Formal Analysis of Quantum Optics

Mohamed Yousri Mahmoud

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy at

Concordia University

Montréal, Québec, Canada

September 2015

CONCORDIA UNIVERSITY

Division of Graduate Studies

This is to certify that the thesis prepared

By: **Mohamed Yousri Mahmoud**

Entitled: **Formal Analysis of Quantum Optics**

and submitted in partial fulfilment of the requirements for the degree of

Doctor of Philosophy

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____ Dr. Lucia Tirca

_____ Dr. Richard Trefler

_____ Dr. Tarek Zayed

_____ Dr. Weiping Zhu

_____ Dr. Otmane Ait Mohamed

_____ Dr. Sofiène Tahar

Approved by _____

Chair of the ECE Department

_____ 2015 _____

Dean of Engineering

ABSTRACT

Formal Analysis of Quantum Optics

Mohamed Yousri Mahmoud

Concordia University, 2015

At the beginning of the last century, the theory of quantum optics arose and led to a revolution in physics, since it allowed the interpretation of many unknown phenomena and the development of numerous powerful, cutting edge engineering applications, such as high precision laser technology. The analysis and verification of such applications and systems, however, are very complicated. Moreover, traditional analysis tools, e.g., simulation, numerical methods, computer algebra systems, and paper-and-pencil approaches are not well suited for quantum systems. In the last decade, a new emerging verification technique, called formal methods, became common among engineering domains, and has proven to be effective as an analysis tool. Formal methods consist in the development of mathematical models of the system subject for analysis, and deriving computer-aided mathematical proofs. In this thesis, we propose a framework for the analysis of quantum optics based on formal methods, in particular theorem proving. The framework aims at implementing necessary quantum mechanics and optics concepts and theorems that facilitate the modeling of quantum optical devices and circuits, and then reason about them formally. To this end, the framework consists of three major libraries: 1) Mathematical foundations, which mainly contain the theory of complex-valued-function linear spaces, 2) Quantum mechanics, which develops the general rules of quantum physics, and 3) Quantum Optics, which specializes these rules for light beams and implements all related concepts, e.g., light

coherence which is typically emitted by laser sources. On top of these theoretical foundations, we build a library of formal models of a number of optical devices commonly used in quantum circuits, including, *beam splitters*, *light displacers*, and *light phase shifters*. Using the proposed framework, we have been able to formally verify common quantum optical computing circuits, namely the *Flip gate*, *CNOT gate*, and *Mach-Zehnder interferometer*.

To My Wife and Daughter, My Mom and Dad, and My Sister.

ACKNOWLEDGEMENTS

I am deeply grateful to Dr. Sofiène Tahar for his help, guidance and encouragement throughout my graduate studies. I really appreciate his efforts and time spent over weekends and staying at night revising my papers to make perfect submissions. Actually, my experience with him was not only about academic research, it also about life. I cannot forget when he comes back of each scientific trip and tells us about his findings in other parts of the world. I could not have wished for a better thesis supervisor. Many thanks to Dr. Vincent Aravantinous for his support, especially at the early stages of my research. I am also grateful to Dr. Prakash Panangaden who provided me with technical insights throughout the thesis given his very strong knowledge of quantum physics. I also would like to thank Dr. Osman Hassan who firstly introduced me to the Hardware Verification Group and encouraged me to work in the area of formal analysis of quantum optics. I am deeply grateful to my wife for her love and patient throughout my PhD study. Especial thanks to my mother and father, to whom I gift this thesis, for all the support they have provided me over the years. Nothing would be possible without them. Many thanks to the members of the thesis committee for the encouragement and feedback they provided at all levels of the research project. Finally, many thanks to my good friends at the Hardware Verification Group for their support and motivation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

BDD	Binary Decision Diagrams
CAS	Computer Algebra System
CNOT	Controlled NOT Gate
CQP	Communicating Quantum Processes
FOL	First Order Logic
HOL	Higher Order Logic
PVS	Prototype Verification System
Qbit	Quantum bit
QCTL	Quantum Computation Tree Temporal Logics
QMC	Quantum Markov Chain
QPL	Quantum Programming Language
SAT	Boolean Satisfiability

Chapter 1

Introduction

Many phenomena have been studied in classical physics. Not all, however, can be described successfully in the classical paradigm, in particular for condensed matters [68]. Examples of such phenomena are: the physics of atomic shells [11], cohesive energy of solids [65], superconductivity [18] and neutron stars [25]. *Quantum mechanics* [23] then answers many questions regarding those phenomena and more, e.g., nuclear physics and quantum optics.

Quantum mechanics dates back to 1900, when Planck explained the spectral distribution of a thermal cavity on the basis of his postulate that the energy emitted by the cavity is quantized (i.e., discrete). This was considered a partial rejection of classical physics rules which assume that such energy is continuous. Later, in 1905, Einstein was able to show that the photoelectric effect can be explained using Planck's hypothesis. In the same direction of rejecting classical rules, Compton proved, with his X-ray electron collision experiment, the particle nature of light, as opposed to classical theory where light is described as an electromagnetic wave [43].

Quantum optics is an essential branch of quantum mechanics, where the particle-nature of light is considered; typically, these particles are called *photons*. Based on this concept, quantum optics investigates new properties and phenomena about light, especially light beams with a low number of photons [56]. This investigation allows a better usage of existing optical devices, e.g., beam splitters [47], and the invention of totally new quantum devices, e.g., single photon devices [52]. These devices help in different fields: Sometimes they enhance the performance, e.g., the detection of gravitational waves [75], and in other cases they define totally new solutions, in particular quantum computers [53].

In 1980, a new theoretical computing machine was proposed based on quantum mechanics, called a quantum computer [16]. The new machine is expected to show a distinguishable capability in computational theory in comparison with classical machines [50] that suffer from different issues, in particular heating problems. It also provides powerful unbreakable security systems [6]. The implementation of the quantum model has been carried out using different means and technologies, such as: super-conducting circuits [8], ion traps [24], quantum dots [51] and optical circuits [49]. Optical circuits and ion traps are quite promising since they are realized with the highest number of bits in the laboratory [45].

