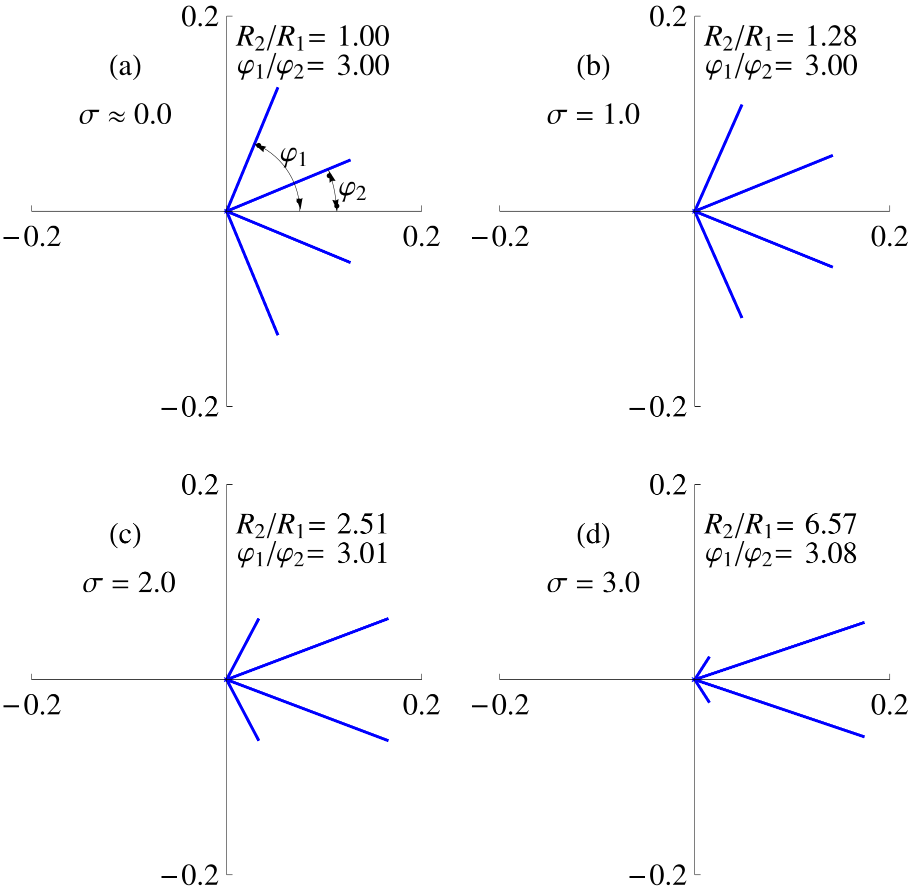

Figure 4.2: Phasor representation of the Gaussian modulated position state given by Eqs. (4.23) and (4.24) for $N = 4$ and $x = \pi/2$. The magnitude and angular spacing exhibit dependency on the value of σ in contrast to the constant and equiangular spaced phasors in the orthogonal case shown in Figure 3.6.

in Figure 4.2(a) appears to coincide with the orthogonal case. We demonstrate that this is indeed the case using Taylor series analysis.

The quantities $R_k(x;\sigma)$ and $\varphi_k(x;\sigma)$ are too opaque to understand limiting behaviour, so we use Taylor series analysis to gain insight. The Taylor series representation of angle $\varphi_k(x;\sigma)$ given by Eq. (4.24) is expressed

$$\varphi_k(x;\sigma) = \frac{(2k-1)x}{N} - \frac{(2k-1)x\sigma^2}{3N^3} + \frac{2(2k-1)x\sigma^4}{45N^5} + O\left(x\sigma^6\right) + O\left(x^3\sigma^2\right) \tag{4.25}$$

where for $\sigma = 0$, we have

$$\varphi_k(x;0) = \frac{(2k-1)x}{N}, \tag{4.26}$$

which presents an equiangular separation between subsequent phasors.

Similarly the Taylor series for magnitude given by Eq. (4.23) is expressed

$$\begin{aligned} R_k(x;\sigma) = &\frac{1}{N\sqrt{\pi}} - \frac{x^2}{6N^3\sqrt{\pi}} + \frac{\left(N^2 - 12(k-1)k - 4\right)\sigma^2}{6N^3\sqrt{\pi}} \\ &+ \frac{\left(-5N^2 + 60(k-1)k + 24\right)\sigma^2 x^2}{180N^5\sqrt{\pi}} + O\left(x^4\sigma^4\right). \end{aligned} \tag{4.27}$$

For $\sigma = 0$, this gives

$$\begin{aligned} R_k(x;0) &= \frac{\sqrt{P}}{N\sqrt{\pi}} - \frac{P^{5/2}x^2}{6N^3\sqrt{\pi}} + \cdots + \frac{(-1)^m\sqrt{P}\left(\frac{Px}{N}\right)^{2m}}{N\sqrt{\pi}(2m+1)!} \\ &= \frac{\sin\left(Px/N\right)}{\sqrt{P\pi}x}, \end{aligned} \tag{4.28}$$

where the last step assumes the limit $m \to \infty$.

Combining the results of Eqs. (4.22), (4.26), and (4.28) gives

$$\phi_z^{(N)}(x;0,P) = \frac{\sin\left(Px/N\right)}{\sqrt{P\pi}x} \sum_{j=1}^{N} (-1)^{z_j} \mathrm{e}^{\mathrm{i}\left(\frac{\mathrm{N}-(2\mathrm{j}-1)}{\mathrm{N}}\right)\mathrm{Px}}, \tag{4.29}$$

which is the encoded orthogonal position wave function given by Eq. (3.23) and in [20]. The proof given for Lemma 3.1 thus suffices to prove Lemma 4.1. □

Lemma 4.2. *For $x = 0$ and $\sigma > 0$ and subject to the balanced constraint $\sum_{j=1}^{N}(-1)^{z_j} = 0$, $\max\left|\Xi_z^{(N)}(0;\sigma)\right|$ occurs for $z \in \mathcal{SB}$.*

Proof. We exploit the structure of the quantity given in Eq. (4.20) with $x = 0$ expressed as

$$A_k(0,\sigma) = \operatorname{erf}\left(\frac{\frac{2k}{N}\sigma^2}{\sqrt{2}\sigma}\right) - \operatorname{erf}\left(\frac{\frac{2(k-1)}{N}\sigma^2}{\sqrt{2}\sigma}\right). \tag{4.30}$$

Using the shorthand $A_k = A_k(0, \sigma)$, we express a set of N terms in the following convenient form

$$\{A_{\frac{N}{2}}, \ldots, A_k, \ldots, A_2, A_1, A_1, A_2, \ldots, A_k, \ldots, A_{\frac{N}{2}}\}, \tag{4.31}$$

for $k = 1, 2, \ldots, \frac{N}{2}$. Note that the A_k are real numbers.

We now show that $A_k > A_{k+1}$. We express the difference between these terms as

$$\begin{aligned} A_k - A_{k+1} = 2\,\mathrm{erf}\left(\frac{\sqrt{2}k\sigma}{N}\right) - \mathrm{erf}\left(\frac{\sqrt{2}(k-1)\sigma}{N}\right) \\ - \mathrm{erf}\left(\frac{\sqrt{2}(k+1)\sigma}{N}\right). \end{aligned} \tag{4.32}$$

Showing that Eq. (4.32) is positive for all k is equivalent to showing that

$$2\,\mathrm{erf}(a) - \mathrm{erf}(a+b) - \mathrm{erf}(a-b) > 0 \tag{4.33}$$

for $a, b \in \mathbb{R}$ and $a, b > 0$.

Over the domain $(0, \infty)$, the error function is strictly monotonically increasing with strictly monotonically decreasing slope $\frac{\mathrm{d}}{\mathrm{d}x}\mathrm{erf}(x) = 2e^{-x^2}/\sqrt{\pi}$. This means that successive increments δx result in decreasing $\delta y = \mathrm{erf}(\delta x)$ increments. This may be expressed as

$$\mathrm{erf}(a) - \mathrm{erf}(a-b) > \mathrm{erf}(a+b) - \mathrm{erf}(a), \tag{4.34}$$

and thus

$$2\mathrm{erf}(a) - \mathrm{erf}(a-b) - \mathrm{erf}(a+b) > 0, \tag{4.35}$$

which establishes that $A_k > A_{k+1}$.

The strategy required to maximize the sum of the N terms of the set (4.31) subject to the balanced constraint is now clear. Since $A_1 > A_2 > \cdots > A_{\frac{N}{2}}$, the maximal term must contain as many of the larger terms as possible. This maximal sum is thus expressed

$$\pm 2\left(A_1 + A_2 + \cdots + A_{\frac{N}{4}} - A_{\frac{N}{4}+1} - A_{\frac{N}{4}+2} - \cdots - A_{\frac{N}{2}}\right). \tag{4.36}$$

This expression manifests the symmetric balanced function definition given in Eq. (4.14) thus proving the Lemma. □

Lemma 4.3. *For $|x| > 0$ and $\sigma \geq 0$ and subject to the balanced condition $\sum_{j=1}^{N}(-1)^{z_j} = 0$, $\max \left|\phi_z^{(N)}(x;\sigma)\right|$ occurs for either $z \in \mathcal{SB}$ or for $z \in \mathcal{AB}$.*

Proof. We modify the set of elements A_k to include the imaginary components resulting from $x > 0$ as

$$\left\{A^*_{\frac{N}{2}}(x,\sigma), \ldots, A^*_1(x,\sigma), A_1(x,\sigma), \ldots, A_{\frac{N}{2}}(x,\sigma)\right\}. \tag{4.37}$$

We now exploit the antisymmetric property of this set. The fact that $\mathrm{erf}(w^*) = \mathrm{erf}^*(w)$ allows us to use the notation

$$A_k(x,\sigma) = \alpha_k + \mathrm{i}\beta_{\mathrm{k}}, \tag{4.38}$$

and

$$A^*_k(x,\sigma) = \alpha_k - \mathrm{i}\beta_{\mathrm{k}} \tag{4.39}$$

to capture the overall of effect of the error function having complex arguments.

The strategy to maximize the sum of the elements in the set given expression (4.37) subject to the balanced constraint is clear. The sum must be either purely real or purely imaginary. A complex sum reduces these achievable maximums in two ways. It causes elements to be subtracted, and it results in a vector sum rather than a liner sum.

The maximum sum subject to the balanced constraint is

$$\begin{aligned} &\sum_{k=1}^{k=N/4} A_k(x,\sigma) + A^*_k(x,\sigma) - \sum_{k=n/4+1}^{k=N/2} A_k(x,\sigma) + A^*_k(x,\sigma) \\ &= 2\left(\sum_{k=1}^{k=N/4} \alpha_k(x,\sigma) - \sum_{k=N/4+1}^{k=N/2} \alpha_k(x,\sigma)\right), \end{aligned} \tag{4.40}$$

which is achieved for the symmetric balanced functions demonstrated in Lemma 2. The maximum imaginary sum subject to the balanced constraint is

$$\begin{aligned}&\sum_{k=1}^{k=N/2} A_k(x,\sigma) - \sum_{k=1}^{k=N/2} A_k^*(x,\sigma)\\ &= 2\mathrm{i} \sum_{\mathrm{k=1}}^{\mathrm{k=N/2}} \beta_{\mathrm{k}}(\mathrm{x},\sigma),\end{aligned} \tag{4.41}$$

which is achieved for the antisymmetric balanced functions.

As x increases from zero, the imaginary component of the error function increases accordingly. For small x, the real part still dominates and the symmetric balanced function is the balanced function with the greatest magnitude. However, there is a point where the antisymmetric balanced function becomes the dominant balanced function. We determine the value of this crossover point, x_c, in terms of σ and P in the following.

The $N = 4$ case is the simplest case which demonstrates the crossover. For this case the set is

$$\{A_2^*(x,\sigma), A_1^*(x,\sigma), A_1(x,\sigma), A_2(x,\sigma)\}. \tag{4.42}$$

The antisymmetric balanced sum is

$$\begin{aligned}&(-A_2^*(x,\sigma) - A_1^*(x,\sigma) + A_1(x,\sigma) + A_2(x,\sigma))\\ &= \alpha_1 + \alpha_2 + \mathrm{i}(\beta_1 + \beta_2) - \alpha_1 - \alpha_2 + \mathrm{i}(\beta_1 + \beta_2)\\ &= \mathrm{i}2(\beta_1 + \beta_2),\end{aligned} \tag{4.43}$$

and the symmetric balanced sum is

$$\begin{aligned}&(-A_2^*(x,\sigma) + A_1^*(x,\sigma) + A_1(x,\sigma) - A_2(x,\sigma))\\ &= 2(\alpha_1 - \alpha_2).\end{aligned} \tag{4.44}$$

The switch over thus occurs when

$$(\beta_1 + \beta_2) = (\alpha_1 - \alpha_2), \tag{4.45}$$

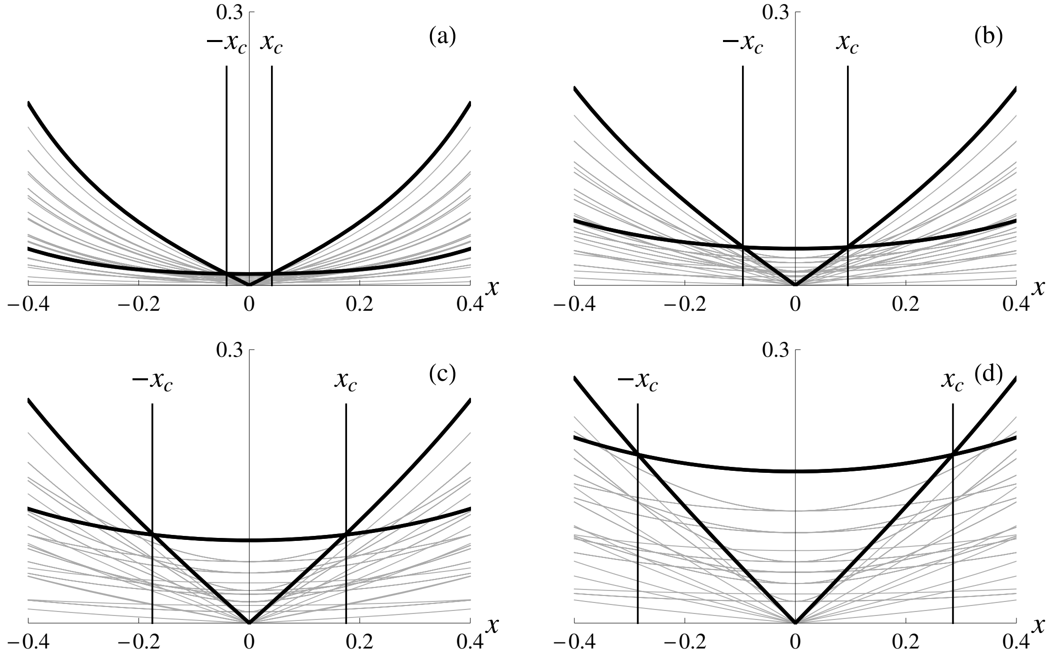

Figure 4.3: Plots of the magnitude of the position modulation function $\left|M_z^{(8)}(x;\sigma,1)\right|$ given by Eq. (4.10) for (a) $\sigma = 0.4$, (b) $\sigma = 0.6$, (c) $\sigma = 0.8$, and (d) $\sigma = 1.0$. For $|x| < x_c$, the symmetric balanced function (bold) dominates all other balanced functions, and for $|x| > x_c$, the antisymmetric balanced function (bold) dominates all other balanced functions.

for which the lowest-order Taylor approximation is

$$x_c \approx \frac{P\sigma^2}{(4 - P^2\sigma^2)}. \tag{4.46}$$

Higher order approximations can be given but this suffices to give an appreciation for the dependencies. □

In Figure 4.3, we demonstrate the switch over from the dominance of the symmetric balanced function to the antisymmetric balanced function at the crossover point x_c given in Eq. (4.46). For $|x| < x_c$, the symmetric balanced function (shown in bold) dominates, and for $|x| > x_c$, the antisymmetric balanced function (also shown in bold) dominates. All remaining balanced functions, of which there are a total of $\binom{8}{4} = 70$ are depicted as light gray lines in Figure 4.3.

4.3.2 Numerical value of single-query success probability

Armed with the knowledge of which balanced functions dominate the measurement window parameterized by δ, we proceed to complete the proof of Theorem 1.