The analysis and verification of such machines, in particular quantum optical circuits, are challenging due to their quantum nature. Traditional analysis techniques are more suitable for systems based on classical theory. Such techniques are, however, applied to quantum circuits, but with certain limitations. In the following, we will discuss some related work on quantum optics analysis, and potential techniques that could solve the problems of the existing work.

1.1 Quantum Optics Analysis: State-of-the-art

System analysis represents a critical issue in every design process. For quantum optics, the analysis techniques currently used are lab-simulation, paper-and-pencil, numerical methods, and computer algebra systems ("CAS"). In the first case, the systems are physically simulated in sophisticated optical laboratories. The simulation of quantum phenomena is a challenging area, where the ultimate goal is to build a universal quantum simulator (or alternatives a *quantum computer*) that can simulate any quantum system. The lab-simulation technology is not yet at this advanced level of building such a universal simulator. However, there are a number of small-scale simulators available, e.g., the usage of *ions traps* to simulate *Dirac equation* [21], and observing *Zitterbewegung* with *Ultracold Atoms* [76]. Note that classical computer simulation is not efficient here since it was proved in 1982 by Feyman that quantum systems cannot be simulated on ordinary computers [16]; the simulation of each time instance requires solving an exponential number of differential equations. Unfortunately, laboratories raise cost and safety issues: optical laboratories cost hundreds of thousands of dollars to build. Moreover, they require a high level of care; otherwise, there would be a high risk of fire [37].

In the paper-and-pencil approach, the whole process (i.e., systems modeling and proving specifications satisfiability) is carried out manually. Typically, the quantum system model is represented as a series of equations, and the analyst tries to derive some intricate quantum properties about the system by subsequent substitutions with the help of his/her knowledge of mathematics and physics. Considering complicated systems in this way results in a large number of mathematical equations which tracking becomes very difficult for a human being, and requires a high degree of expertise

in all aspects. Thereby, the paper-and-pencil analysis turns to be error-prone and very time consuming, particularly for large scale systems. Therefore, computer-aided methods have been developed to help the human – and thus decrease the risk of errors – which fall into the following two categories: CAS (e.g., Maple [36] and Mathematica [14]) and numerical methods (e.g., MATLAB [74]). In the first approach, a quantum mechanics library is developed for educational purposes, where the tools can be used interactively in teaching quantum mechanics courses. The library benefits from the symbolic integration and differentiation capabilities of CAS tools to solve, e.g., the Schrödinger equation, a pillar of quantum theory. The library consists of the analysis of a number of basic quantum systems, e.g., *Free Particle Wavepacket*, *Harmonic Oscillator*. For each system or application, the equations are rewritten in the designated CAS tool and produce the solution symbolically if possible (sometimes numerically). So an application ends up with a series of equations that do not have any abstract object that associates them together. Moreover, each application is developed from scratch and does not benefit from the existing results: For example the *uncertainty principle* is proved for the system of free single particle, but not in a general form [14]. Now, in order to prove the uncertainty principle for another system, it should be tackled from scratch. Later in this thesis, we will see this theorem proved in general for any system.

The second computer-aided approach is based on numerical methods such as the MATLAB toolbox for quantum and atomic optics [74]. This work goes further than the CAS tools since it provides some generality that helps designers (or analysts) to build their own new systems and reason about them. This typically goes beyond the educational purposes which are the main objective of the above mentioned CAS tools. The toolbox is based on representing all quantum objects, e.g., *quantum state*,

Hamiltonian and *density operator* in the form of vectors and matrixes. It also enables to build composite quantum systems out of existing ones with the help of the *tensor product*. The work in [74] showed its effectiveness by tackling a number of applications, e.g., *three-level atom*, *composite system of two-level atoms*, and *laser light force on atoms*. The end goal of any of these applications is to generate a certain differential equation which can be numerically solved either by MATLAB or other external tools, and then generate some graphs about the system behavior. The advantage of [74] is that it tries to implement a low level of abstraction by having meaningful physics objects rather than several disconnected differential equations. On the other hand, it always assumes a finite dimension of quantum states space in order to use the finite MATLAB objects, i.e., vectors and matrixes. Actually this is a problem with many programming languages to have the appropriate data structure that represents the true quantum objects. In this thesis, we generalize our definitions to consider both finite and infinite dimension quantum states space

In the last decade, *formal methods* [31] became a common alternative to traditional computer-aided techniques such as simulation or CAS. This approach involves the development of a formal (i.e., mathematical) proof in which the system model (or implementation) satisfies its specifications. It is in fact a computerized analytical solution that mechanizes the paper-and-pencil approach. There are two main approaches for formal analysis [31]: 1) *model checking* and 2) *theorem proving*. In model checking, the system is modeled as a finite-state automaton, and the specifications are expressed using *temporal logic* (a type of logic which takes time into consideration) [3]. The main advantage of this technique is the automation of the analysis process. However, finite automata cannot express analog and continuous-time physical systems (including quantum ones). In addition, its performance degrades with the size of the

system because of the so-called state explosion problem (i.e., the number of states in the finite automaton becomes tremendous) [3]. Therefore, this technique is more suitable for small systems or for those that can be abstracted. Model checking has been applied in the area of quantum information and quantum cryptography where the quantum systems physical state can be abstracted into two *quantum bits*. This makes the system subject to discrete analysis evolving in finite dimension. In this regard, a number of model checkers were developed for quantum cryptography protocols property checking and quantum circuit equivalence checking, e.g., [78], [15] (to be discussed in next section).

All the above mentioned analysis techniques either suffer from the lack of generality of the developed results, i.e., CAS tools, or the lack of expressiveness of the underlying logic, which does not allow the analysis of real systems but only abstracted ones, i.e., no general notion of quantum mechanics or quantum optics.

Theorem proving is a good candidate to deal with the drawback of the before mentioned techniques. A *theorem prover* is a type of software allowing the specification and model of a given system to be expressed in mathematical logic: either *First-Order-Logic* (FOL) [71] or *Higher-Order Logic* (HOL) [7]. We can then prove properties about the system *inside* the theorem prover (i.e., we prove that the model of the system satisfies its specifications). The main advantage of this technique is its expressiveness (e.g., we can formalize many physical systems regardless of their size and complexity). Hence, theorem proving can help where model checking cannot. However, not all theorem provers are fully automated, in particular HOL provers: they require human interaction. Several theorem provers exist such as HOL4 [70], HOL Light [29], PVS [64], Isabelle [62] or Coq [57].