Proof. For the measurement step, we follow the same approach taken in [20] and calculate the probability of detecting a particular wave function in the interval ±δ as

$$\Pr\left[\phi_z^{(N)}(x;\sigma,P)\right] = \int_{-\delta}^{\delta} \left|\phi_z^{(N)}(x;\sigma,P)\right|^2 \mathrm{d}x. \tag{4.47}$$

Since the wave function may be encoded with a constant string or a balanced string, we need to determine the optimal value of δ that maximizes our ability to distinguish between these cases.

We need to maximize the separation between detecting a balanced string and a constant string. To this end, we define the following quantities

$$\Delta_{\mathrm{AB}}(\sigma,P,\delta) = \left|\Pr\left[\phi_{z\in\mathcal{C}}^{(N)}\right] - \Pr\left[\phi_{z\in\mathcal{AB}}^{(N)}\right]\right|, \tag{4.48}$$

and

$$\Delta_{\mathrm{SB}}(\sigma,P,\delta) = \left|\Pr\left[\phi_{z\in\mathcal{C}}^{(N)}\right] - \Pr\left[\phi_{z\in\mathcal{SB}}^{(N)}\right]\right|, \tag{4.49}$$

where for brevity, we have suppressed the arguments in Eq. (4.47). The single-query success probability is defined in these terms as

$$\Pr{}_{\checkmark}^{\sharp} = \min\left[\Delta_{\mathrm{AB}}(\sigma,P,\delta), \Delta_{\mathrm{SB}}(\sigma,P,\delta)\right]. \tag{4.50}$$

This expression assumes that either the antisymmetric or the symmetric balanced strings dominate all other balanced strings in the region ±δ as proved in the previous subsection. We seek to determine the values of δ and σ that maximize the separation between these two probabilities.

With $\Delta_{\mathrm{AB}}(\delta,\sigma,P)$ defined in Eq. (4.48), we set $\Delta'_{\mathrm{AB}}(\delta,\sigma,P) = \frac{\partial}{\partial\delta}\Delta_{\mathrm{AB}}(\delta,\sigma,P)$, which gives us

$$\begin{aligned}
&\Delta'_{\mathrm{AB}}(\delta,\sigma,P) = \\
&\frac{e^{-\frac{\delta^2}{\sigma^2}}}{2\sqrt{\pi}\sigma\,\mathrm{erf}(P\sigma)}\left[\mathrm{erf}\left(\frac{P\sigma^2-\mathrm{i}\delta}{\sqrt{2}\sigma}\right)^2+\mathrm{erf}\left(\frac{P\sigma^2+\mathrm{i}\delta}{\sqrt{2}\sigma}\right)^2\right.\\
&+2\,\mathrm{erf}\left(\frac{P\sigma^2-\mathrm{i}\delta}{\sqrt{2}\sigma}\right)\mathrm{erf}\left(\frac{\mathrm{i}\,\delta}{\sqrt{2}\sigma}\right)+2\,\mathrm{erf}\left(\frac{\mathrm{i}\,\delta}{\sqrt{2}\sigma}\right)^2\\
&\left.+2\,\mathrm{erf}\left(\frac{P\sigma^2+\mathrm{i}\delta}{\sqrt{2}\sigma}\right)\mathrm{erf}\left(\frac{\mathrm{i}\,\delta}{\sqrt{2}\sigma}\right)\right]. \qquad (4.51)
\end{aligned}$$

It suffices to set $\Delta'_{\mathrm{AB}}(\delta,\sigma,P)=0$ to maximize the separation. Before doing so, we elect to simplify Eq. (4.51) by ‘normalizing’ the standard deviation σ and the measurement ‘length’ δ with respect to the encoding ‘length’ P.

We assume that the uncertainty relation [20] remains true but for a constant different than $\pi/2$. We express this as

$$P\delta = \bar{\delta}. \qquad (4.52)$$

This assumption and analysis of the error function arguments of Eq. (4.51) result in a similar uncertainty relationship between P and σ, which we express as

$$P\sigma = \bar{\sigma}. \qquad (4.53)$$

Making the substitutions given by Eq. (4.52) and Eq. (4.53) into Eq. (4.51) and setting it to 0 results in the following expression

$$\begin{aligned}
&\Delta'_{\mathrm{AB}}\left(\bar{\delta},\bar{\sigma}\right)=0\\
&=\left[\mathrm{erf}\left(\frac{\bar{\sigma}^2-\mathrm{i}\,\bar{\delta}}{\sqrt{2}\bar{\sigma}}\right)^2+\mathrm{erf}\left(\frac{\bar{\sigma}^2+\mathrm{i}\,\bar{\delta}}{\sqrt{2}\bar{\sigma}}\right)^2\right.\\
&+2\,\mathrm{erf}\left(\frac{\bar{\sigma}^2-\mathrm{i}\,\bar{\delta}}{\sqrt{2}\bar{\sigma}}\right)\mathrm{erf}\left(\frac{\mathrm{i}\,\bar{\delta}}{\sqrt{2}\bar{\sigma}}\right)+2\,\mathrm{erf}\left(\frac{\mathrm{i}\,\bar{\delta}}{\sqrt{2}\bar{\sigma}}\right)^2\\
&\left.+2\,\mathrm{erf}\left(\frac{\bar{\sigma}^2+\mathrm{i}\,\bar{\delta}}{\sqrt{2}\bar{\sigma}}\right)\mathrm{erf}\left(\frac{\mathrm{i}\,\bar{\delta}}{\sqrt{2}\bar{\sigma}}\right)\right]. \qquad (4.54)
\end{aligned}$$

Note that the variables $\bar{\sigma}$ and $\bar{\delta}$ are, in some sense the 'normalized' Gaussian standard deviation σ and the measurement width δ, 'scaled' by the momentum 'length' P.

Eq. (4.54) is dependent on the two variables, $\bar{\delta}$ and $\bar{\sigma}$ and is thus insufficient to find the global optimum values of $\bar{\sigma}$ and $\bar{\delta}$. We obtain the needed constraint from the similar equation derived from the symmetric balanced function. Following the same steps we did in Eq. (4.51) and Eq. (4.54), we obtain the following expression

$$\begin{aligned}
&\Delta'_{\text{SB}}\left(\bar{\delta},\bar{\sigma}\right) = 0 \\
&= \left[\operatorname{erf}\left(\frac{\bar{\sigma}^2 - i\bar{\delta}}{2\sqrt{2}\bar{\sigma}}\right) + \operatorname{erf}\left(\frac{\bar{\sigma}^2 + i\bar{\delta}}{2\sqrt{2}\bar{\sigma}}\right)\right] \\
&\times \left[-\operatorname{erf}\left(\frac{{}^2 - i\bar{\delta}}{2\sqrt{2}\bar{\sigma}}\right) + \operatorname{erf}\left(\frac{\bar{\sigma}^2 - i\bar{\delta}}{\sqrt{2}\bar{\sigma}}\right)\right. \\
&\left. - \operatorname{erf}\left(\frac{\bar{\sigma}^2 + i\bar{\delta}}{2\sqrt{2}\bar{\sigma}}\right) + \operatorname{erf}\left(\frac{\bar{\sigma}^2 + i\bar{\delta}}{\sqrt{2}\bar{\sigma}}\right)\right].
\end{aligned} \tag{4.55}$$

We solve Eqs. (4.54) and (4.55) simultaneously to establish the optimum values of the measurement lengths $\bar{\delta}_{\text{AB}}$ and $\bar{\delta}_{\text{SB}}$ in terms of the normalized standard deviation $\bar{\sigma}$.

In Figure 4.4, we plot the distributions $|\phi_z^{(N)}(x;\sigma,P)|^2$ for $z \in \{\mathcal{AB},\mathcal{SB},\mathcal{C}\}$ for several values of σ. We also plot vertical lines corresponding to the values of $\bar{\delta}$ corresponding to $\Delta'_{\text{AB}} = 0$ and $\Delta'_{\text{SB}} = 0$. Note that there are values of the normalized parameters $\bar{\sigma}$ and $\bar{\delta}$ where $\Delta'_{\text{AB}}\left(\bar{\delta},\bar{\sigma}\right) = 0$, and $\Delta'_{\text{SB}}\left(\bar{\delta},\bar{\sigma}\right) = 0$ simultaneously. This situation occurs where $\bar{\delta} \approx 2.30$ and $\bar{\sigma} \approx 2.11$ and is depicted in Figure 4.4(b).

However, these values do not optimize the success probability since

$$\Delta_{\text{AB}}(2.30, 2.11) \approx 0.68, \tag{4.56}$$

and

$$\Delta_{\text{SB}}(2.30, 2.11) \approx 0.54. \tag{4.57}$$

Lack of optimality is manifest in the lower of the two above values, which is less than single-query success probability for the orthogonal case $\Pr_{\checkmark}^{\perp} \approx 0.61$.

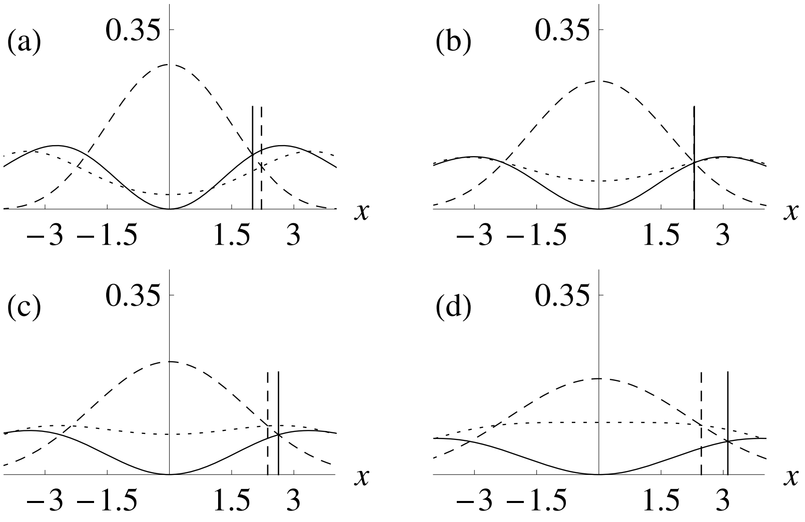

Figure 4.4: Plots of $\left|\phi_z^{(N)}(x;\sigma,1)\right|^2$ for $z \in \{\mathcal{AB},\mathcal{SB},\mathcal{C}\}$ (solid, dotted, dashed respectively) for: (a) $\sigma = 1.67$, (b) $\sigma = 2.11$, (c) $\sigma = 2.5$, and (d) $\sigma = 3.0$. The solid vertical lines in all four sub-plots correspond to $\Delta'_{\text{AB}}\left(\bar{\delta},\bar{\sigma}\right) = 0$, and the dashed vertical lines correspond to $\Delta'_{\text{SB}}\left(\bar{\delta},\bar{\sigma}\right) = 0$. Note that (a) corresponds to the values that optimize the single-query success probability.

Increasing the value of $\bar{\sigma}$ further serves to increase $\Delta_{\text{AB}}\left(\bar{\delta},\bar{\sigma}\right)$ and decrease $\Delta_{\text{SB}}\left(\bar{\delta},\bar{\sigma}\right)$, which worsens the success probability. Reducing the value of $\bar{\sigma}$ brings them together. The quantity $\Delta_{\text{AB}}\left(\bar{\delta},\bar{\sigma}\right)$ thus takes on its maximum value when

$$\Delta'_{\text{AB}}\left(\bar{\delta},\bar{\sigma}\right) = 0 \tag{4.58}$$

subject to the constraint

$$\Delta_{\text{AB}}\left(\bar{\delta},\bar{\sigma}\right) = \Delta_{\text{SB}}\left(\bar{\delta},\bar{\sigma}\right). \tag{4.59}$$

The equality required by Eq. (4.59) occurs at a value of $\bar{\delta}_{\text{AB}} \approx 2.01$ and $\bar{\sigma} \approx 1.67$. Optimality is manifest since

$$\Delta_{\text{AB}}\left(\bar{\delta},\bar{\sigma}\right) = \Delta_{\text{SB}}\left(\bar{\delta},\bar{\sigma}\right) \approx 0.68. \tag{4.60}$$

The optimal situation is depicted in Figure 4.4(a).

For $P = 1$, we express the optimal parameters as

$$\delta^\sharp \approx 2.01, \tag{4.61}$$

and

$$\sigma^\sharp \approx 1.67. \tag{4.62}$$

For these values, the single query success probability of the Gaussian model with sharp information cut-off model is

$$\mathrm{Pr}^\sharp_\checkmark \approx 0.68. \tag{4.63}$$

This is approximately 10% greater than the single query success probability for the orthogonal case, which is $\mathrm{Pr}^\perp_\checkmark \approx 0.61$ given by Eq. (3.43). Thus we have

$$\mathrm{Pr}^\sharp_\checkmark > \mathrm{Pr}^\perp_\checkmark \tag{4.64}$$

as required. □

At first glance, the increase in single-query success probability of the Gaussian case over the orthogonal case appears somewhat surprising. The Gaussian wave functions are coherent states and therefore non-orthogonal [49]. Intuitively, the orthogonal states should be optimal especially given that the finite extent of the momentum wave functions provides a natural fit for encoding finite infirmation.

Upon closer inspection however, we see that the improvement results from the ability to 'tune' the Gaussian spread, represented by σ, to match the encoding length P. No such 'tuning' is possible with the finite states. We depict this in Figure 4.5(a) for the constant case with $P = 1$ and optimal $\sigma^\sharp = 1.67$. We see that the encoded momentum Gaussian wave function is on average narrower than the orthogonal pulse wave function. Since the momentum and position wave functions are Fourier transform pairs, narrowing of one results in broadening of the other.

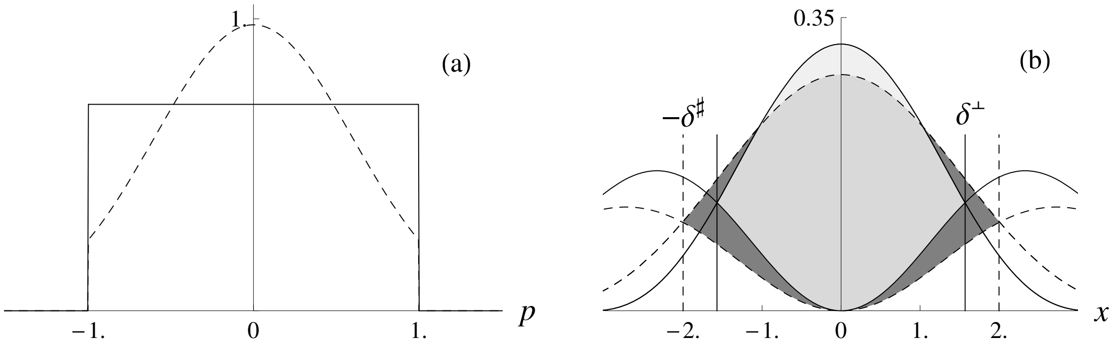

Figure 4.5: Comparison between the orthogonal (solid) and the Gaussian (dashed) wave functions. The amplitude of the encoded momentum functions for the constant case is depicted in (a). The respective algorithm success probabilities are depicted by the shaded regions of (b), where light gray correspond to the orthogonal case and dark gray corresponds to the Gaussian case. Medium grey corresponds to the region of overlap. Note that the encoded momentum Gaussian is on average narrower than the orthogonal pulse, which results in respectively broader position wave functions.

The subsequent broadening of the encoded Gaussian wave functions results in a wider optimal measurement window $\delta^\sharp > \delta^\perp$. This leads to a greater single-query success probability and is represented by the shaded regions in Figure 4.5(b). The larger dark gray region corresponds to the single-query success probability offered by the Gaussian wave functions. We thus conclude that the increased success probability is achieved through the extra degree of freedom afforded by σ. For $P = 1$, this requires that the input state be squeezed to $\sigma \approx 1.67$.

4.4 Summary

We have proven that a simple-harmonic-oscillator quantum computer solving oracle decision problems performs better using non-orthogonal Gaussian wave functions as the algorithm input rather than the orthogonal top-hat wave functions. We have also shown that the limiting case of the Gaussian model for $\sigma \to 0$ and non-zero P corresponds to the model employing orthogonal states. In both cases, the computational bases are orthogonal, and encoding takes place in the momentum domain and information processing and measurement take place in the dual position domain. Also in both cases, the single-query success probability is dependent on the maximum separation between the position wave function encoded with the constant string and the position wave function encoded with the worst-case balanced string, which is the antisymmetric balanced string.

In the orthogonal case, N-bit strings are uniquely encoded into the computational basis formed by the top-hat functions, and the overall-width of the encoded string is set by the encoding length P. In the dual position domain, the encoded string is represented by a sum of N equi-angularly spaced, equi-length phasors multiplied by a sinc function. The rate at which the constant sinc function falls off its peak and the rate that the antisymmetric balanced sinc function rises from its minimum sets the size of the optimum position domain measurement window. Thus the optimum position domain measurement is set by sharpness of the sinc function, which is dependent on the encoding length only.