Accordingly, we believe that formal methods, specifically theorem proving, are able

to deal with the problems of other traditional techniques. Remarkably, HOL provers showed good advancements in engineering and physics domains, e.g., ray optics [69] and electromagnetic optics [41]. Note that HOL theorem provers are more expressive because of high-order-logic, and cost-effective compared to optical laboratories. In this thesis, we propose to use the HOL Light theorem prover which contains a powerful and robust multivariate complex-number library that forms a foremost mathematical foundation of the quantum theory.

The following table compares the currently-used quantum optics analysis methods, in addition to theorem proving as a potential technique, showing the pros and cons of each.

	Expressiveness	Soundness	Cost Effectiveness	Safety
Lab. Simulation	-	+	-	-
Paper-and-pencil	+ +	-	-	+
MATLAB	-	-	+	+
CAS	+	+	+	+
Model Checking	-	+	-	+
Theorem Proving	+ +	+ +	-	+

Table 1.1: Quantum Optics Analysis Methods (- weak, + strong, + + very strong)

1.2 Related Work

To the best of our knowledge, the application of theorem proving in the area of quantum optics has not been tackled before. However, there is exist some work about applying model based verification techniques in the area of quantum information and quantum cryptography. These techniques suit for the analysis and verification of quantum information circuits, where quantum mechanics is abstracted to the *two quantum*

bits. For instance in [78], it is proposed to use a special kind of *Binary Decision Diagrams* (BDD), that are adapted for *quantum gates*, for the equivalence checking of reversible quantum circuits. This work classifies quantum circuits into two types: *properly-quantum* and *not-properly-quantum*. A circuit is properly-quantum if it contains quantum gates that exploit superposition quantum, e.g., *Hadamard gate* [78]. Accordingly, not-properly-quantum circuits are those that do not contain such gates. For the non-properly-quantum circuits, the method generates the corresponding BDD and thus uses conventional equivalence checking techniques. For those circuits that are properly-quantum, the method tries to separate the properly-quantum sub-circuit, if it is found to be a small circuit then it is simulated for equivalence-checking purposes. The remaining non-properly-quantum part is checked using conventional equivalence checking techniques. Note here that the quantum computing circuit can be simulated only if its size is small, since the number of expositional cases to be generated are controllable. The work in [78], also tries to speed up the equivalence-checking by using a *Satisfiability* (SAT) solver at certain instances, whenever combinational logic can represent the non-properly quantum circuits. The proposed methodology has been applied to interesting circuits, e.g., the *quantum carry-ripple* adder, *linear-nearest-neighbor CNOT* gate, and parts of *Grover's quantum search* algorithm. However, this work is still restricted to reversible non-properly-quantum circuits, or properly-quantum with a small number of gates.

One of the earliest efforts on formally modeling and verifying quantum systems is the work of Gay and Nagarajan [19], where they developed a process algebra for *communicating quantum processes* (CQP) on top of pi-calculus [59]. In particular, they developed special quantum semantics and *behavioral transitions rules* to capture quantum communication concepts and types, e.g., *quantum bits, unitary operator,*

statistical measurement and no-cloning property. Using such algebra, they verified the behavior of the key distribution protocol BB84. Many work emerged out of this interesting work in the area of quantum cryptography.

Quantum cryptography is another area where model-based verification techniques have been applied since it is abstracted to the quantum bits model, and the quantum state evolution is restricted. For instance in [20], [2] and [15], the authors are proposing model checkers for *quantum communication protocols*, each is based on a different representation of the model of cryptography protocols. For instance, [20] uses communicating quantum processes (CQP) and [2] uses a quantum programming language (QPL), whereas [15] uses quantum Markov chain (QMC) for modeling quantum protocols. All of them, however, are using quantum computation tree temporal logics (QCTL) to write the protocols specifications. As benchmark applications, the effectiveness of each model checker has been shown by verifying the correctness of one or two major cryptography applications: e.g., [15] verifies *super-dense coding* and *quantum key distribution protocols*, [2] verifies the *bit blip error correction code protocol* and *teleportation protocol*, and [20] verifies the *quantum coin-flipping protocol*.

We believe that the most related work to our is [17], where the process algebra of CQP [19] is extended to include quantum linear optical concepts, e.g., *beam splitter*. Then using these concepts, the behavior correctness of a linear quantum optical gate, the single-photon CNOT gate, was tackled. Although the work of [17] formalizes quantum optical components and gates (as we propose to do in our work), it cannot handle the satisfiability of optical physical properties. For instance the authors only consider beam splitters of real parameters, whereas our work proposes to formalize beam splitters of complex parameters. This is due to the limited semantics of CQP since it is designed to work at the level of behavioral models.

In a nutshell, most of the prior research in the formal analysis and quantum theory is centralized around quantum information and quantum cryptography where abstraction techniques can work well and avoid the exponential behavior of classical computation of quantum physics. To the best of our knowledge, there is no work that tackles the formal analysis of quantum optics at the quantum mechanics level where physics properties, such as *optical coherence*, can be investigated and complicated devices, e.g., *optical displacer*, can be modeled.

1.3 Quantum Optics Analysis Framework

In this thesis, we propose to adopt the HOL Light theorem prover in the analysis of quantum optical systems. Formalizing quantum optics in HOL Light, however, requires the formalization of quantum mechanics preliminaries, which themselves require the development of infinite dimension complex-valued function linear spaces. In particular, it requires the formalization of Hilbert space L^2 [5] that contains square integrable complex-valued functions, which typically describes the physical state of a quantum system. The formalization of such theory has been considered in different theorem provers. There currently exist only four significant formalizations of linear algebra: two in HOL-Light ([29] and [40]), one in PVS [32], and one in Coq [73]. The three former focus essentially on n-dimensional Euclidean and complex spaces, whereas our formalization requires infinite-dimension vector spaces of complex numbers (more precisely, complex-valued-function spaces). The work in [73], and similarly in the Metamath [58] and Mizar [13] provers, formalize extensively a chapter of a classical textbook but, as far as we know, they do handle many other useful concepts like operator algebra, linear operators, Hermitian adjoints, eigenvectors or inner product:

(in the case of Metamath, it has the inner product but not the remaining). In addition, the proofs developed under the three former theorem provers are lengthy, and the provers themselves do not have built-in automation, and do not allow user-automation [77]. In a nutshell, the essential difference between these works and ours is that ours is oriented towards quantum applications rather than a systematic formalization of a textbook. In addition, it benefits from the HOL provers' advantages, e.g., built-in automation [77].