In the Gaussian case, N-bit strings are uniquely encoded into the computational basis formed by more complicated Gaussian-modulated basis states. The overall-width of the encoded string is again set by the encoding length P, but it is also shaped by the Gaussian spread σ. In the dual position domain, the encoded string is represented by a sum of non equi-angularly spaced and non equi-length phasors multiplied by a Gaussian function. The rate at which the constant encoded function falls off its peak and the rate that the antisymmetric balanced function rises from its minimum is governed by both the Gaussian spread σ and the encoding length P. More importantly, the rate set by the optimal values of σ and P is more gradual than that achievable in the orthogonal case allowing for greater separation between the two probabilities.

We thus conclude that the Gaussian allows for an improved trade-off between encoding, processing and measuring of the information. Encoding takes place in the momentum

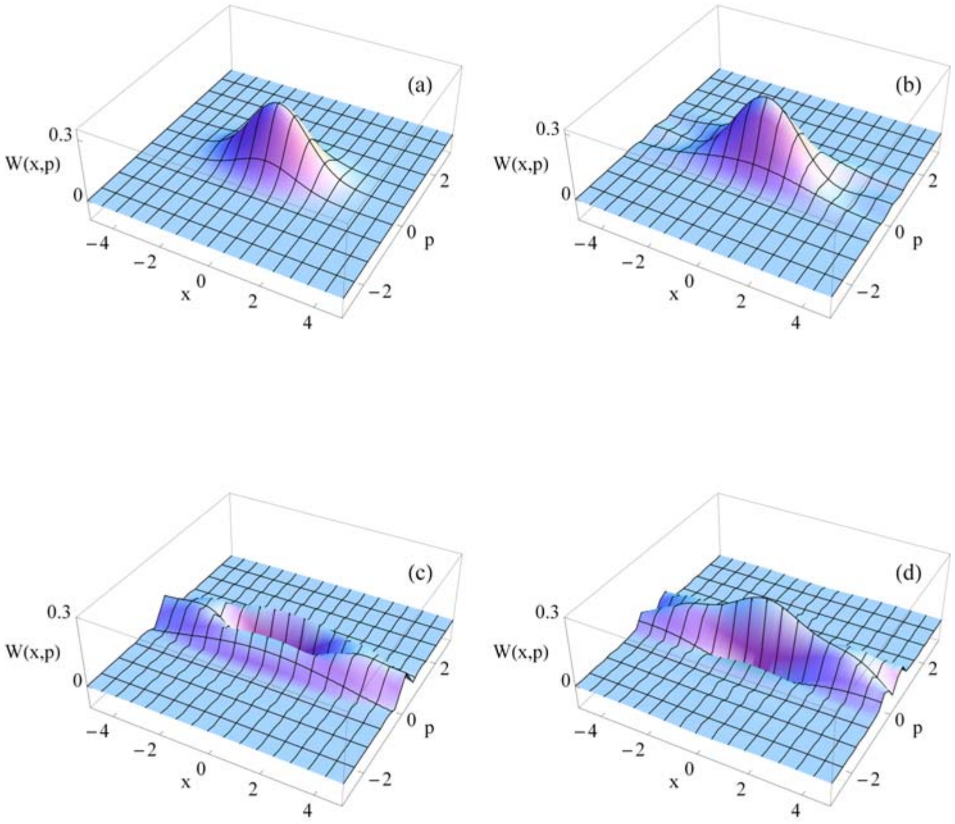

Figure 4.6: Wigner functions of encoded coherent states. The optimally squeezed vacuum with $\sigma = 1.67$ is presented in (a). The optimally encoded constant, antisymmetric and symmetric functions with $P = 1$ are presented in (b), (c) and (d) respectively.

domain, and the Gaussian takes better advantage of the space available to encode the information. Correspondingly, information processing and measurement take place in the dual position domain. The Gaussian-encoded position wave function enables a wider measurement window, which means more of the encoded information is available for distinguishing between a wave function encoded with a constant string and a wave function encoded with the worst-case balanced string.

In Figure 4.6 we present the Wigner functions given by Eq. (2.65), for the four key wave functions presented in our analysis of the continuous-variable algorithm employing coherent states. These help develop intuition for the effect of encoding information in this manner.

The Wigner function of the optimally squeezed vacuum with $\sigma = 1.67$ is presented in Figure 4.6(a). Figure 4.6(b) represents the Wigner function of the optimally encoded constant function with $P = 1$. Note the effect of truncation results in narrowing in the momentum dimension and corresponding broadening in the position dimension compared to the squeezed vacuum. Figure 4.6(c) and Figure 4.6(d) represent the encoded antisymmetric and symmetric functions respectively. We observe a single 'hole' (where the Wigner function takes on negative values) in the antisymmetric Wigner function. We see two holes in the symmetric case. We can conclude from looking at these Wigner functions of encoded balanced functions, that some smoothing of the abrupt encoding techniques employed would result in simpler Wigner functions.

In the next Chapter, we extend the tools we have developed for the analysis of quantum algorithms in continuously-parameterized, infinite-dimensional systems to the analysis of quantum algorithms in continuously-parameterized, yet finite-dimensional systems. We demonstrate that the application of these tools using coherent spin states leads to the discovery of an efficient way for solving a special case of the bounded-distance oracle decision problem [61].

Chapter 5

QUANTUM COMPUTATION WITH COHERENT SPIN STATES

5.1 Introduction

In this chapter, we explore another continuous variable model of quantum computation based on coherent spin states and show that it can inspire a new algorithm. Although fixed total spin states span a finite-dimensional Hilbert space, continuous encoding is possible using the continuously-parameterized, squeezed spin coherent states [51], which are analogous to the squeezed coherent states of the harmonic oscillator [49, 53, 54]. The finite-dimensional Hilbert space has the effect that squeezing of coherent spin states is Heisenberg-limited [51], unlike the coherent states of the harmonic oscillator where the allowed amount of squeezing is unbounded. We consider encoding quantum information into the highest-squeezed spin state that can be achieved.

We demonstrate that this optimally squeezed state may be approximated well by a superposition of two discrete states, thus idealizing the computation model beyond encoding into squeezed spin states while keeping the spin gates. This approach allows us to discover a new algorithm that can be processed using the circuit model of quantum computation. Our investigation of a continuous variable spin model of quantum computation has thus inspired us to find a new quantum algorithm.

This chapter is presented as follows. In Sec. 5.2, we introduce an oracle decision problem parameterized by $N = 2^n$-bit strings. We refer to this oracle decision problem as the *close Hadamard problem*, which is related to the digital coding techniques employed in classical communications [32, 62]. The close Hadamard problem is a special case of what is sometimes referred to as the bounded-distance decoding problem [61, 63]. We differentiate between two different versions of the close Hadamard problem.

The first version is referred to as the *unrestricted* close Hadamard problem. Using the algorithm employing spin states presented in this Chapter, we prove that this problem can be solved for the case where the number of codeword errors l is less than $N/16$ in a linear $O(m)$ number of queries with exponentially small probability $O(e^{-m})$. We note that it can also be solved using the Bernstein–Vazirani algorithm, which solves the bounded-distance decoding problem [61, 63], in a linear number of queries with exponentially small error probability where the number of codeword errors $l <\approx .15N$.

However, we prove that the second version of the problem, which we refer to as the *restricted* close Hadamard problem, can be solved in a single query with certainty independent of the size of the problem using the algorithm employing spin states where the number of codeword errors $l < N/4$. The Bernstein–Vazirani algorithm solves the restricted version of the problem with the same efficiency as it solves the unrestricted version.

The significantly improved efficiency in which the restricted close Hadamard problem can be solved is due to error cancellation that results from employing a symmetric superposition of spin states as algorithm input. The technique of using a superposition of spin states as algorithm input is inspired by the continuous-variable operators and displacement and squeezing tools used in the continuously-parameterized, infinite-dimensional Hilbert space case adapted to the continuously-parameterized, finite-dimensional Hilbert space of a spin system.

In Sec. 5.3, we introduce the spin-system model. We discuss the preparation of coherent spin states and use Q-functions as a visualization aid. We introduce the concept of spin squeezing and show how squeezing changes the amplitude distribution of the individual spin states. We show, that for a particular coherent spin state, the limiting squeezed state is asymptotically approximated by a symmetric superposition of two discrete states with constant error independent of the size of the Hilbert space. We use this superposition as the algorithm input state.

In Sec. 5.4, we prove that our quantum algorithm efficiently solves the *restricted* close Hadamard problem with certainty in a single oracle query. We also prove that our algorithm solves the *unrestricted* close Hadamard problem with arbitrarily small error in a constant number of queries and compare this performance with that achievable using the Bernstein–Vazirani algorithm. In the restricted case in particular, this speed-up is the result of the cancellation of bit errors of certain patterns and results from using a superposition of two states as algorithm input. We also show that any known classical algorithm requires $\Omega(n)$ queries.

In Sec. 5.5, we discuss generalization of the computational model by showing that if the Hadamard operation is replaced by the discrete Fourier transformation, the oracle decision problem changes. We conclude that this model of quantum computation can be used to inspire the efficient solution of additional problems.

5.2 The Close Hadamard Oracle Decision Problem

In this subsection, we specify the particular sets A, B and C required by Definition 1 for the close Hadamard problem. We refer to this decision problem as *close* because we are interested in strings that are *close* in the sense of Hamming distance to the $N = 2^n$-bit strings referred to as Hadamard codewords [64, 32].

The problem of discriminating between codewords received after transmission over a noisy channel is well-known in classical digital coding theory employing linear block codes [31]. Linear block codes are characterized by the triplet $[N, k, t]$, where N is the total length of the codeword, $k < N$ is the amount of information coded, and $t - 1$ is the number of errors that the code can correct.

The Hadamard code is a linear block code with $N = 2^n$, $k = n+1$, and $t = N/4 - 1$. The Hadamard code has a poor information rate k/N, but it has excellent error-correcting capability. Because of this latter feature, the $[32, 6, 7]$ Hadamard code was used to encode picture information on Mariner space craft missions [64].

For $N = 2^n$, the matrix comprising Hadamard codewords is

$$\mathsf{W}^{(N)} = \log_{(-1)}\left[\sqrt{N}\mathsf{H}^{\otimes n}\right], \tag{5.1}$$

where $\mathsf{H}^{\otimes n}$ is the familiar Hadamard matrix of quantum computation [5]. The expression $\log_{(-1)}[x]$ is the *entry-wise* logarithm of x to the base -1, and, if $x = (-1)^y$, then $y = \log_{(-1)}[x]$.

For counting purposes we define the set $\mathbb{Z}_N = \{0, 1, \ldots, N-1\}$. For $j \in \mathbb{Z}_N$, the j^{th} Hadamard codeword corresponds to the j^{th} row of the matrix $\mathsf{W}^{(N)}$ and is expressed as $\mathsf{W}_j^{(N)}$. For example

$$\mathsf{W}^{(4)} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}, \tag{5.2}$$

and $\mathsf{W}_3^{(4)} = 0110$. Note that all Hadamard codewords are balanced with the exception of $\mathsf{W}_0^{(N)}$, which is constant. Also note that all N Hadamard codewords are separated from each other by Hamming distance

$$d\left(\mathsf{W}_j^{(N)}, \mathsf{W}_k^{(N)}\right) = N/2. \tag{5.3}$$

An arbitrary string $z \in \{0,1\}^N$ having Hamming distance $d\left(z, \mathsf{W}_j^{(N)}\right) < N/4$ from any Hadamard codeword is said to be within the $t-$error-correcting capability of the Hadamard code [62].

In our analysis, we introduce Hadamard codewords with two types of bit errors: *unrestricted* errors and *restricted* errors. Unrestricted bit errors can occur at any of the N bit positions, whereas restricted errors are limited to $N/2$ specific bit positions.

5.2.1 Codewords with unrestricted errors

The codewords having *unrestricted* errors are the strings having Hamming distance d from any Hadamard codeword $\mathsf{W}_j^{(N)}$. We define the set of codewords with errors specified with respect to any particular codeword through the use of an error syndrome, which represents all the possible ways an error of d bits can occur.

The error syndrome for d unrestricted errors is

$$U_d = \left\{ z \in \{0,1\}^N \mid |z| = d \right\}. \tag{5.4}$$

The set of codewords having d unrestricted errors with respect to the j^{th} codeword is

$$\Xi_{j,d}^{(N)} = \left\{ z \oplus \mathsf{W}_j^{(N)} \mid z \in U_d \right\}, \tag{5.5}$$

and the set of all correctable codewords with zero to $N/16$ unrestricted errors is

$$\Xi_j^{(N)} = \left\{ \Xi_{j,m}^{(N)} \mid m \in \mathbb{Z}_{N/16} \right\}. \tag{5.6}$$

We proceed in a similar manner with the definition of the Hadamard codewords having restricted errors, which are a subset of $\Xi_j^{(N)}$. Note that the Bernstein–Vazirani algorithm can improve upon this distance, and we discuss this in Sec. 5.4.

5.2.2 Codewords with restricted errors

The codewords having *restricted* errors are the strings with Hamming distance d from Hadamard codeword $\mathsf{W}_j^{(N)}$, but the errors are restricted to the $N/2$ specific bit positions where the codeword $\mathsf{W}_{N-1}^{(N)}$ contains a one. The error syndrome for d *restricted* errors is

$$R_d = \left\{ z \in \{0,1\}^N \mid |z| = d \text{ and } z \preccurlyeq \mathsf{W}_{N-1}^{(N)} \right\}. \tag{5.7}$$

We say that a vector $a \in \{0,1\}^N$ is dominated by a vector $b \in \{0,1\}^N$, denoted $a \preccurlyeq b$, if whenever $a_i = 1$ then also $b_i = 1$. The set of codewords having d restricted errors with respect to the j^{th} codeword is

$$\tilde{\Xi}_{j,d}^{(N)} = \left\{ z \oplus \mathsf{W}_j^{(N)} \mid z \in R_d \right\}. \tag{5.8}$$

We present two examples of the sets given by Eq. (5.8).

For the $N = 8$ case where there is a single restricted error, we have $\mathsf{W}_7^{(8)} = 01101001$, and, for the particular codeword $\mathsf{W}_4^{(8)} = 00001111$,

$$\tilde{\Xi}_{4,1}^{(8)} = \{01001111, 00101111, 00000111, 00001110\}. \tag{5.9}$$

Inspection of the set given in Eq. (5.9) reveals the error alignment with the bit positions where $\mathsf{W}_7^{(8)} = 1$.

For $N = 8$ where there are two restricted errors, the errors may occur at any two of four possible bit positions represented as $\{a, b, c, d\}$, for which there are the $\binom{4}{2} = 6$ distinct bit error pairings $\{ab, ac, ad, bc, bd, cd\}$. The codewords with two errors in this case are

$$\tilde{\Xi}_{4,2}^{(8)} = \{00000110, 00100111, 00101110, 01000111, 01001110, 01101111\}. \tag{5.10}$$

For the general case, the set having m-tuple restricted errors has size $\left|\tilde{\Xi}_{j,m}^{(N)}\right| = \binom{N/2}{m}$.

The set of all correctable codewords with zero to $N/4-1$ restricted errors with respect to the j^{th} Hadamard codeword is

$$\tilde{\Xi}_j^{(N)} = \left\{ \tilde{\Xi}_{j,m}^{(N)} \mid m \in \mathbb{Z}_{N/4} \right\}. \tag{5.11}$$

The size of this set is exponential in N since

$$\left|\tilde{\Xi}_j^{(N)}\right| = \sum_{m=0}^{N/4-1} \binom{\frac{N}{2}}{m} = \frac{1}{2}\left[2^{\frac{N}{2}} - \binom{\frac{N}{2}}{\frac{N}{4}}\right]. \tag{5.12}$$

We now define two variations of the close Hadamard problem in terms of Definition 1.

Problem 2. *Given the set of codewords $\tilde{A} = \tilde{\Xi}^{(N)}_{N/2-1}$, which contains strings that are close (in the restricted sense) to the Hadamard codeword $\mathsf{W}^{(N)}_{N/2-1}$ and the set of codewords $\tilde{B} = \tilde{\Xi}^{(N)}_{k}$, which contains strings that are close (in the restricted sense) to any other Hadamard codeword $\mathsf{W}^{(N)}_{k}$ with $k \in \mathbb{Z}_{N/2-1}$ and a string z randomly selected with uniform distribution μ such that $z \in_{\mu} \tilde{C} = \tilde{A} \cup \tilde{B}$, the* ***restricted close Hadamard problem*** *is to determine if $z \in \tilde{A}$ or $z \in \tilde{B}$ with the fewest oracle queries.*

In our formulation of Problem 2, we have made a technical assumption by setting $j = N/2-1$ in our definition of set $\tilde{A}$. We could have selected any other $j \in \mathbb{Z}_{N/2}$ as long as we excluded the selection from the definition of set $\tilde{B}$. We make this assumption because our quantum algorithm requires the measurement of some qubit. We have arbitrarily, and without loss of generality, set it to the qubit that corresponds to $j = N/2 - 1$. The same assumption is made in our formulation of Problem 3.