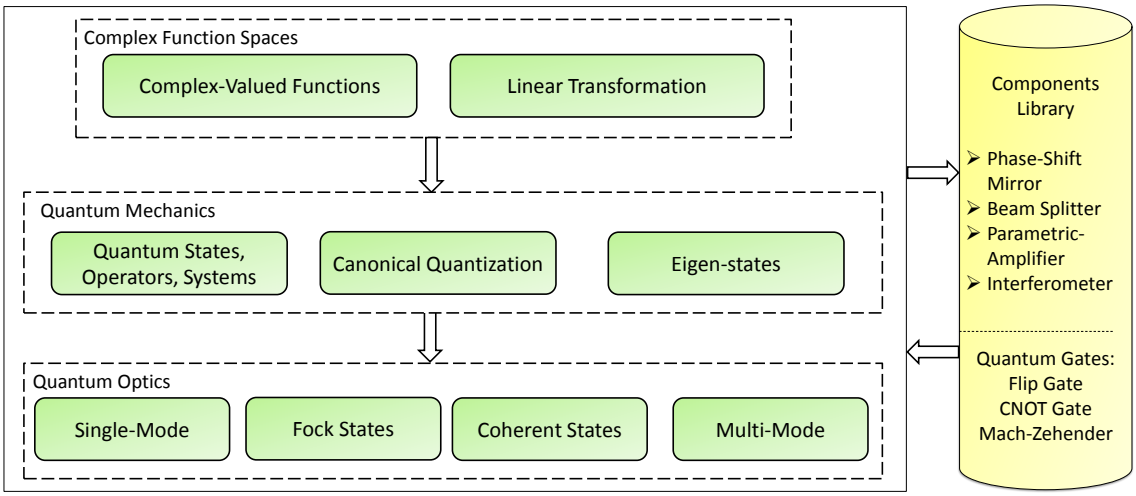

Figure 1.1: Quantum Optics Analysis Framework

In this thesis, we propose to build a formal analysis framework which encompasses the formalization of both mathematics and physics foundational theories of quantum optics in HOL Light (see Figure 1.1). The framework consists of two parts: on the left side, we have what we call the theoretical part; it does not deal with devices and circuits. This part is formed by three major libraries: 1) Complex function spaces, which formalizes function spaces, linear operators and their algebra, and infinite summation over complex functions. These notions form the mathematical foundation of the quantum theory; 2) Quantum mechanics, in which we define notions of *quantum*

and eigen states of a system, *systems observation* (or quantum operators), and how we can convert a system from classical to quantum paradigm using *canonical quantization*; and 3) Quantum Optics, which contains the formalization of photons in both *Single mode* (which corresponds for single-input/single-output systems), and *Multi-mode*. In addition, we study some special cases of light beams, in particular *coherent states*, which are typically emitted by laser sources, and *fock states*, which describe light beams that contain a deterministic number of photons. These notions will be covered later in detail.

On the right side, we have the components library. This is more oriented for quantum computing circuits verification. Similar to classical computers, quantum computers consist of a notion of bits, called quantum bits (abbreviated as *qbit*), and a set of quantum gates that perform processing over qbits, e.g., the flip gate (the quantum counterpart of the classical NOT gate) [61]. As a first step towards building such a library, in this thesis we have considered the verification of three optical quantum gates that are realized using coherent light [66] and single-photon [38] technologies. Namely, we formally verify the *flip gate, controlled NOT gate* and *Mach-Zehender interferometer* which are based on the optical devices: *beam splitters*, *phase conjugating mirrors*, *phase shifters*. We believe the theoretical part is rich enough to cover other components and circuits.

1.4 Thesis Contributions

The main contribution of this thesis [1] is the application of theorem proving in the area of quantum optics which was firstly proposed in [Bio-Cf7]. In particular, we

[1] References in this section are available in the author's Biography provided at the end of the document.

develop a formal analysis framework in the HOL Light theorem prover that covers both theoretical and physical aspects of quantum optics. This thesis is part of a large project [2] about the formalization of the physics of optics in the HOL Light theorem prover [42]. In the following, we list the contributions of this thesis, which focuses on the formalization of the theory of quantum optics:

- Formalization of the infinite/finite complex-valued function spaces which have a wide range of applications in mathematics and engineering, e.g., Fourier analysis, electromagnetic. It provides several useful notions, e.g., self-adjointness, closure and linearity, L^2 Hilbert space [Bio-Cf6]. In fact, the developed library became part of the latest HOL Light theorem prover release [27].

- A customizable quantum mechanics HOL Light library, in which we provide the bases of quantum theory, e.g., quantum states, quantum operators, etc. This allows the study of several systems in the quantum paradigm since they inherit the same rules. The optical beam is a typical example which in turn results in the formal theory of quantum optics [Bio-Jr2].

- A quantum optics library that contains the major concepts, namely single-mode theory that models the single-input/single-output optical systems, special optical quantum states, fock state and coherent states [Bio-Cf5]. In addition, it provide the multi-mode theory that generalize all these concepts for multi-input/multi-output optical systems [Bio-Jr3].

- The formalization of a number of important optical devices: beam splitters, optical displacer, optical phase shifter, and mirrors. Using these devices, we

[2] `http://hvg.ece.concordia.ca/projects/optics/`

formally built and verified the behavior of three quantum gates: flip gate [Bio-Cf3], controlled NOT gate, and Mach-Zehnder Interferometer [Bio-Cf1].

- Throughout our formalization, we have developed a number of tactics (theorem provers' utility functions that automate the proof steps or parts of it). These tactics help reducing the proof scripts. For example, our code for the complex-valued function arithmetics has been reduced from more than 300 lines of code to around 50 lines [Bio-Jr3]. In other cases, they facilitate and speed up the verification of quantum optics application, e.g., the controlled NOT gate and Mach-Zehnder Interferometer [Bio-Cf1].

The complete HOL script developed in this thesis is available through the project web page at [55].

1.5 Thesis Organization

The rest of the thesis is organized as follows: In Chapter 2, we provide a brief overview of quantum theory starting from the preliminaries of quantum mechanics to quantum optics, where a light beam is considered a quantum system. We also provide an introduction to theorem proving and higher-order logic notations that are used in our development.

In Chapter 3, we present the formalization of infinite/finite complex-valued function spaces theory, where we implement the linear transformation over such linear spaces, and then extend such spaces to inner product ones, where quantum states reside. In addition, we develop some interesting operators, e.g., self-adjoint and Hermitian operators. This chapter contains all the required mathematical notions for formalizing

quantum theory.

Chapter 4 includes the main objective of the thesis, where the preliminaries of quantum mechanics are formalized and then extended to implement different quantum optics notions. Basically, we build the general quantum mechanics rules and, in particular, we define the concepts of quantum states, operators. We then customize the general rules for optical beams, where we formalize single-mode fields which mimic the single-input/single-output optical systems and multi-mode fields in order to deal with multi-input/multi-output optical systems.

Finally, in Chapter 5, we show the practicality of the proposed framework in the formal modeling and analysis of optical circuits, in particular quantum computing circuits. We tackle the formalization of seven different applications. We start with single-mode optical devices, e.g., *optical phase shifters* and *the flip gate*. Next, we address the formalization of multi-mode optical elements, e.g., *beam splitters* and *the Controlled NOT gate*.

Chapter 6 concludes the thesis by providing some facts about the developed framework that include the merits and challenges of this work, and what are the future perspectives and directions.

Chapter 2

Preliminaries

The development of quantum theory is highly dependent on linear algebra aspects, thereby we briefly introduce them in the first section of this chapter (readers who are familiar with such concepts, can skip it). The second part of this chapter provides an introduction to quantum theory, including both quantum optics and quantum mechanics. In the last part of this chapter, we briefly introduce higher-order logic and theorem provers, and provide a list of symbols used in the rest of the thesis.

2.1 Linear Algebra Aspects

This section briefly lists all linear algebra definitions that are being used throughout the thesis, namely linear space, inner product, linear transformation, functions integrability and measures.

Definition 2.1. *A vector linear space over a field F (typically, in our case, $\mathbb{C}$) is a set V of vectors with an operation $+ : V \times V \rightarrow$ and $* : F \times V \rightarrow V$.*

Example 2.1.1. *The set of complex-valued functions forms a vector space where $+$*

is defined as: $f1 + f2 = \lambda\ x.f1(x) + f2(x)$ *and* $*$ *is defined as:* $a * f = \lambda\ x.a * f(x)$

Definition 2.2. *A function* $L : V \to V$ *is called a linear transformation iff:*

1. $\forall x, y \in V.\ L(x + y) = L(x) + L(y)$.
2. $\forall x \in V, c \in F.\ L(c * x) = c * L(x)$.

For a vector space V over a complex field F, $\lambda \in F$ and $\nu \in V$ are an *eigenvalue* and an *eigenvector*, respectively, of the linear transformation L iff $L(\nu) = \lambda * \nu$ and $\nu \neq 0$.

Definition 2.3. *A dual space of* V *is a set of all linear transformations over* V*. Dual space is a linear function space that preserves the same properties as Definition 2.1.*

Definition 2.4. *A vector space* V *over the complex field* $\mathbb{C}$ *is an inner product space iff there is a function* $I : V \times V \to \mathbb{C}$*, called an* inner product*, which satisfies the following:*

1. *Conjugate symmetry:* $\forall x\ y \in V.\ I(x, y) = \overline{I(y, x)}$.
2. *Linearity(1):* $\forall x, y \in V, a \in \mathbb{C}.\ I(x, a * y) = a * I(x, y)$.
3. *Linearity(2):* $\forall x, y, z \in V.\ I(x + y, z) = I(x, z) + I(y, z)$.
4. *Positive-definiteness:* $\forall x \in V.\ I(x, x) \geq 0$ *(it is clear from 1 that* $I(x, x) \in \mathbb{R}$*). It can be proved that the operation* $\sqrt{I(x, x)}$ *satisfies the axioms of a norm. We thus write this value as* $\parallel x \parallel$.

According to inner product properties, we can prove the Schwarz inequality:

$$\forall f, g \in V.\ \parallel f \parallel * \parallel g \parallel \geq |I(f, g)|^2$$

For a linear transformation H, $H^\dagger$ is called the *hermitian* of H iff:

$$\forall x, y \in V. I(x, H(y)) = I(H^\dagger(x), y)$$

The linear transformation is called a *self-adjoint* (or hermitian operator) iff:

$$\forall x, y \in V. I(x, H(y)) = I(H(x), y)$$

Definition 2.5. *A Hilbert space $\mathcal{H}$ is a complete inner product space iff for every infinite sequence of vectors $\sum_0^\infty x_i : \sum_0^\infty \parallel x_i \parallel < \infty \Rightarrow \sum_0^\infty x_i \in \mathcal{H}$.*

Example 2.1.2. *Consider the set of all square integrable complex-valued functions, i.e., such that: $\int_{-\infty}^{\infty} f^*(\overrightarrow{x}) f(\overrightarrow{x}) d\overrightarrow{x} < \infty$. It is a Hilbert space, called L^2, with inner product Inner: $\forall f, g \in L^2. Inner(f, g) = \int_{-\infty}^{\infty} f^*(\overrightarrow{x}) g(\overrightarrow{x}) d\overrightarrow{x}$*

Definition 2.6. *Given a set σ of subsets of a set A, which is closed under set-operations, then a function $\mu : (A \longrightarrow bool) \longrightarrow \mathbb{R}$ is measure of σ iff:*

1. $\forall E. E \in \sigma \Rightarrow \mu(E) \geq 0$.
2. $\mu(\phi) = 0$.
3. $E_{i \in \mathbb{N}} \in \sigma$ *are pairwise disjoint sets* $\Rightarrow \ \mu(\bigcup_{i \in \mathbb{N}} \ E_i) = \sum_{i \in \mathbb{N}} \mu(E_i)$.