Problem 3. *Given the set of codewords $A = \Xi^{(N)}_{N/2-1}$, which contains strings that are close (in the unrestricted sense) to the Hadamard codeword $\mathsf{W}^{(N)}_{N/2-1}$ and the set of codewords $B = \Xi^{(N)}_{k}$, which contains strings that are close (in the unrestricted sense) to any other Hadamard codeword $\mathsf{W}^{(N)}_{k}$ with $k \in \mathbb{Z}_{N/2-1}$ and a string z randomly selected with uniform distribution μ such that $z \in_{\mu} C = A \cup B$. The* ***unrestricted close Hadamard problem*** *is to determine if $z \in A$ or $z \in B$ with the fewest oracle queries.*

5.2.3 Approach to solution using coherent spin states

The quantum circuit presented in Figure 5.1 is a modified version of the single-mode circuit used in Chapters 3 and 4 and presented in Figure 3.2. Our approach is to use a coherent spin system to determine the appropriate input states and identify operators R and $\mathsf{R}^{\dagger}$ that will enable us to re-use this circuit in the provision of an efficient solution to the restricted close Hadamard problem (Problem 2) and the unrestricted close Hadamard problem (Problem 3).

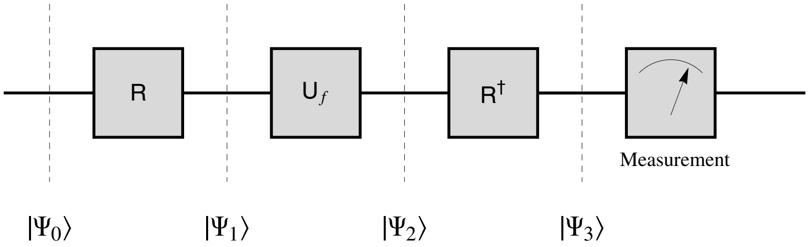

Figure 5.1: Single-mode quantum circuit for oracle decision problems [20].

We thus state two claims in high level terms.

Claim 1. *The restricted close Hadamard problem can be solved with certainty in a single oracle query using the quantum circuit given in Figure 5.1 and the appropriate input state.*

Claim 2. *The un-restricted close Hadamard problem can be solved with arbitrarily small error probability in a constant number of oracle queries using the quantum circuit given in Figure 5.1 and the appropriate input state.*

In the next section, we analyze a coherent spin states in order to define the appropriate input states and to identify the operators that enable us to prove Claim 1 and Claim 2.

5.3 Spin System Model

The quantum circuit represented in Figure 5.1 solves the Deutsch-Jozsa oracle decision problem employing logical states encoded in the infinite-dimensional Hilbert space of the harmonic oscillator [20, 25]. In the harmonic oscillator case, $\Psi(x)$ is a coherent state of the harmonic oscillator, and R and $\mathsf{R}^\dagger$ identified in the figure are the easily implementable continuous Fourier transform and its inverse. Here we adapt this approach to the use of continuously-parameterized coherent spin states.

We will continue to use the oracle represented by the unitary operator [20] given in Eq. (3.3) and repeated here

$$\hat{\mathsf{U}}_f = \begin{pmatrix} (-1)^{z_1} & 0 & \cdots & 0 \\ 0 & (-1)^{z_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & (-1)^{z_N} \end{pmatrix}, \tag{5.13}$$

where the z_i are the bits of the unknown string given in Definition 1 of oracle decision problems. A key aspect of the approach using the coherent states of the harmonic oscillator is the physical accessibility of the harmonic oscillator ground state and the availability of linear and quadratic operators that enable us to prepare the logical input state $|\Psi_0\rangle$. The coherent spin states have similarly accessible states and operators available to us [51].

Mapping this algorithm to the finite-dimensional, continuously-parameterized coherent spin system must first deal with the step that takes the ground state of the spin system to the logical input state $|\Psi_0\rangle$. This step also employs physically accessible, linear spin rotation and quadratic spin squeezing. We make use of coherent spin states [54, 53] in creating an alternative model of continuous-variable quantum computation. Our spin system is a collection of $2S$ elementary $1/2$ spins. Since $2s$ is an odd integer, we choose $2s+1 = N$ so that $N = 2^n$-bit strings may be naturally represented. We refer this as an s-spin system [51].

Just as squeezing is beneficial in continuous-variable quantum computing using coherent states of the harmonic oscillator, we make use of spin squeezing here [51]. However the amount of squeezing in coherent spin systems is bounded, and this will affect our approach. We use the optimally squeezed spin state [51] as input to our algorithm and show that it can be approximated by a superposition of two discrete states with constant error independent of the size of the Hilbert Space.

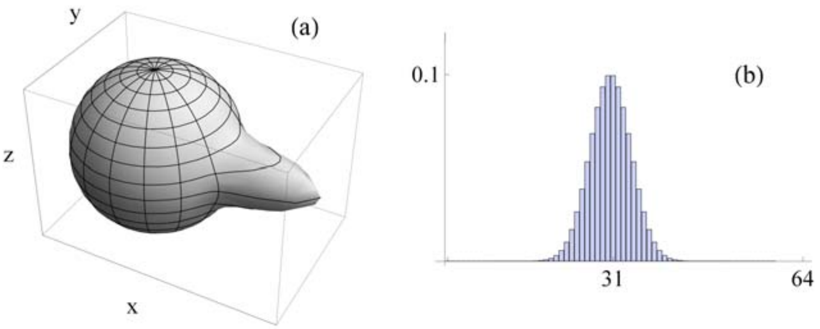

Figure 5.2: (a) Spherical Q-function of the state given by Eq. (5.14) for $s = \frac{63}{2}$, and (b) Plot of the respective Dicke-state probability distribution.

5.3.1 Input state preparation

In our analysis continuous-variable quantum computation using coherent states presented in Chapter 4, the input state was a squeezed ground state. We chose the ground state because the displacement constants only added unnecessary notational complexity. Here we choose to squeeze a maximally rotated coherent state, because the distribution of the spin states are more amenable to computation. In this subsection, we derive an expression for the optimally squeezed coherent spin-state, which we will use as algorithm input.

Using the spin system analogue of the displacement operator Eq. (2.41), we rotate the spin system ground state using the rotation operator $\mathsf{R}_{\theta,\phi}$ given in Eq. (2.59). This results in the coherent spin state

$$\begin{aligned} \left|\pi/2,0\right\rangle_s =& \mathsf{R}_{\pi/2,0}\left|-s\right\rangle_s \\ =& 2^{-s}\sum_{k=0}^{2s}\binom{2s}{k}^{\frac{1}{2}}\left|s-k\right\rangle_s . \end{aligned} \tag{5.14}$$

In Figure 5.2(a), we plot the spherical Q-function for $\left|\pi/2,0\right\rangle_s$. Note that this coherent spin state appears as an ‘equatorial’ state with isotropic uncertainty distribution when represented this way. In Figure 5.2(b), we demonstrate that this state has a Dicke-state amplitude spectrum whose squared magnitude is the binomial probability distribution with $p = q = 1/2$ shown in Figure 5.2(b).

For the coherent states of the harmonic oscillator, the amount of squeezing is unbounded. We selected the amount of squeezing based on an optimal tradeoff between squeezing, the encoding length and the size of measurement window. Since the degree of spin squeezing is bounded by the Heisenberg limit, we take a slightly different approach and develop an expressional for the maximally squeezed spin state. To do this, we employ the two-axis counter-twisting operator [51] given in Eq. (2.62) and repeated here as

$$\mathsf{S}_\mu = e^{\mathrm{i}\frac{\pi}{4}\hat{\mathsf{S}}_\mathrm{x}} e^{\mathrm{i}\mu\left(\hat{\mathsf{S}}_\mathrm{z}^2 - \hat{\mathsf{S}}_\mathrm{y}^2\right)}, \tag{5.15}$$

where μ is the squeezing parameter [51] and the spin operators $\hat{\mathsf{S}}_\mathrm{i}$ are defined in Eq (2.53). The operator $e^{\mathrm{i}\frac{\pi}{4}\hat{\mathsf{S}}_\mathrm{x}}$ orients the resulting anisotropic uncertainty distribution in the y, z directions.

Applying the operator S_μ to

$$|\Psi\rangle = |\pi/2, 0\rangle_s \tag{5.16}$$

allows us to reduce the variance $\Delta\hat{\mathsf{S}}_z^2$ at the expense of enhancing the variance $\Delta\hat{\mathsf{S}}_y^2$. The reduced variance may be expressed as

$$V_- = \langle\hat{\mathsf{S}}_z^2\rangle = \langle\Psi|\,\mathsf{S}_\mu^\dagger\hat{\mathsf{S}}_z^2\mathsf{S}_\mu\,|\Psi\rangle \tag{5.17}$$

since the first moment $\langle\hat{\mathsf{S}}_z\rangle = 0$. In Figure 5.3(a), we plot the quasi-probability distribution of a squeezed spin state. The reduced variance of the squeezed state in the z direction and increased variance in the y direction is evident.

The minimum value of the reduced variance V_- asymptotically approaches 1/2 with increasing s [51]. We refer to the optimal value of the squeezing parameter at this minimum as μ_{opt}. For $\mu > \mu_{\mathrm{opt}}$, the distribution variance increases and the distribution quasi-probability distribution becomes skewed [51]. It can shown be that $\mu_{\mathrm{opt}} \to \frac{1}{s}$ as

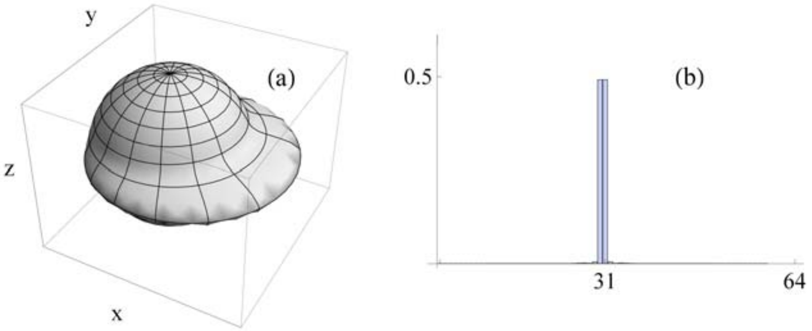

Figure 5.3: (a) Spherical Q-function of the squeezed state given by Eq. (5.18) for $s = \frac{63}{2}$ and $\mu = \mu_{\rm opt}$, and (b) Plot of the respective Dicke-state probability distribution.

$s \to \infty$. This limit is understandable since the variance of a binomial distribution with $p = q = 1/2$ is $N/4$, and squeezing simply has the effect of removing the distribution variance of the dependency on $N = 2s + 1$.

We express the optimally-squeezed spin state as

$$\begin{aligned} \left|\Phi^{(N)}\right\rangle &= \left|\pi/2, 0, \mu_{\rm opt}\right\rangle_s \\ &= \mathsf{S}_{\mu_{\rm opt}} \left|\Psi\right\rangle, \end{aligned} \tag{5.18}$$

with $|\Psi\rangle$ defined in Eq. (5.16). In Figure 5.3(b), we plot the Dicke-state probability distribution of the optimally squeezed state. It is evident that this state approximates the superposition of two spin states. We wish to provide a bound on how well approximated the squeezed state is by a two-component superposition.

Analysis of the variance of the squeezed state's probability distribution is facilitated using the qudit representation rather than the spin sate representation. We thus represent this N-dimensional squeezed state in terms of the qudits $|i\rangle$ as

$$\left|\Phi^{(N)}\right\rangle = \sum_{i=0}^{N-1} \alpha_i |i\rangle. \tag{5.19}$$

The probability distribution associated with $\left|\Phi^{(N)}\right\rangle$ may be represented as the set

$$\mathcal{P}^{(N)} = \{|\alpha_0|^2, \ldots, |\alpha_i|^2, \ldots, |\alpha_{N-1}|^2\}, \tag{5.20}$$

with individual probabilities $\mathcal{P}_i^{(N)} = |\alpha_i|^2$. We note that the squeezed state is symmetric about the centre, and thus the two central states have

$$\mathcal{P}^{(N)}_{(N/2-1)} = \mathcal{P}^{(N)}_{(N/2)} = P_c^{(N)}, \tag{5.21}$$

and thereby form the principle components of the probability distribution of optimally squeezed states.

For $s > \frac{3}{2}$, the expression for the reduced variance given by Eq. (5.17) requires solving eigenvalue problems of degree greater than eight and is no longer analytic, and we must resort to numerical analysis. For $s = 3/2$ and $N = 4$ the expression for the reduced variance given by Eq. (5.17) is analytic, and $\mu_{\text{opt}} = \frac{\pi}{6\sqrt{3}}$. For $N = 4$, we can thus represent the optimal squeezed state as

$$\left|\Phi^{(4)}\right\rangle = e^{\mathrm{i}\phi}\left(0|0\rangle + \frac{1}{\sqrt{2}}|1\rangle + \frac{1}{\sqrt{2}}|2\rangle + 0|3\rangle\right), \tag{5.22}$$

where $e^{\mathrm{i}\phi}$ is a global phase picked up by the action of S_μ. The associated probability distribution is

$$\mathcal{P}^{(4)} = \{0, 1/2, 1/2, 0\}. \tag{5.23}$$

For this case we achieve what we refer to as 'perfect' squeezing, where 'perfect' means that the two central components have probability equal to a half, and the probability of the other two components is zero.

However, this four component distribution has a variance of only a quarter, where the distribution variance is expressed as

$$\mathrm{Var}\left[\mathcal{P}^{(N)}\right] = \sum_{i=0}^{N-1} i^2 |\alpha_i|^2 - \left(\sum_{i=0}^{N-1} i\,|\alpha_i|^2\right)^2. \tag{5.24}$$

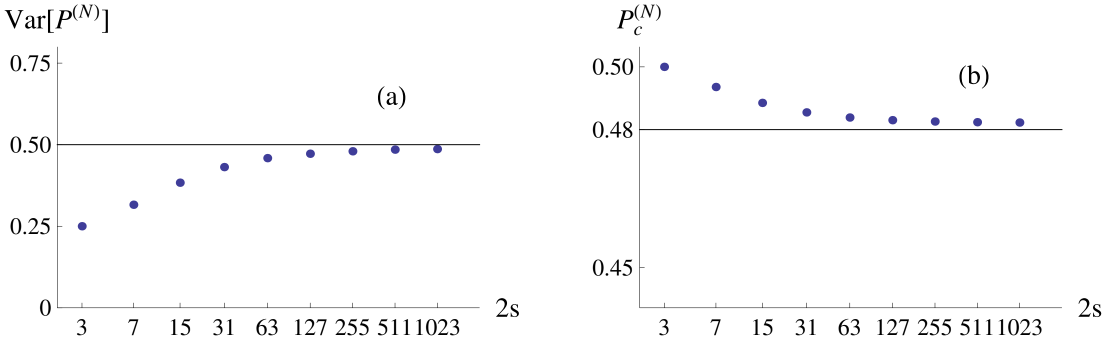

Figure 5.4: (a) Calculated value of the reduced variance of the probability distribution given by Eq. (5.17) approaches 1/2 with increasing s as predicted [51]. (b) Calculated value of the probability of the two central components given by Eq. (5.21) is bounded by the constant given by Eq. (5.28).

Indeed for all N if $\mathcal{P}_c^{(N)} = 1/2$ then,

$$\mathrm{Var}\left[\mathcal{P}^{(N)}\right] = \frac{1}{2}\left((N/2-1)^2 + (N/2)^2\right) - \frac{1}{4}\left(N-1\right)^2 = \frac{1}{4}. \tag{5.25}$$

Since the distribution variance approaches 1/2 as $N = 2s+1$ approaches infinity, perfect squeezing in the sense we have defined is not possible. We use the variance equals 1/2 as a means to bound $\mathcal{P}_c^{(N)}$ defined in Eq. (5.21). In Figure 5.4(a), we plot the calculated values of the reduced variance given by Eqs. (5.17) and (5.24) as a function of s from $s = 3/2$ to $s = 1023/2$, where we observe that the variance approaches $V = 1/2$ as predicted. In order to bound the limiting value of the two central components $\mathcal{P}_c^{(N)}$, we bound the 'tails' of the probability distribution $\mathcal{P}^{(N)}$.