If such μ exists the A is a measurable set.

2.2 Quantum Mechanics

Any physical system has a mathematical model that describes its *state*. In classical physics, a system state can be deterministically evaluated at any time, e.g., the position equation of a moving particle gives the precise position at any time. However, in quantum theory, a system state has a probabilistic nature. According to Dirac [23], a *quantum state* is a complex-valued function (i.e., of type $A \longrightarrow \mathbb{C}$, where A is an abstract object that contains the basic system parameters) and is written as $|\psi\rangle$. The quantum state squared norm, i.e., $\parallel |\psi\rangle * |\psi^*\rangle \parallel$, forms a probability density function, which is the source of indeterminism in quantum theory. Note that $|\psi^*\rangle$ is a function

that returns the complex conjugate of $|\psi\rangle$, given certain parameters A.

The set of all possible quantum states forms an infinite dimensional inner-product functions space (in mathematics this is called L^2 Hilbert space). Recall that an infinite dimension function space is a linear subspace, the bases of which are countably infinite. The inner product space is a linear space on which we can define a product function that receives, in our case, two quantum states and returns a complex value. In other words, it has the type $(A \rightarrow \mathbb{C}) \times (A \rightarrow \mathbb{C}) \rightarrow \mathbb{C}$. This product should satisfy certain properties (according to Dirac, the inner product of two quantum states can be written as $\langle\phi|\psi\rangle$):

- Conjugate-Symmetry: $\langle\phi|\psi\rangle = (\langle\psi|\phi\rangle)^*$.
- Linearity-Addition: $\langle\phi|\psi_1 + \psi_2\rangle = \langle\phi|\psi_1\rangle + \langle\phi|\psi_2\rangle$.
- Linearity-Scalar Multiplication: $\langle a * \phi|\psi\rangle = a * \langle\phi|\psi\rangle$.
- Positive-definiteness: $0 \leq \langle\psi|\psi\rangle$

In quantum mechanics, this product function is typically a Lebesgue integral. This integral varies based on the system constructing variables (or coordinates), i.e., it is a single-integral for single-coordinate systems, and double integral for two-coordinate systems. However, for any system, quantum states always should be normalized, i.e., $\langle\psi|\psi\rangle = 1$.

Now, we have quantum states that contain probabilistic information about the system subject to study. To enquire about specific information, e.g., particle positions or velocity, Dirac defines the notion of quantum observable (or operators), written as $\hat{O}$. A quantum operator is a linear self-adjoint mapping function over the quantum states space, i.e., of the type $(A \rightarrow \mathbb{C}) \rightarrow (A \rightarrow \mathbb{C})$:

- Linearity-Addition: $\hat{O}(|\psi_1\rangle + |\psi_2\rangle) = \hat{O}\ |\psi_1\rangle + \hat{O}\ |\psi_2\rangle$.
- Linearity-Scalar Multiplication: $\hat{O}(a * |\psi\rangle) = a * (\hat{O}|\psi\rangle)$.
- Self-adjointness: $\langle \hat{O}\ \phi|\psi\rangle = \langle\phi|\hat{O}\ \psi\rangle$

Since quantum states are probabilistic, the measurement of such observables is also probabilistic. Hence, we cannot calculate the measurement precisely, but rather its *expectation*, and evaluate the error (or precision) of the measurement using the notion of *variance*. In Dirac notation, the measurement expectation and variance, at a state $|\psi\rangle$, are defined as follows:

$$E[\hat{O}] = \langle\psi|\hat{O}\ \psi\rangle \tag{2.1}$$

$$V[\hat{O}] = E[(\hat{O} - E[\hat{O}])^2] = \langle\psi|(\hat{O} - E[\hat{O}])^2\ \psi\rangle. \tag{2.2}$$

In a nutshell, we can summarize the quantum mechanics primitives as follows:

- A quantum state (and, more generally, any element of L^2 space) is written as $|\psi\rangle$.
- The inner product between two different states $|\phi\rangle$ and $|\psi\rangle$ is written as $\langle\phi|\psi\rangle$.
- Operators are generally written as: $\hat{o},\ \hat{p},\ \hat{q}, ...$
- The application of an operator $\hat{O}$ to a state $|\psi\rangle$ is simply written $\hat{O}|\psi\rangle$.
- For any quantum state $|\psi\rangle$: $\langle\psi|\psi\rangle = 1$.
- The application of an observable $\hat{O}$ to $|\psi\rangle$ is also an element of L^2 space, so we write it as: $|\hat{O}\psi\rangle$.

- A self-adjoint operator has the following property: $\langle\phi|\hat{O}\ \psi\rangle = \langle\hat{O}\ \phi|\psi\rangle$.
- An operator $\hat{O}_1$ is the Hermitian of the operator $\hat{O}_2$ iff $\langle\phi|\hat{O}_1\ \psi\rangle = \langle\hat{O}_2\ \phi|\psi\rangle$. Note that self-adjointness is a special case of the Hermitian relation.
- The expectation of an observable $\hat{O}$ is written as $\langle\hat{O}\rangle = \langle\psi|\hat{O}\ |\psi\rangle$.

2.3 Quantum Optics

In light of previous elementary rules, we can study the optical beam as a quantum system, which deals with optics as a stream of particles called photons, in contrast with classical optics theory, which considers it as a ray or electromagnetic wave. For systems studied based on classical physics, quantum mechanics capitalizes on what already exists, and tries to convert it to the quantum paradigm through a process called *canonical quantization* [12]. In this process, the system to be converted is described by a set of special observables, called canonical coordinates, and a *Hamiltonian* observable that expresses the total energy in the system. Two observables, $\hat{A}$ and $\hat{B}$, are called canonical if their commutator is equal to $i\hbar$, i.e., $\hat{A}\ \hat{B} - \hat{B}\ \hat{A} = i\hbar$, where $\hbar$ is the Planck constant [33]. Usually, the operation $\hat{A}\ \hat{B} - \hat{B}\ \hat{A}$ is denoted as $[\hat{A}, \hat{B}]$. Typical examples of canonical coordinates (or observables) are the position and momentum of a moving particle. In the following, we will apply the canonical quantization on the *single-mode* electromagnetic field, which typically mimics a single optical beam and single-input/single-output optical systems. Then, we present the *multi-mode* fields that cover the case of multi-input/multi-output systems.