In Figure 5.5, we plot histograms calculated from the squeezed distribution, $\mathcal{P}^{(N)}$, for several values of s, where we have scaled the ordinate to reveal the structure of the tail components. We see that the components immediately adjacent to the central components have

$$\mathcal{P}_{N/2+1}^{(N)} = \mathcal{P}_{N/2-2}^{(N)} \approx 0, \tag{5.26}$$

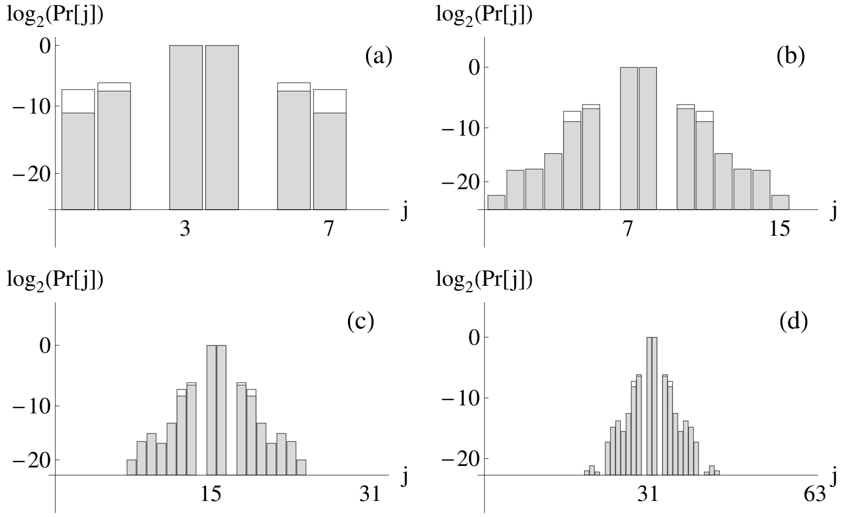

Figure 5.5: Histograms of the probability distribution $\mathcal{P}^{(N)}$ for (a) $s = 7/2$, (b) $s = 15/2$, (c) $s = 31/2$ and (d) $s = 63/2$ with a logarithmic scale for the ordinate. The bounding distribution given by Eqs. (5.27) and (5.28) is overlayed on each of the histograms.

and further outlying terms tail off in an exponential-like fashion. We thus introduce the following bounding probability distribution

$$\mathcal{P}_{\mathcal{B}}^{(N)} = \left\{0, \ldots, 0, \frac{\epsilon}{3}, \frac{2\epsilon}{3}, 0, \frac{1}{2} - \epsilon, \frac{1}{2} - \epsilon, 0, \frac{2\epsilon}{3}, \frac{\epsilon}{3}, 0, \ldots, 0\right\}, \tag{5.27}$$

in order to calculate a bound on the two central components of the distribution.

Solving Var $\left[\mathcal{P}_{\mathcal{B}}^{(N)}\right] = 1/2$ for ϵ gives the probability of the two central components

$$\mathcal{P}_{\mathcal{B}_c}^{(N)} = \frac{1}{2} - \epsilon \approx 0.484. \tag{5.28}$$

In Figure 5.4(b), we plot the calculated values of $\mathcal{P}_c^{(N)}$, where we note that it goes from 1/2 at $s = 3/2$ and asymptotically approaches the constant bounded from below by Eq. (5.28). The bounding distribution is also overlayed on the histograms presented in Figure 5.5.

Since greater than 98% of the probability is manifest in the two central components, we approximate the optimally squeezed input state by the superposition of two states

$$|\Psi_0\rangle = \frac{1}{\sqrt{2}}\left(\left|\frac{1}{2}\right\rangle_s + \left|-\frac{1}{2}\right\rangle_s\right), \tag{5.29}$$

and we can leave the spin system model behind because the ideal state given in Eq. (5.29) can be used as input on any quantum computer. We emphasize that the idea to use it as input to the algorithm give in Figure 5.1 has been inspired through the analysis of the continuously parameterized finite-dimensional Hilbert space of the s-spin system.

5.4 Efficient Algorithm for the Close Hadamard Problem

An important principle in quantum information processing is the solving of problems with increased efficiency compared to classical information processing. Efficiency can be measured in terms of a problem's query complexity, and in this section we present two claims quantifying the query complexity required to solve the close Hadamard problem in the quantum setting.

For a quantum algorithm employing the quantum circuit in Figure 5.1 with $\mathsf{R} = \mathsf{R}^\dagger = \mathsf{H}^{\otimes n}$, we state the number of oracle queries required to solve the restricted and the unrestricted close Hadamard problem defined in Problem 2 and Problem 3 respectively. We prove these claims in three Lemmas. We follow this with a discussion of the query complexity of classical algorithms.

With the idealized input state simplified to

$$|\Psi_0\rangle = \left|\frac{1}{2}\right\rangle_s + \left|-\frac{1}{2}\right\rangle_s, \tag{5.30}$$

where we have suppressed the normalization factor $\frac{1}{\sqrt{2}}$ in Eq. (5.29), the action of the algorithm on the input state is expressed as

$$|\Psi_3\rangle = \mathsf{H}^{\otimes n}\hat{\mathsf{U}}_z\mathsf{H}^{\otimes n}\,|\Psi_0\rangle\,. \tag{5.31}$$

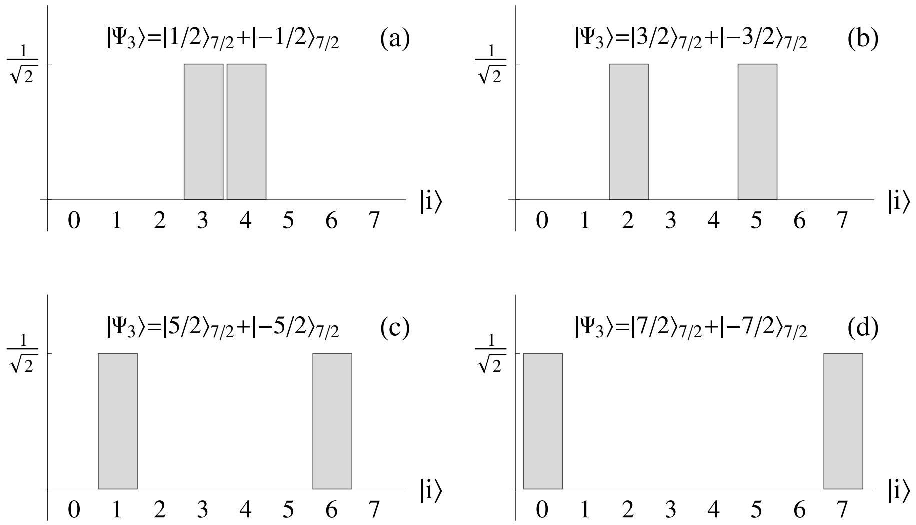

Figure 5.6: For the example of $s = 7/2$, $N = 8$, the output state $|\Psi_3\rangle$ given by Eq. (5.34) remains a symmetric superposition of two states for the Hadamard codewords: (a) $\mathsf{W}_0^{(8)}$, (b)$\mathsf{W}_1^{(8)}$, (c) $\mathsf{W}_2^{(8)}$, and (d) $\mathsf{W}_3^{(8)}$.

The N-bit string z represents the function f. We show that the algorithm efficiently solves both versions of the close Hadamard problem.

5.4.1 Algorithm Response to the Hadamard Codewords

A key feature of the input state is that it is a symmetric superposition of two basis states. When Hadamard codewords are encoded into the oracle, the action of the algorithm preserves the symmetric superposition. This preservation is demonstrated in Figure 5.6. We thus prove the following Lemma is true.

Lemma 5.1. *Given the input $|\Psi_0\rangle = \left|\frac{1}{2}\right\rangle_s + \left|-\frac{1}{2}\right\rangle_s$ to the circuit shown in Figure 5.1 and the oracle encoded with one of the Hadamard codewords $z = \mathsf{W}_j^{(N)}$ for $N = 2s + 1$ and $0 \le j < \frac{N}{2}$, the output state is another superposition of spin states $|\Psi_3\rangle = \left|\frac{1}{2} + j\right\rangle_s + \left|-\frac{1}{2} - j\right\rangle_s$.*

Proof. In order to simplify notation in the following, we suppress the superscript N in

$\mathsf{W}_j^{(N)}$. With W defined as the set of Hadamard codewords given by Eq. (5.1), the pair $(\mathsf{W}, \oplus)$ forms a group under addition modulo two [62]. In particular, the identity element is W_0, and each element is its own inverse since

$$\mathsf{W}_j \oplus \mathsf{W}_j = \mathsf{W}_0. \tag{5.32}$$

It follows from the group property that the addition of any two codewords is another codeword. For the Hadamard codeword pairs W_j and W_{N-1-j}, it can be readily shown that

$$\mathsf{W}_j \oplus \mathsf{W}_{N-1-j} = \mathsf{W}_{N-1} \tag{5.33}$$

for all $j \in \mathbb{Z}_N$.

With some algebraic manipulation, the state $|\Psi_3\rangle$ may be expressed as

$$|\Psi_3\rangle = \frac{1}{N} \sum_{y=0}^{N-1} \left(\sum_{x=0}^{N-1} \alpha_{x,y} \right) |y\rangle \,. \tag{5.34}$$

We use the qudit representation $|y\rangle$ rather than the spin state representation $|m\rangle_s$. We translate back to spin state representation as the last step.

The symbol

$$\alpha_{x,y} = (-1)^{\left(z \oplus \mathsf{W}_{y+N/2}\right)_x} + (-1)^{\left(z \oplus \mathsf{W}_{N/2-(y+1)}\right)_x}, \tag{5.35}$$

where $z = \mathsf{W}_j$ is the string encoded in the oracle, and the symbol x represents the x^{th} bit of the N-bit strings. Note that the sums $y + N/2$ and $N/2 - (y + 1)$ in Eq. (5.35) are modulo N sums.

The Hadamard codewords are balanced with the exception of W_0, which is constant. For $j \neq k$, this allows us to write

$$0 = \sum_{x=0}^{N-1} (-1)^{(\mathsf{W}_j \oplus \mathsf{W}_k)_x}, \tag{5.36}$$

and for $j = k$,

$$N = \sum_{x=0}^{N-1} (-1)^{(\mathsf{W}_j \oplus \mathsf{W}_j)_x}, \tag{5.37}$$

where we have used the group inverse relation given in Eq. (5.32). Using this result, we see that a non-zero sum of $\alpha_{x,y}$ in Eq. (5.34) occurs exactly twice when $y = (N/2 + j) \mod N$ and when $y = (N/2 - 1 - j) \mod N$.

In the qudit representation, the output state is thus expressed

$$|\Psi_3\rangle = |N/2 - 1 - j\rangle + |N/2 + j\rangle\,, \tag{5.38}$$

where qudit indices are understood to be modulo N. For $0 \leq j < N/2$, this translates back to the spin basis as

$$|\Psi_3\rangle = |-1/2 - j\rangle_s + |1/2 + j\rangle_s\,, \tag{5.39}$$

thus completing the proof of Lemma 1. □

We now show that the superposition of two states is also preserved for codewords with restricted errors.

Lemma 5.2. *Given the input* $|\Psi_0\rangle = \left|\frac{1}{2}\right\rangle_s + \left|-\frac{1}{2}\right\rangle_s$ *to the circuit shown in Figure 5.1 and the oracle encoded with* $z \in \tilde{\Xi}_j^{(N)}$ *given by Eq. (5.11), which is a codeword having restricted errors, and with* $0 \leq j < \frac{N}{2}$*, the output state is another superposition of spin states* $|\Psi_3\rangle = \left|\frac{1}{2} + j\right\rangle_s + \left|-\frac{1}{2} - j\right\rangle_s$.

Proof. Observe that the form of Eq. (5.35) allows for cancelling of errors. Under certain conditions if an error occurs at bit position x, the effect on the left-hand side of the plus sign is cancelled by the opposite effect on the right-hand side. Consider the x^{th} bit error in Eq. (5.35), where we have cancellation

$$0 = (-1)^{\left(z \oplus \mathsf{W}_{y+N/2}\right)_x} + (-1)^{\left(z \oplus \mathsf{W}_{N/2-(y+1)}\right)_x}. \tag{5.40}$$

This identity implies that

$$\begin{aligned} 1 &= \big(\mathsf{W}_{y+N/2} \oplus \mathsf{W}_{N/2-(y+1)}\big)_x \\ &= (\mathsf{W}_{N-1})_x, \end{aligned} \tag{5.41}$$

where we have used the result expressed in Eq. (5.33). This is exactly the same as the requirement to be a member of set $\tilde{\Xi}_j^{(N)}$ defined in Eq. (5.11). Under this condition, the result of Lemma 1 holds and $|\Psi_3\rangle = \left|\frac{1}{2}+j\right\rangle_s + \left|-\frac{1}{2}-j\right\rangle_s$.

□

We now show that the perfect superposition of two states is no longer preserved for codewords having unrestricted errors. The effect of errors that are not of the restricted type is to degrade the superposition by distributing amplitude evenly across all other states. However as long as the number of these errors is less than $N/16$, it is still possible to efficiently identify the desired state.

Lemma 5.3. *Given the input $|\Psi_0\rangle = \left|\frac{1}{2}\right\rangle_s + \left|-\frac{1}{2}\right\rangle_s$ to the circuit shown in Figure 5.1 and the oracle encoded with $z \in \Xi_j^{(N)}$ given by Eq. (5.6), which is a codeword having less than $N/16$ unrestricted errors, the desired state can be identified with success probability of at least $\frac{9}{16}$.*

Proof. Let $V_j \in \Xi_{j,1}^{(N)}$ be a Hadamard codeword with a single unrestricted error. We have already shown that if the error is of the restricted type, two-component superpositions are preserved. If the error is not a restricted error, we have no error cancellation, so the single bit-error breaks the balanced and constant sums defined by Eqs. (5.36) and (5.37), respectively. For $j \neq k$ this gives

$$2 = \sum_{x=0}^{N-1} (-1)^{(\mathsf{W}_j \oplus V_k)_x}, \tag{5.42}$$

and for $j = k$,

$$N - 2 = \sum_{x=0}^{N-1} (-1)^{(\mathsf{W}_j \oplus V_j)_x}. \tag{5.43}$$

As the input state is a two-component superposition, the above sums result in all the amplitudes of the output state either acquiring or losing an amount of amplitude proportional to four — two from the amount in the balanced or constant sums given in Eqs. (5.42) and (5.43) and two from the effect of there being two components in the input state.

Thus, we express the output state for the worst case of a single unrestricted error as

$$\begin{aligned} |\Psi_3\rangle = & \frac{1}{\sqrt{2}} \left(1 - \frac{4}{N}\right) \left(\left| \frac{1}{2} + j \right\rangle_s + \left| -\frac{1}{2} - j \right\rangle_s \right) \\ & + \frac{4}{\sqrt{2}N} \sum_{\substack{k=-s-1/2 \\ k \neq \pm j}}^{k=s-1/2} \pm \left| \frac{1}{2} + k \right\rangle_s . \end{aligned} \tag{5.44}$$

For the worst case of l unrestricted errors, where no errors are of the restricted type, the principle components have amplitude

$$\alpha = \frac{1}{\sqrt{2}} \left(1 - \frac{4l}{N}\right), \tag{5.45}$$

and the amplitude of the next largest component is

$$\beta = \frac{4l}{\sqrt{2}N}. \tag{5.46}$$

The amplitude reduction of the principle components by an amount directly proportional to the number of errors results from the constant sum given by Eq. (5.43) being reduced by double the number of errors. However, the balanced sums are variable since errors can cancel. The worst case occurs when the errors are ‘in phase’ resulting in the amplitude of the next-largest component being proportional to the number of errors. The effect of codewords with unrestricted errors on the input superposition is presented in Figure 5.7.

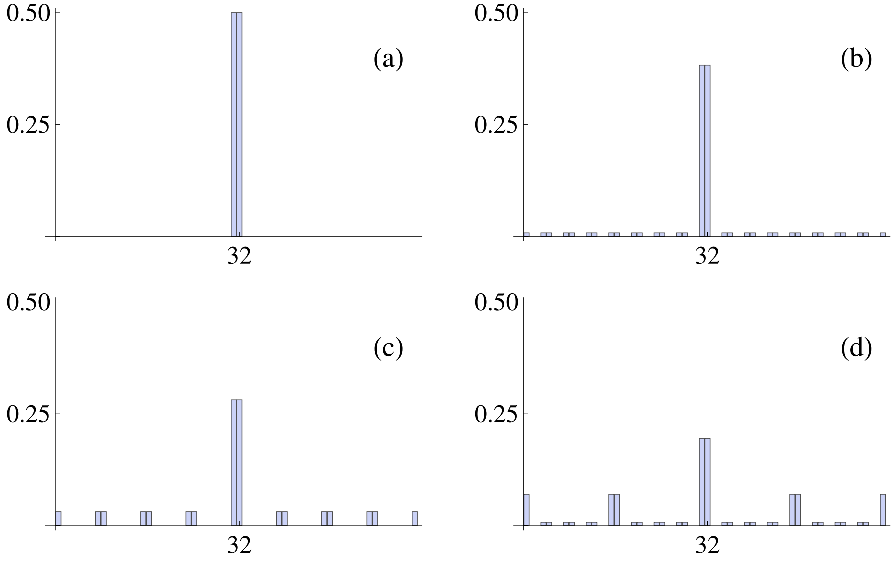

Figure 5.7: For $N = 64$, the effect of l unrestricted errors, where none of the l errors are of the restricted type, on the probability of the central components is demonstrated for (a) $l = 0$, (b) $l = 2$, (c) $l = 4$, and (d) $l = 6$. Figure (c) corresponds to the case that $l = N/16$.

The probability of the two central components is

$$|\alpha|^2 \geq \frac{1}{2}\left(1 - \frac{8l}{N} + \frac{16l^2}{N^2}\right). \tag{5.47}$$

The equality holds for the worst case where none of the errors are of the restricted type. If $l = N/16$, then $|\alpha|^2 \geq \frac{1}{2}\left(\frac{9}{16}\right)$. The amplitudes of the two central states can be combined into a single state with amplitude $\sqrt{2}\alpha$ by an appropriate unitary operation. Since l is less than $N/16$, the desired state can be identified with probability $2\,|\alpha|^2$, which is at least $\frac{9}{16}$. □

The Bernstein–Vazirani algorithm for the solution of the bounded-distance problem given in [61] and presented in the more familiar quantum-circuit framework in [63] gives an improvement over our algorithm for the unrestricted case only. The standard Bernstein–Vazirani algorithm applies the inner-product oracle, which encodes the function

$$f : x \to a \cdot x, \tag{5.48}$$

with $a, x \in \{0,1\}^N$ in this case. Their algorithm determines the N-bit string a in single query.

If the oracle's string has Hamming distance d from the nearest codeword, then the output will have inner product $1 - 2d/N$ with the state that would have been output if the oracle contained the codeword itself. Thus, by performing the standard Bernstein–Vazirani algorithm, the nearest codeword is obtained with probability $(1 - 2d/N)^2$. For $d = N/16$, the success probability is thus $49/64$, which is greater than the success probability of $9/16$ achieved by our algorithm.

Indeed we could improve the number of unrestricted errors handled to $N/8$ since in this case, the Bernstein–Vazirani success probability is $9/16$. Note that the overall query complexity remains unchanged if the single-query success probability is bound from $1/2$ by a constant. In the unrestricted case, the Bernstein–Vazirani algorithm provides a marginal improvement, but we require the error cancelling, which is enabled through use of the input superposition, to achieve the significant improvement our algorithm gives in the restricted case.

In the restricted case, our algorithm allows for codewords having as many as $N/4 - 1$ errors to be determined in a single query. The Bernstein–Vazirani algorithm only identifies the nearest codeword with probability $1/4 + O(1/N)$ at $d = N/4 - 1$. This single-query success probability is inadequate for probability amplification, which needs $d \approx .15N$ since $(1 - 2d/N)^2 = 1/2$ for $d = N/2\left(1 - \sqrt{2}/2\right) \approx .15N$.

We continue with the proof of Claim 1 and Claim 2 with the proviso that the Bernstein–Vazirani algorithm allows for a greater number of errors to be corrected than the $N/16$ we have claimed here in the unrestricted case only. In the more interesting restricted case, we prove the we can allow for $N/4 - 1$ errors and still achieve success in a single query whereas the Bernstein–Vazirani algorithm can only correct for $d < .15N$ errors and requires a linear number of queries to achieve exponentially small error probability.

5.4.2 Solution of the Close Hadamard Problem

We now use results of Lemmas 1, 2 and 3 to show that Claims 1 and 2 are true.

Proof of Claim 1.

By Lemmas 1 and 2, $|\Psi_3\rangle = \left|\frac{1}{2}+j\right\rangle_s + \left|-\frac{1}{2}-j\right\rangle_s$ for $z \in \tilde{\Xi}_j^{(N)}$. Immediately prior to the measurement step, we require a unitary operator $\hat{\mathsf{U}}_{2\mapsto 1}$ that maps this superposition of two states into a single basis state such that

$$\hat{\mathsf{U}}_{2\mapsto 1}\,|\Psi_3\rangle = \left|\frac{1}{2}+j\right\rangle_s . \tag{5.49}$$

Since we know that the unknown string z is either in set $\tilde{\Xi}_{N/2-1}^{(N)}$ or in $\tilde{\Xi}_k^{(N)}$, we wish to measure the outcome of the qudit $|s\rangle_s$ in the spin basis. We define the projection operator [5]

$$\mathsf{M}_s = |s\rangle_s \langle s| , \tag{5.50}$$

and outcome probability is

$$\Pr[s] = \left\langle \frac{1}{2}+j\right|_s \mathsf{M}_s \left|\frac{1}{2}+j\right\rangle_s . \tag{5.51}$$

If $j = N/2-1$, then $\Pr[s] = 1$ and $z \in A$, and if $j \neq N/2-1$, then $\Pr[s] = 0$ and $z \in B$. Thus, the restricted close Hadamard problem is solved with certainty in a single oracle query. □

Proof of Claim 2.

By Lemmas 1 and 2, $|\Psi_3\rangle = \left|\frac{1}{2}+j\right\rangle_s + \left|-\frac{1}{2}-j\right\rangle_s$ for $z \in \tilde{\Xi}_j^{(N)}$. By Lemma 3, the effect of including l unrestricted bit errors on the output state may be expressed as

$$\begin{aligned} |\Psi_3\rangle =& \alpha\left(\left|\frac{1}{2}+j\right\rangle_s + \left|-\frac{1}{2}-j\right\rangle_s\right) \\ & + \beta\sum_{\kappa} \pm \left|\frac{1}{2}+k\right\rangle_s + \gamma\sum_{\lambda} \pm \left|\frac{1}{2}+k\right\rangle_s , \end{aligned} \tag{5.52}$$

where the symbols α and β are given by Eqs. (5.45) and (5.46) respectively, and $|\gamma| < |\beta|$. Note that $\kappa + \lambda = N - 2$, so that all N possible states are accounted for, but the specific value of γ, κ and λ are dependent on N and l. The outcome probabilities of measuring the state $|s\rangle_s$ are thus

$$\Pr[s] = \left\langle \frac{1}{2} + j \right|_s \mathsf{M}_s \left| \frac{1}{2} + j \right\rangle_s . \tag{5.53}$$

If $j = N/2 - 1$, then $\Pr[s] > |\alpha|^2$, and if $j \neq N/2 - 1$, then $\Pr[s] < |\beta|^2$. Assuming that the number of unrestricted errors l is less than $N/16$, then an error of $O(e^{-q})$ can be achieved by making $O(q)$ repetitions of the algorithm [20]. Thus, the unrestricted close Hadamard problem is solved with arbitrarily small error probability in a constant number of queries. □

5.4.3 Classical Algorithm

In this subsection we compare the performance of any classical algorithm to the performance of the quantum algorithm.

Claim 3. *Any classical deterministic algorithm requires $\Omega(n)$ oracle queries of the bit positions to solve the close Hadamard problem with certainty,* even *if there are no bit errors. A randomized algorithm with bounded error probability also requires $\Omega(n)$ queries,* even *if there are no bit errors.*

Claim 3 follows from information theoretical considerations. The goal of the classical strategy is to determine which of the $N/2$ Hadamard codewords is loaded into the oracle. The number of possible solutions is then initially $\Omega(2^n)$. Whenever a classical strategy performs a query, it can eliminate at most half of the remaining possible solutions, even if there are no errors. To reduce the number of possible solutions to a single solution, the classical strategy therefore requires at least $\Omega(n)$ queries[1]. The lower bound also holds

[1] See for example paragraph 6.1 in [65] for an introduction to information theoretic lower bounds.

when the strings loaded into the oracle are Hadamard codewords with errors.

In the next session we discuss how changing the unitary operators R and $\mathsf{R}^\dagger$ in the algorithm shown in Figure 5.1 changes the oracle decision problem that can be solved.

5.5 Alternative Algorithm

The continuously-parameterized, finite-dimensional Hilbert space of the spin system inspired an efficient algorithm for the solution of the close Hadamard problem. The group structure of the Hadamard codewords is implicit in the use of Hadamard operators in the quantum algorithm. We now show that this computation model can inspire other algorithms. Other unitary operators can be employed in the quantum circuit shown in Figure 5.1. The discrete Fourier transform [5] is an obvious alternative. We provide a sketch of how the Fourier transform changes the group structure of the codewords and point to the need for further exploration of problems that could benefit from this computational model.

We replace the operators R and $\mathsf{R}^\dagger$ in Figure 5.1 with the discrete Fourier transform F and $\mathsf{F}^\dagger$. The matrix representation of the discrete Fourier transform is expressed as

$$\mathsf{F}^{(N)} = \frac{1}{\sqrt{N}} \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & \omega & \cdots & \omega^{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{(N-1)} & \cdots & \omega^{(N-1)(N-1)} \end{pmatrix}, \tag{5.54}$$

where $\omega = e^{\frac{\mathrm{i}\pi}{N}}$ [5].

In our analysis of the F-based algorithm, we adopt a similar approach to that taken for the H-based algorithm and define the 'Fourier codewords' as

$$\mathsf{T}^{(N)} = \log_{(-1)} \left[\sqrt{N} \mathsf{F}^{(N)} \right]. \tag{5.55}$$

Similar to the Hadamard codewords, the j^{th} Fourier codeword is the j^{th} row of the matrix $\mathsf{T}^{(N)}$. As an example, we express the $N = 8$ matrix as

$$\mathsf{T}^{(8)} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{4} & \frac{1}{2} & \frac{3}{4} & 1 & -\frac{3}{4} & -\frac{1}{2} & -\frac{1}{4} \\ 0 & \frac{1}{2} & 1 & -\frac{1}{2} & 0 & \frac{1}{2} & 1 & -\frac{1}{2} \\ 0 & \frac{3}{4} & -\frac{1}{2} & \frac{1}{4} & 1 & -\frac{1}{4} & \frac{1}{2} & -\frac{3}{4} \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & -\frac{3}{4} & \frac{1}{2} & -\frac{1}{4} & 1 & \frac{1}{4} & -\frac{1}{2} & \frac{3}{4} \\ 0 & -\frac{1}{2} & 1 & \frac{1}{2} & 0 & -\frac{1}{2} & 1 & \frac{1}{2} \\ 0 & -\frac{1}{4} & -\frac{1}{2} & -\frac{3}{4} & 1 & \frac{3}{4} & \frac{1}{2} & \frac{1}{4} \end{pmatrix}, \tag{5.56}$$

with $\mathsf{T}_4^{(8)} = 01010101$. We see that the Fourier codewords are not bit strings but rather can be thought of as fractional bits. These fractional bits can still be encoded into the oracle function $\hat{\mathsf{U}}_z$ given in Eq. (5.13).

We now define what we term the *simple Fourier codeword* oracle decision problem and show that it can be solved in a single query using the modified algorithm. Note that we have structured this problem along the same lines as the close Hadamard problem with no errors.

Problem 4. *Given the string $\check{A} = \mathsf{T}_{N/2-1}^{(N)}$ and a set of strings $\check{B} = \mathsf{T}_k^{(N)}$, with $k \in \{\mathbb{Z}_N \mid k \neq N/2 - 1\}$ and a string z randomly selected with uniform distribution μ such that $z \in_\mu \check{C} = \check{A} \cup \check{B}$, the **Simple Fourier Codeword** problem is to determine if $z \in \check{A}$ or $z \in \check{B}$ with the fewest oracle queries.*

The action of the algorithm on the input state is expressed as

$$|\Psi_3\rangle = \mathsf{F}^{\dagger(N)} \hat{\mathsf{U}}_z \mathsf{F}^{(N)} \left(\left| \frac{1}{2} \right\rangle_s + \left| -\frac{1}{2} \right\rangle_s \right). \tag{5.57}$$

For $z = \mathsf{T}_j^{(N)}$ and $j \in 0, 1, \ldots, N-1$, the output state can be shown to be

$$|\Psi_3\rangle = \left|\frac{1}{2} + j\right\rangle_s + \left|-\frac{1}{2} + j\right\rangle_s, \tag{5.58}$$

where $\frac{1}{2} + j$ and $-\frac{1}{2} + j$ are modulo s sums in the sense that $\left|\frac{1}{2} + \frac{N}{2}\right\rangle = |-s\rangle$.

We apply $U_{2\mapsto 1}$ given in Eq. (5.49) to the state $|\Psi_3\rangle$ given in Eq. (5.58). Measuring the qudit $|s\rangle_s$ in the spin basis with the measurement operator M_s given in Eq. (5.50), distinguishes whether the encoded string is in set $\breve{A}$ or set $\breve{B}$ thereby solving the simple Fourier codeword problem in a single query.

The result given by Eq. (5.58) is achieved by exploiting group properties similar to those expressed in Eqs. (5.32) and (5.33) for the Hadamard codewords. The columns of $\mathsf{F}^{(N)}$ represent the *multiplicative* cyclic group of order N, where the generator is the first non-trivial column of $\mathsf{F}^{(N)}$. The matrix of codewords $\mathsf{T}^{(N)}$ represents the *additive* cyclic group of order N as a result of taking the logarithm of $\mathsf{F}^{(N)}$.

Each element of the group has the inverse relation

$$\mathsf{T}_j + \mathsf{T}_{N-j} = \mathsf{T}_0, \tag{5.59}$$

and each codeword also obeys the sum relation

$$\mathsf{T}_j + \mathsf{T}_{N/2-j} = \mathsf{T}_{N/2}, \tag{5.60}$$

where $N-j$ and $N/2-j$ are understood to be modulo N sums. In Figure 5.8 we clearly see that, unlike, the Hadamard codewords that preserve the superposition of two symmetric states, the Fourier codewords preserve the superposition of two adjacent states.

Comparison of the effect of the different operators is interesting. The Hadamard codewords preserve symmetric superpositions, and there are $N/2$ unique symmetric superpositions. The Fourier codewords preserve adjacent superpositions, and there are N unique adjacent superpositions.

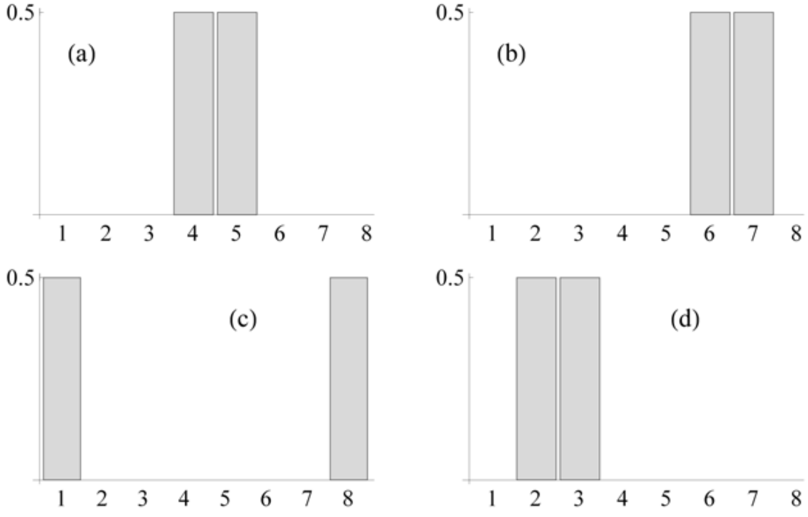

Figure 5.8: For $s = 7/2$, the Fourier-based algorithm probability distributions for the states given by Eq. (5.58) for (a) $j = 0$, (b) $j = 2$ (c) $j = 4$ and (d) $j = 6$. The F-based algorithm preserves the 'adjacency' of the input superposition, whereas the H-based algorithm preserves the 'mirror' symmetry of the input superposition as shown in Figure 5.6.