2.3.1 Single-Mode Fields

The coordinate of a single optical beam is typically the amount of charges $\hat{q}$ inside the beam. Accordingly, we can select the corresponding inner product of the optical quantum states space as the complex Lebeuage integral. For instance, the inner product of ψ_1 and ψ_2 is $\int_{-\infty}^{\infty} \psi_4 1^*(q)\ \psi_2(q)dq$. Hence, the optical quantum states are normalized as follows:

$$\int_{-\infty}^{\infty} \psi^*(q)\ \psi(q)dq = \int_{-\infty}^{\infty} |\psi(q)|^2 = 1 \tag{2.3}$$

where $|\psi(q)|$ is the norm of complex value $\psi(q)$.

For the sake of canonical quantization, we identify the intensity of flux $\hat{p}$ as the canonical coordinate of q, and the Hamiltonian function is:

$$\hat{H} = \frac{\omega^2}{2}\hat{q}^2 + \frac{1}{2}\hat{p}^2 \tag{2.4}$$

where ω is the resonance frequency. Using Equation (2.4), the property of canonical coordinates, and the quantum mechanics rules, we can prove that an optical beam contains energy even though there is no charge or flux. This theorem does not have a classical counterpart. In the following, we will present the proof of this theorem, and at the same time we will introduce some important notions.

The main idea in the proof of the minimum energy theorem is to express the system energy $\hat{H}$ in terms of two important quantum operators, namely *annihilator* $\hat{a}$ and *creator* $\hat{a}^\dagger$: First, we define a set of new quantum operators to simplify our proof as follows:

$$\hat{a}\ = \sqrt{\frac{\omega}{2\hbar}}\hat{q} + i\sqrt{\frac{1}{2\omega\hbar}}\hat{p} \tag{2.5}$$

$$\hat{a}^\dagger = \sqrt{\frac{\omega}{2\hbar}}\hat{q} - i\sqrt{\frac{1}{2\omega\hbar}}\hat{p} \tag{2.6}$$

A justification of this naming will be provided later. Note that $\hat{a}$ is the Hermitian of $\hat{a}^\dagger$.

Given that $\hat{q}$ and $\hat{p}$ are the canonical coordinates of the system, i.e., $[\hat{q}, \hat{p}] = i\hbar$, we can prove that the commutator $[\hat{a}, \hat{a}^\dagger] = 1$. This leads to the rewriting of the energy operator in terms of the creator and annihilator we are looking for:

$$\hat{H} = \hbar\omega\left(\hat{a}^\dagger\hat{a} + \frac{1}{2}\right) \tag{2.7}$$

Then, the measurement expectation of the energy $\hat{H}$ at the quantum state $|\psi\rangle$ is expressed according to the rules presented in the previous section as follows:

$$\langle\psi|\hat{H}|\psi\rangle = \langle\psi|\hbar\omega\left(\hat{a}^\dagger\hat{a} + \frac{1}{2}\right)|\psi\rangle \tag{2.8}$$

Given that $\langle\psi|\psi\rangle = 1$ (because quantum states are normalized) and $\hat{a}^\dagger$ is the Hermitian of $\hat{a}$, we can conclude that:

$$\langle\psi|\hat{H}|\psi\rangle = \hbar\omega\langle\hat{a}\psi|\hat{a}|\psi\rangle + \frac{\hbar\omega}{2} \tag{2.9}$$

Since $\langle\hat{a}\psi|\hat{a}\psi\rangle$ is a positive real number, i.e., $\langle\hat{a}\psi|\hat{a}\psi\rangle \geq 0$, then:

$$\langle\psi|\hat{H}|\psi\rangle \geq \frac{\hbar\omega}{2} \tag{2.10}$$

where $\frac{\hbar\omega}{2}$ is called the zero point energy. The corresponding practical meaning of such a result in quantum optics is that energy always exists, even in the absence of photons.

Optical States and Operator

Given that quantum states form a linear function space, then indeed there is a set of independent quantum states that span the whole space (i.e., the basis). These states

are typically called *pure states*. At any time, the optical beam is described either by a pure state or a mixed one, which is expressed as follows:

$$|\psi\rangle = \sum |c_i| * |\psi\rangle_i \; i = 0, 1, 2, ... \tag{2.11}$$

where c_i is a complex number, $\sum |c_i|^2 = 1$ and $|\psi\rangle_i$ is a pure state. A system is at a pure state i if $c_i = 1$ and for any $j \neq i, c_j = 0$.

In quantum optics, the set of such pure states are called *fock states*. An optical beam in a fock state $|n\rangle$, where $n = 0, 1, 2...$, means that the light stream exactly contains n photons. The special case of $|0\rangle$ represents a vacuum state where there are no photons but energy of $\frac{\hbar\omega}{2}$, as proved earlier.

Another interesting quantum state is coherent light. Typically, the number of photons in a coherent light stream is probabilistically Poisson distributed. In other words, the probability of having (or observing) n photons is:

$$P(N = n) = \frac{|\alpha|^n \; e^{|\alpha|}}{n!} \tag{2.12}$$

where $|\alpha|$ is the expected number of observed photons (α is a complex number). A coherent light with expected photons $|\alpha|$ is in the quantum state $|\alpha\rangle$. Such a state is expressed in terms of the basis fock states as follows (see Equation (2.11)):

$$|\alpha\rangle = e^{-\frac{|\alpha|^2}{2}} \sum_{n=0} \frac{\alpha^n}{\sqrt{n!}} |n\rangle \tag{2.13}$$

The effect of the *creator* and *annihilator* operators, defined in Equations (2.5) and (2.6), on fock and coherent states is crucial. Their names suggest how these operators affect a stream of photons. An annihilator $\hat{a}$ decreases the number of photons by one (i.e., destroys a photon):

$$\hat{a}|n\rangle = \sqrt{n}|n-1\rangle \tag{2.14}$$

Note that the resulting quantum state is not exactly the demoted one; it is scalar-multiplied by $\sqrt{n}$. Similarly, the creation $\hat{a}^\dagger$ increases the number of photons by one (i.e., creates a photon):

$$\hat{a}^\dagger|n\rangle = \sqrt{n+1}|n+1\rangle \tag{2.15}$$

It is important to mention here that the scalar-multiplication does not change the behavior of a quantum state. Thereby, the resulting states in (2.14) and (2.15) still have $n-1$ and $n+1$ photons, respectively.