The exploration of the structure of error cancellations along the lines of Eq. (5.41) using the relationship Eq. (5.60) for the Fourier codewords may lead to new problems that can be efficiently solved using this model of computation. For example, it is natural to apply this error cancellation concept to the simple Fourier codeword oracle decision problem presented in Problem 4 so that the equivalent of Fourier codewords with errors may be included.

5.6 Summary

We have shown that the continuously-parameterized coherent spin state representation of a finite-dimensional Hilbert space of elementary 1/2 spins gives us a new continuous variable model of quantum computation. Like continuous variable quantum computation using the states of the harmonic oscillator, this spin system is amenable to physical preparation with linear rotation and quadratic squeezing operators. Unlike the harmonic oscillator case, spin squeezing is Heisenberg limited. We have shown that the optimally-squeezed coherent spin state is well approximated by a superposition of two discrete states.

Using this idealized state as input into a quantum algorithm presented in Figure 5.1, which is adapted from the single-mode continuous variable case, we have discovered a new oracle decision problem, which we call the close Hadamard problem. This problem is related to the digital error correction technique of employing Hadamard strings as error-tolerant codewords.

We have shown that the unrestricted version of this problem can be solved in a constant number of queries with arbitrarily small error probability, whereas its classical counterpart requires $\Omega(n)$ queries. More interestingly, we have also shown the restricted version of the problem is solved in a single query with certainty.

The observed speedup, in the restricted case, results from the combination of the group structure of the Hadamard codewords, the employment of the Hadamard operator $\mathsf{H}^{\otimes n}$ as the operator R in Figure 5.1 and the use of a symmetric superposition of two basis states as the algorithm input. The tolerance of errors in this case is a direct result of the cancellation of errors that results from this combination.

We have also shown that this algorithm can be generalized. We can change the operator R in Figure 5.1 to the discrete Fourier transform, and loading Fourier codewords into the oracle preserves the symmetry of the input state superposition in a manner analogous to the close Hadamard problem. This pairing between operators and the codewords loaded into the oracle offers the promise of discovery of new problems that can be efficiently solved this way.

We conclude that the continuously-parameterized representation of quantum dynamical systems having a finite-dimensional Hilbert space gives us a model of quantum computation that inspires efficient solutions of new problems.

Chapter 6

SUMMARY AND CONCLUSIONS

This thesis presents a code-state formalism for the solution of oracle decision problems using quantum algorithms in the continuous-variable setting. Prior to this work, the lack of formalism in the infinite-dimensional case resulted in false conclusions regarding the performance of quantum algorithms, and in the finite-dimensional case, continuous-variable algorithms had not been fully explored.

Continuous-variable studies are typically linked to harmonic oscillators because quantum optics has powerful tools to prepare, process and measure optical field modes, which are analogous to harmonic oscillators. Significant progress has been made in recent years in understanding and controlling quantum optics systems. Our code-state formalism furthers this progress by enabling enhanced understanding of the encoding and processing of information in a single mode of the harmonic oscillator. Furthermore we use our formalism to prove that single-mode continuous-variable algorithms in the quantum optics setting are necessarily probabilistic due to an uncertainty relation between the continuous representation and its Fourier-transform dual representation.

Our work addresses the misconception that continuous-variable systems are always infinite dimensional. The continuously parameterized system of coherent spin states is finite dimensional, but the states can be squeezed in a manner similar to states of the harmonic oscillator. This code-state formalism thus enables algorithms to be studied in different continuous-variable systems, which can engender the discovery of new problems and algorithms.

This concluding Chapter is organized as follows. In Sec. 6.1, we list the main conclusions in concise statements of the inferences made as the result of this work. In Sec. 6.2, we summarize the new knowledge contribution made in this thesis. In Sec. 6.3, we present directions for future research. We end with closing remarks in Sec. 6.4.

6.1 Conclusions

A code-state formalism for continuous-variable quantum computation in infinite-dimensional Hilbert spaces is developed in Chapter 3 and in Chapter 4. This code-state formalism is extended to finite-dimensional Hilbert spaces in Chapter 5. We summarize the conclusions for each of the chapters in the following.

6.1.1 Code-state formalism with orthogonal states

In Chapter 3, we establish our code-state formalism using the top-hat/sinc function Fourier transform pair as our orthogonal wave functions. We choose this pair of orthogonal wave functions because of the finite extent of the top-hat wave function. We encode finite-length information strings into the finite region of the momentum domain p of the top-hat function extending from $-P$ to P. Information represented by finite-length bit strings $z \in \{0,1\}^N$ with N the number of bits is encoded into the momentum wave function $\frac{1}{\sqrt{2P}} \sum_{i=0}^{N-1} (-1)^{z_i} |p_{z_i}\rangle$. The kets $|p_{z_i}\rangle$ are phase-modulated by their corresponding bit values, and each of the possible 2^N strings is uniquely represented.

The encoded momentum wave function is Fourier transformed back into the position domain, where the encoded position wave function is represented by a sum of N phasors similarly modulated by $(-1)^{z_i}$ and enveloped by a sinc function. We measure in the position domain, and the optimum measurement window is determined by maximizing the probability between the encoded constant and the dominant balanced function. Since the sinc function has unbounded extent in the position domain and the measurement window is finite, we conclude that the algorithm is necessarily probabilistic and that the single-query success probability $\Pr_{\checkmark}^{\perp} = 0.61$. Thus our formalism clearly establishes that the performance of this algorithm is limited by the very nature of the Fourier transform. This result contradicts material published in the literature [18, 20].

6.1.2 Code-state formalism with coherent states of the harmonic oscillator

The orthogonal states can not be readily created in the laboratory, and in Chapter 4, we extend our formalism to the coherent states of the harmonic oscillator. We are motivated to use the Gaussian wave functions because they represent physically meaningful coherent states, but we have to deal with encoding finite information in their unbounded extent.

We thus modify the approach of embedding information into a momentum top-hat function by instead embedding the information into a Gaussian wave function having width set by its standard deviation σ. The Gaussian is truncated for $|p| > P$, which has the effect that each of the computational basis states, although still orthogonal, are no longer identically sized. We establish that the Gaussian allows for an improved trade-off between encoding, processing and measuring of the information, which is manifest in the single-query success probability for the Gaussian case $\Pr_{\checkmark}^{\sharp} > \Pr_{\checkmark}^{\perp}$ is greater than the single-query success probability for the orthogonal case [25].

Our code-state formalism thus not only allows us to determine fundamental limitations on characteristics of the elements making up the algorithm, it also allows us to learn that alternative encodings can be beneficial. Furthermore, with the exception of the oracle, all elements of the algorithm may be realized using the standard tools of quantum optics. We discuss details of the physical implementation of the algorithm in Sec. 6.2 and describe some of the challenges we face implementing the oracle.

6.1.3 Code-state formalism with coherent spin states

Chapter 5, the formalism is extended to quantum computation in a finite-dimensional Hilbert space, which is "continuous variable" from the perspective of continuously parameterized preparation. Although fixed total spin states span a finite-dimensional Hilbert space, continuous encoding is possible using the continuously parameterized, squeezed spin coherent states [51]. We encode quantum information into the highest-squeezed spin state that can be achieved and demonstrate that this optimally squeezed state may be approximated well by a superposition of two discrete states. We thus idealize the computation model beyond encoding into squeezed spin states while keeping the spin gates, and this approach allows us to discover a new algorithm for the solution of a restricted case of the bounded-distance decoding problem [61, 63].

We refer to this as the restricted close Hadamard problem and demonstrate that the significantly improved efficiency in which it can be solved is due to error cancellation that results from employing a symmetric superposition of spin states as algorithm input. The technique of using a superposition of spin states as algorithm input is inspired by the optimally squeezed spin state, which in turn is inspired from the continuous-variable operators and displacement and squeezing tools used in the infinite-dimensional case adapted to the finite-dimensional Hilbert space of a spin system. Our investigation of a continuous variable spin model of quantum computation has thus inspired us to find a new quantum algorithm.

6.2 Summary of contributions

In this section, we summarize the contributions to knowledge made through the research and the preparation of this thesis. In creating our code-state formalism we needed to: recognize and solve the regularization problem, create a single-mode algorithm, and develop mathematical techniques for bounding the single-query success probability. We also demonstrate that continuously parameterized, finite-dimensional Hilbert spaces may be explored using the techniques used in the infinite-dimensional setting and discover a new algorithm in the process.

6.2.1 Reconciling the regularization problem

Problems become apparent with the continuous-variable quantum algorithms for the solution of the Deutsch–Jozsa in the literature [18], when one considers that the information is missing. It would make intuitive sense that we would be able to ‘see’ the effect of the encoded information the way it can be seen in the discrete case. It is apparent that this obfuscation is the result of encoding information in an unbounded momentum domain and then measuring an infinitesimal position state represented by the Dirac delta functional.

In the discrete quantum domain, the advantage demonstrated by quantum algorithms solving oracle decision problems results from encoding oracle information in a basis that has maximum inner-product overlap with the final measurement basis. This overlap is a constant depending only on the size of the discrete vector space. In the continuous-variable stetting, the quantum advantage is again realized by maximum overlap, but in this setting, the overlap is manifest by a set of unbounded real numbers related to another set of unbounded real numbers by the continuous Fourier transform. This is the source of the regularization problem.

The regularization problem is solved by requiring that the input states be constrained to square-integrable functions. This necessitates a trade-off between the precision of the orthogonal basis used to encode the oracle information and the precision of the measurement window in the Fourier dual domain. This restriction allows us to ‘see’ the individual N-bit strings uniquely represented by bit-modulated sums of continuously parameterized sinc or error functions in the cases studied here. This beautiful representation of discrete information modulating continuous wave functions is ‘swept under the rug’ when ill-defined position states and unbounded momentum states are employed. Discrete information is at the heart of the code-state formalism, and we may embed information in any number of candidate wave functions as long as they meet the square-integrable requirement.

6.2.2 Construction of a single mode algorithm

Simply mapping the standard $(n+1)$-qubit discrete quantum algorithm to the continuous-variable setting, as has been previously done in the literature [18], has two fundamental problems. The first problem is the regularization problem discussed in Sec. 6.2.1. The second is the use of two separate position states requiring two oscillator modes. This choice is the result of a direct mapping of the traditional algorithm to the continuous-variable setting. Two-modes are not strictly required, and in constructing the simplest code-state formalism in the continuous-variable setting, it is desirable to employ a single oscillator mode. This in return necessitates a discrete algorithm without the target state.

In the discrete setting, the single qubit target state provides the 'phase kickback' that encodes the oracle N-bit string into N component phases of the n-qubit control state thus enabling the quantum speedup. In the continuous-variable setting, directly mapping the $(n+1)$-qubit algorithm requires two independent oscillator modes. The continuous-variable control state needs to encode the N-bit oracle string, and the continuous-variable target state needs to provide the phase feedback. This in turns requires that the target state be defined by the infinite-precision real number $|\pi\rangle$ as demonstrated in Eq. (2.88).

We deal with this unrealistic requirement by going back to the discrete case and creating an N-qubit algorithm, which does not require the target state. The traditional oracle may be represented by the $2N \times 2N$ operator given in Eq. (3.2), and our modified but equivalent oracle may be represented by the $N \times N$ operator Eq. (3.3). Mapping this N-qubit algorithm to the continuous-variable setting requires only a single mode of the harmonic oscillator. This innovation allows for our code-state formalism to be expressed in its simplest form. This innovation is also applicable to other quantum algorithms where target states are employed.

6.2.3 Techniques for bounding success probability

For the Deutsch–Jozsa oracle decision problem, we require a technique to distinguish between the constant and balanced cases. In the discrete case, it is sufficient to simply measure the first qubit. Because the discrete algorithm is deterministic, if the measurement returns a 1, the unknown string is constant. If the measurement returns a 0, the string is one of $\binom{N}{N/2}$ balanced strings. It is a little less straightforward in the continuous-variable case because measurement is a probabilistic procedure and because each of the exponential number of balanced strings has a unique probability distribution.

The ability to effectively distinguish between two random events is proportional to the separation of the individual probabilities of occurrence. We thus need to determine which of the balanced stings dominates the measurement window $\pm\delta$. We numerically bound the single-query success probability using the separation between the probability distribution of the constant function and the probability distribution of the worst case balanced function. In the case employing orthogonal wave functions, we prove that the dominant balanced function is the antisymmetric function, which is given in Eq. (3.25). In the case employing the coherent states of the harmonic oscillator, we prove that the dominant balanced function is either the antisymmetric function or the symmetric function given in Eq. (4.14) dependent on the degree of squeezing of the coherent state.

This particular technique of determining the worst case balanced function is necessary for bounding the success probability of the continuous-variable algorithm solving the Deutsch–Jozsa oracle decision problem, but the general technique of determining the bounding functions is applicable to any oracle decision problem.

6.2.4 Continuous-variable algorithms in finite-dimensional Hilbert spaces

In the infinite dimensional continuous-variable case, our formalism demonstrates that squeezing provides an advantage, and the amount of squeezing is only limited by the regularization problem. In the finite dimensional continuous-variable case, we recognized that spin squeezing could also be used to advantage, but in this setting, the amount of squeezing is limited by the Heisenberg uncertainty principle.

We demonstrate that the maximally squeezed spin state may be well approximated by a two-component superposition. Using this superposition as input into the discrete version of the framework transcends the usual approach of using a single computational basis state as algorithm input. This novel use of a superposition as input leads to the discovery and efficient solution of the restricted close Hadamard problem.

6.3 Future research

In this section, we present some suggestions for future research. We first discuss how the algorithm may be implemented in a single mode of the harmonic oscillator. We emphasize that while we know that there must exist a set of unitary transformations that will implement the unitary oracle transformation, the specific set of transformations remains unknown. We also would like to see a tight bound on the single-query success probability in the infinite-dimensional case be generated. We think that further exploration of the operator codeword paring in the finite-dimensional setting would be beneficial.

6.3.1 Physical implementation

Most of the continuous-variable quantum algorithm for the solution of oracle decision problems employing coherent states presented in Chapter 4 can be implemented using the current tools of quantum optics. The exception is the implementation of the oracle

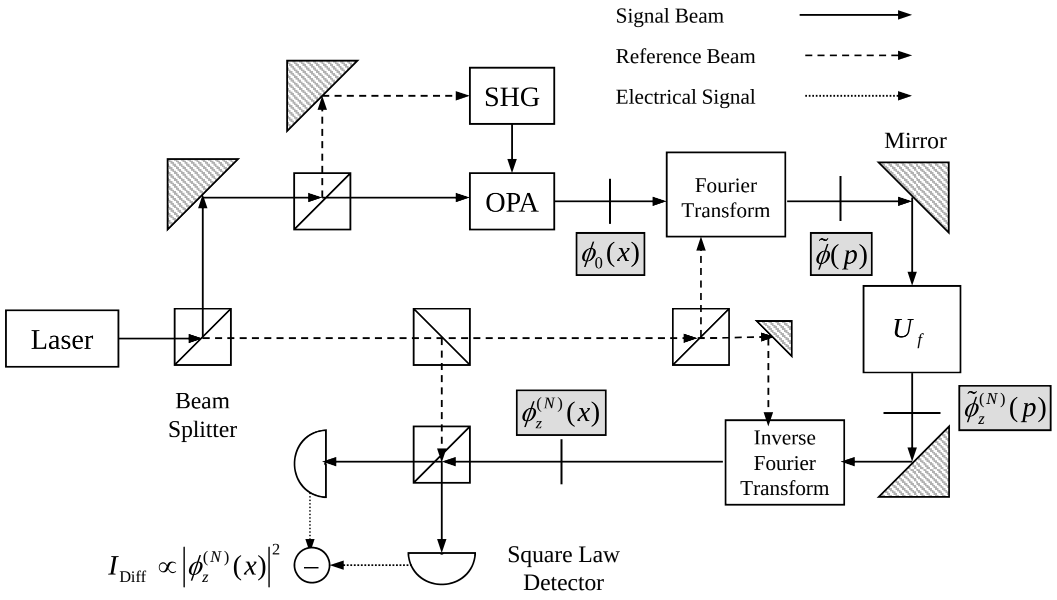

Figure 6.1: Physical implementation of the single-mode, continuous-variable algorithm with the states shown in Fig 3.2 identified with respect to quantum optics elements. A laser generates a coherent state, which is then split into a main beam and a reference beam. The main beam is split again and the second harmonic generator (SHG) is used to pump the optical parametric amplifier (OPA) to create the squeezed input state represented by Eq. (4.3). This state is Fourier transformed with respect to the reference beam. The oracle is applied and the inverse Fourier transform performed. A square-law detector measures the in-phase quadrature.

itself. In this subsection, we describe each of the steps of the algorithm.

Employing a single mode of light, we prepare in the position x domain, encode in the Fourier dual momentum p domain and measure in the position domain. In the language of quantum optics, x is referred to as the in-phase quadrature, and p is referred to as the out-of-phase quadrature. With reference to Figure 6.1, we describe the steps of the algorithm implementation as follows.

1. Prepare a coherent state. High quality lasers generate coherent states [46].

2. Using a beam splitter, split off a reference beam. Beam splitters are linear optical devices [46] and are used in several places in the implementation for the provision

of a reference phase.

3. Squeeze the input coherent state by the optimum amount. (For $P = 1$, $\sigma = 1.67$.) This is achieved using a second harmonic generator to pump an optical parametric amplifier at twice the frequency (the second harmonic) of the main beam [46].

4. Perform a Fourier transform on the squeezed state.

5. Apply the oracle. We will discuss the challenges of oracle implementation.

6. Perform an inverse Fourier transform.

7. Perform measurement using a square law detector implemented by balanced homodyne detection [46], in which the in-phase quadrature component (x) only is measured. Filtering of the frequency of the resulting electrical signal allows for setting of the measurement window.

Regarding the physical implementation of the oracle, we know that there exists a transformation that describes the oracle because the oracle itself is a unitary transformation, which we derive and give in Eq. (3.3). The conditions for universal continuous-variable quantum computation have been proven by [47] and independently by [21] in terms of the standard elements of quantum optics. These conditions imply that a decomposition exists for any unitary operator, and therefore the oracle transformation can be broken up into a series of things that can be implemented (e.g., oscillators, delay channels, squeezers, photon detectors, etc).

However, the decomposition is unknown, and therefore the oracle implementation is unknown. We have treated the oracle as a black box. The unitary operator representation of the oracle is given in Eq. (3.3), and an interesting area of future research is to discover its decomposition in terms of optical elements.

6.3.2 Tightening the bound on the single-query success probability

Another area of interesting future research is to determine a tight bound on the single-query success probability of the continuous-variable Deutsch–Jozsa algorithm. Other encoding techniques are possible. We have rigourously analyzed two encodings in this thesis, but the improvement seen in the Gaussian encoding over the orthogonal encoding indicates that there could be a better regularization that would enable us to saturate the bound.

One possible area of exploration would be to smoothen the abrupt transition that results from truncating the encoding used in the Gaussian case. This could be achieved by introducing a smooth encoding where the antisymmetric and symmetric balanced functions, which are essentially square waves, are replaced with sinusoids having the lowest-frequency terms of their respective Fourier series representations.

6.3.3 Explore operator/codeword symmetries in spin setting

In Sec. 5.5, we discussed an alternative algorithm where we matched the Fourier codewords to the discrete Fourier transform in the algorithm. Further exploration of these operator/oracle-codeword symmetries in the spin setting would make an interesting area of future research. This could be expanded to include employing different oracle functions like the inner-product oracle given in [61] with different operators.

6.4 Closing Remarks

We have learned that we need square-integrable wave functions as the basis of our code-state formalism. The fact that infinitesimal position states can not be used in a code-state formalism, implies that in the case of computing with continuous variables at least, the best we can do is approximate the continuous system in much the same way floating-point numbers approximate real numbers on discrete classical systems. Using one continuously-parameterized system to simulate another in the same way classical analogue computers work does not have the difficulty associated with infinitesimal position states.

Bibliography

[1] Van Meter, R., Itoh, K. M., and Ladd, T. D., "Architecture-dependent execution time of Shor's algorithm," *Proceedings of the International Symposium on Mesoscopic Superconductivity and Spintronics*, 2006, pp. 183–188, available at http://arxiv.org/abs/quant-ph/0507023v2.

[2] Shor, P. W., "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *Society for Industrial and Applied Mathematics Journal on Computing*, Vol. 26, 1997, pp. 1484–1509. DOI:10.1137/S0097539795293172.

[3] Arora, S. and Barak, B., *Computational Complexity: A Modern Approach*, Cambridge University Press, Cambridge, MAJ, 2007.

[4] Sipser, M., *Introduction to the Theory of Computation*, PWS Publishing Company, Boston, MA, 1997.

[5] Nielsen, M. A. and Chuang, I. L., *Quantum Computation and Quantum Information*, Cambridge University Press, Cambridge, UK, 2000.

[6] Kielpinski, D., Monroe, C., and Wineland, D. J., "Architecture for a large-scale ion-trap quantum computer," *Nature*, Vol. 417, 2002, pp. 709–711. DOI:10.1038/nature00784.

[7] Vandersypen, L. M. K., Steffen, M., Breyta, G., Yannoni, C. S., Sherwood, M. H., and Chuang, I. L., "Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance," *Nature*, Vol. 414, 2001, pp. 883–887. DOI:10.1038/414883a.

[8] Loss, D. and DiVincenzo, D. P., “Quantum computation with quantum dots,” *Physical Review A*, Vol. 57, 1998, pp. 120–126. DOI:10.1103/PhysRevA.57.120.

[9] Braunstein, S. L. and Pati, A. K., editors, *Quantum Information with Continuous Variables*, Kluwer Academic Publisher, Dordrecht, NL, 2003.

[10] Furusawa, A., Sørensen, J. L., Braunstein, S. L., Fuchs, C. A., Kimble, H. J., and Polzik, E. S., “Unconditional quantum teleportation,” *Science*, Vol. 282, 1998, pp. 706–709. DOI:10.1126/science.282.5389.706.

[11] Grosshans, F. and Grangier, P., “Continuous variable quantum cryptography using coherent states,” *Physical Review Letters*, Vol. 88, 2002, pp. 057902. DOI:10.1103/PhysRevLett.88.057902.

[12] Appel, J., Figueroa, E., Korystov, D., Lobino, M., and Lvovsky, A. I., “Quantum memory for squeezed light,” *Physical Review Letters*, Vol. 100, 2008, pp. 093602. DOI:10.1103/PhysRevLett.100.093602.

[13] Akamatsu, D., Yokoi, Y., Arikawa, M., Nagatsuka, S., Tanimura, T., Furusawa, A., and Kozuma, M., “Ultraslow propagation of squeezed vacuum pulses with electromagnetically induced transparency,” *Physical Review Letters*, Vol. 99, 2007, pp. 153602. DOI:10.1103/PhysRevLett.99.153602.

[14] Braunstein, S. L., “Error correction for continuous quantum variables,” *Physical Review Letters*, Vol. 80, 1998, pp. 4084–4087. DOI:10.1103/PhysRevLett.80.4084.

[15] Eisert, J., Plenio, M. B., and Scheel, S., “Distilling Gaussian states with Gaussian operations is impossible,” *Physical Review Letters*, Vol. 89, 2002, pp. 137903. DOI:10.1103/PhysRevLett.89.137903.

[16] Bartlett, S. D., Sanders, B. C., Braunstein, S. L., and Nemoto, K., "Efficient classical simulation of continuous variable quantum information processes," *Physical Review Letters*, Vol. 88, 2002, pp. 097904. DOI:10.1103/PhysRevLett.88.097904.

[17] Bartlett, S. D. and Sanders, B. C., "Efficient classical simulation of optical quantum information circuits," *Physical Review Letters*, Vol. 89, 2002, pp. 207903. DOI:10.1103/PhysRevLett.89.207903.

[18] Braunstein, S. L. and Pati, A. K., "Deutsch–Jozsa algorithm for continuous variables," 2002, in [9].

[19] Deutsch, D. and Jozsa, R., "Rapid solution of problems by quantum computation," *Proceedings Royal Society London A*, Vol. 439, December 1992, pp. 553–558. DOI:10.1098/rspa.1992.0167.

[20] Adcock, M. R. A., Høyer, P., and Sanders, B. C., "Limitations on continuous variable quantum algorithms with Fourier transforms," *New Journal of Physics*, Vol. 11, 2009, pp. 103035. DOI:10.1088/1367-2630/11/10/103035.

[21] Gottesman, D., Kitaev, A., and Preskill, J., "Encoding a qubit in an oscillator," *Physical Review A*, Vol. 64, 2001, pp. 012310. DOI:10.1103/PhysRevA.64.012310.

[22] Ghose, S. and Sanders, B. C., "Non-Gaussian ancilla states for continuous variable quantum computation via Gaussian maps," *Journal of Modern Optics.*, Vol. 54, 2007, pp. 855–869. DOI:10.1080/09500340601101575.

[23] Deutsch, D., "Quantum theory, the Church-Turing principle and the universal quantum computer," *Proceedings Royal Society London A*, Vol. 400, July 1985, pp. 97–117. DOI:10.1098/rspa.1985.0070.

[24] Adcock, M. R. A., Høyer, P., and Sanders, B. C., "Gaussian quantum computation with oracle-decision problems," *Quantum Information Processing*, 2012. DOI:10.1007/s11128-012-0489-1.

[25] Adcock, M. R. A., Høyer, P., and Sanders, B. C., "Quantum computation with coherent spin states and the close Hadamard problem," 2011, available at `http://arxiv.org/abs/1112.1446`.

[26] Jordan, S. P., Lee, K. S. M., and Preskill, J., "Quantum algorithms for quantum field theories," *Science*, Vol. 336, 2012, pp. 1130–1133. DOI:10.1126/science.1217069.

[27] Shannon, S., editor, *Trends in Quantum Computing Research*, Nova Science Publishers, New York, NY, 2006.

[28] Piotrowski, E. W. and Sladkowski, J., "Quantum games and programmable quantum systems," 2005, in [27].

[29] Savage, J. E., *Models of Computation: Exploring the Power of Computing*, Addison-Wesley, Boston, MA, 1998.

[30] Turing, A. M., "On computable numbers with an application to the Entscheidungs problem," *Proceedings London Mathematical Society*, Vol. 1, 1936, pp. 230–265. DOI:10.1112/plms/s2-42.1.230.

[31] Michelson, A. M. and Levesque, A. H., *Error Control Techniques for Digital Communication*, John Wiley and Sons, New York, NY, 1985.

[32] MacWilliams, F. J. and Sloane, N. J. A., *The Theory of Error-Correcting Codes*, North Holland, New York, NY, 1977.

[33] Yi, J., Huacan, H., and Yangtian, L., "Ternary optical computer architecture," *Physica Scripta*, Vol. T118, 2005, pp. 98–101. DOI:10.1238/Physica.Topical.118a00098.

[34] Nielsen, M. A., “Cluster-state quantum computation,” *Reports on Mathematical Physics*, Vol. 57, 2006, pp. 147–161. DOI:10.1016/S0034-4877(06)80014-5.

[35] Lloyd, S., “Almost any quantum logic gate is universal,” *Physical Review Letters*, Vol. 75, 1995, pp. 346. DOI:doi:10.1103/PhysRevLett.75.346.

[36] Dawson, C. and Nielsen, M., “The Solovay-Kitaev algorithm,” *Quantum Information and Computation*, Vol. 6, January 2006, pp. 81–95, available at `http://arxiv.org/abs/quant-ph/0505030`.

[37] Cleve, R., Ekert, A., Macchiavello, C., and Mosca, M., “Quantum algorithms revisited,” *Proceedings Royal Society London A*, Vol. 454, September 1998, pp. 339–354. DOI:10.1098/rspa.1998.0164.

[38] Grover, L. K., “A fast quantum mechanical algorithm for database search,” *Proceedings of the twenty-eighth annual Association for Computing Machinery symposium on theory of computing*, 1996, pp. 212–219, available at `http://arXiv:quant-ph/9605043v3`.

[39] Zalka, C., “Grovers quantum searching algorithm is optimal,” *Physical Review A*, Vol. 60, 1999, pp. 27462751. DOI:10.1103/PhysRevA.60.2746.

[40] Neeley, M., Ansmann, M., Bialczak, R. C., Hofheinz, M., Lucero, E., O’Connell, A. D., Sank, D., Wang, H., Wenner, J., Cleland, A. N., Geller, M. R., and Martinis, J. M., “Emulation of a quantum spin with a superconducting phase qudit,” *Science*, Vol. 325(5941), 2009, pp. 722–725. DOI:10.1126/science.1173440.

[41] Blum, L., Shub, M., and Smale, S., “On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines,” *Proceedings of the twenty-ninth annual symposium on the foundations of computer science*, Vol. 21, 1988, pp. 387–397. DOI:10.1109/SFCS.1988.21955.

[42] Jackson, A. S., *Analog Computation*, McGraw-Hill, New York, NY, 1974.

[43] Patterson, D. A. and Hennessy, J. L., *Computer Organisation and Design: The Hardware/Software Interface*, Morgan Kaufmann, Burlington, MA, 1994.

[44] Aaronson, S., "NP-complete problems and physical reality," *Association for Computing Machinery Special Interest Group on Algorithms and Complexity News*, Vol. 36, 2005, pp. 30–52. DOI:10.1145/1052796.1052804.

[45] Taub, H. and Schilling, D. L., *Principles of Communications Systems*, McGraw Hill, New York, NY, 1971.

[46] Leonhardt, U., *Measuring the Quantum State of Light*, Cambridge University Press, Cambridge, UK, 1997.

[47] Lloyd, S. and Braunstein, S., "Quantum computation over continuous variables," *Physical Review Letters*, Vol. 82, 1999, pp. 1784–1787. DOI:10.1103/PhysRevLett.82.1784.

[48] Pati, A. K., Braunstein, S. L., and Lloyd, S., "Quantum searching with continuous variables," 2001, available at `http://arXiv:quant-ph/0002082v2`.

[49] Perelomov, A., *Generalized Coherent States and their Applications*, Springer–Verlag, New York, NY, 1972.

[50] Sobrino, L., *Elements of Non-Relativistic Quantum Mechanics*, World Scientific Publishing, River Edge, NJ, 1996.

[51] Kitagawa, M. and Ueda, M., "Squeezed spin states," *Physical Review A*, Vol. 47, 1993, pp. 5138–5143. DOI:10.1103/PhysRevA.47.5138.

[52] Walls, D. F. and Milburn, G. J., *Quantum Optics*, Springer–Verlag, New York, NY, 1995.

[53] Arrechi, F. T., Courtens, E., Gilmore, R., and Thomas, H., "Atomic coherent states in quantum optics," *Physical Review A*, Vol. 6, 1972, pp. 2211–2237. DOI:10.1103/PhysRevA.6.2211.

[54] Radcliffe, J. M., "Some properties of coherent spin states," *Journal of Physics A: General Physics*, Vol. 4, 1971, pp. 313. DOI:10.1088/0305-4470/4/3/009.

[55] Sanders, B. C., "Quantum dynamics of the nonlinear rotator and the effects of continual spin measurement," *Physical Review A*, Vol. 40, 1989, pp. 2417–2427. DOI:10.1103/PhysRevA.40.2417.

[56] Bracewell, R. N., *The Fourier Transform and Its Applications, 2nd Ed.*, McGraw-Hill, New York, NY, 1986.

[57] Bartlett, S. D., de Guise, H., and Sanders, B. C., "Quantum encodings in spin systems and harmonic oscillators," *Physical Review A*, Vol. 65, 2002, pp. 052316. DOI:10.1103/PhysRevA.65.052316.

[58] Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C., *Introduction to Algorithms*, McGraw-Hill, Cambridge, MA, 2001.

[59] Canny, J. F., "Chernoff bounds," 2001, available at `http://www.cs.berkeley.edu/~jfc/cs174/lecs/lec10/lec10.pdf` Reference downloaded 20 Oct. 2008.

[60] Zwierz, M., Pérez-Delgado, C. A., and Kok, P., "Unifying parameter estimation and the Deutsch–Jozsa algorithm for continuous variables," *Physical Review A*, Vol. 82, 2010, pp. 042320. DOI:10.1103/PhysRevA.82.042320.

[61] Bernstein, E. and Vazirani, U., "Quantum Complexity Theory," *Proceedings of the twenty-fifth annual Association for Computing Machinery symposium on theory of computing*, 1993, pp. 11–20. DOI:10.1145/167088.167097.

[62] Horadam, K. J., *Hadamard Matrices and Their Applications*, Princeton University Press, Princeton, NJ, 2007.

[63] Mosca, M., *Quantum Computer Algorithms*, Ph.D. thesis, Universtiy of Oxford, 1999, available at `http://www.iqc.ca/~mmosca/web/papers/moscathesis.pdf`.

[64] Hall, M., "Semi-automorphisms of Hadamard matrices," *Mathematical Proceedings of the Cambridge Philosophical Society*, Vol. 77, 1975, pp. 459–473. DOI:10.1017/S0305004100051288.

[65] Atallah, M. J., *Algorithms and Theory of Computation Handbook*, CRC Press LLC, Boca Raton, FLA, 1998.