Solving Equation (2.15) as a recurrence relation, we obtain a general representation of any fock state $|n\rangle$:

$$|n\rangle = \frac{(\hat{a}^\dagger)^n\ |0\rangle}{\sqrt{n!}} \tag{2.16}$$

where $|0\rangle$ is called a vacuum state since it does not contain any photon. Note here that the power notation used in $(\hat{a}^\dagger)^n$ means the application of the creation operator n times (Recall that quantum operators are functions).

According to 2.13 and 2.16, we can re-express the coherent state in terms of the vacuum state and creation operator:

$$|\alpha\rangle = e^{-\frac{|\alpha|^2}{2}} \left(\sum_{n=0} \frac{(\alpha\hat{a}^\dagger)^n}{n!} \right) |0\rangle \tag{2.17}$$

Note that for a linear operator $a^\dagger$, $(\alpha\hat{a}^\dagger)^n = \alpha^n(\hat{a}^\dagger)^n$.

Given the definition of coherent states in Equation 2.13 and the annihilator effect in Equation 2.14, we can deduce the effect of the annihilator on any coherent state $|\alpha\rangle$:

$$\hat{a}|\alpha\rangle = \alpha|\alpha\rangle \tag{2.18}$$

Note that this result shows that coherent states are eigenstates of the annihilator operator.

2.3.2 Multi-Mode Fields

All the above mentioned definitions, formulas and equations form the single-mode optical beams theory [56]. This theory is suitable as long as we are dealing with systems that involve no more than one single beam. In order to tackle more general systems with multiple optical beams, we should consider the theory of multi-modes. The core idea is how to consider two independent optical beams (or particles), given that we have the individual physical description of each. For this purpose, we utilize the mathematical tool of a *tensor product*. Let us assume the existence of two beams with quantum states $\big|\psi_1\big\rangle$ and $\big|\psi_2\big\rangle$, then we have a new quantum state $\big|\psi_1 \otimes \psi_2\big\rangle$ that describes both beams simultaneously. The new state satisfies the following properties:

$$\big|c * \psi_1 \otimes \psi_2\big\rangle = c * \big|\psi_1 \otimes \psi_2\big\rangle \text{ and}$$

$$\big|\psi_1 + \psi_2 \otimes \psi_3\big\rangle = \big|\psi_1 \otimes \psi_3\big\rangle + \big|\psi_2 \otimes \psi_3\big\rangle$$

For these kind of states, we need to develop suitable operators based on existing ones. For instance, for two annihilation operators we will have a new tensor product operator $\hat{a}_1 \otimes \hat{a}_2$, subscript refers to the modes to which they belong. This operator when it is applied to $\big|\psi_1 \otimes \psi_2\big\rangle$, results in $\big|\hat{a}_1\psi_1 \otimes \hat{a}_2\psi_2\big\rangle$. It also satisfies similar properties such as the tensor product of states, e.g., $(\hat{a}_1^\dagger + \hat{a}_1) \otimes \hat{a}_2^\dagger = \hat{a}_1^\dagger \otimes \hat{a}_2^\dagger + \hat{a}_1 \otimes \hat{a}_2^\dagger$.

Note: most of the formulas and definitions presented in the last two sections are taken from [56] which we believe to be a very comprehensive book that maintains both mathematical and physics aspects. Unlike other physics books that omit many details and assumptions, it provides a thorough explanation and goes step by step in particular for non-physicists.

In the next chapters we will tackle the formal development of all concepts and notions

presented in this section in the HOL Light theorem prover. In the sequel, we give a short introduction to higher-order logic and HOL Light theorem prover.

2.4 Theorem Proving

2.4.1 Higher-Order Logic

Given a logic, the most frequent problem is to try to determine whether a given sentence is true or not. This is done by considering a set of *axioms*, i.e., basic sentences that are assumed to be true (e.g., $P \vee \neg P$), and *inference rules*, i.e., rules that allow the truth of a sentence to be derived depending upon the truth of other sentences (e.g., if P and Q are true sentences, then $P \wedge Q$ is a true sentence). Using axioms and inference rules, one can thus *prove* or disprove the sentences of a logic. This idea is at the core of theorem proving: the language definition, the axioms and inference rules can be implemented in the theorem prover, which allows the user to write down mathematical sentences and then prove their correctness.

In order to reason about mathematical and physics theory, propositional logic is not sufficient where one needs to talk about the objects and properties of those, rather than simple logical statements. This problem can be partially answered by first-order logic (FOL) (also called predicate logic) that introduces *terms* (which formalize the notion of "object") and *predicates* (which formalize the notion of "property of an object"). In order to get even closer to the usual mathematical language it also introduces the notion of *variable*, and allows quantification over such variables: *for all* $\forall$ and *there exists* $\exists$. This allows the representation of simple mathematical objects, e.g., natural numbers. However, still others, e.g., real and complex numbers, cannot

be represented in FOL, which requires more expressive logic. Higher-order logic then answers this need for high expressiveness by expanding the concept of quantifiers over predicates and functions. This not only allows the formalization of complicated types, but also advanced theory, e.g., integration, differentiate, measure theory, limit and convergence, probability theory etc. Thereby, it becomes suitable for dealing with complex systems.

2.4.2 HOL Light Theorem Prover

HOL Light is a typed higher-order proving system, where every variable appears in an expression has a type. Sometimes, the prover users do not need to explicitly write the type if the type can be inferred from the expression context. In the following, we provide a couple of examples to illustrate how definitions and theorems are written in HOL Light. Before that, we refer to Table 2.1, which lists all HOL Light mathematical/logical symbols and operations that are being used through the thesis.

Given a function `f_max` that receives two real values as parameters and returns the maximum one, the corresponding HOL expression of such a function is as follows:

$$\texttt{f\_max} = \lambda\ \texttt{x} : \texttt{real}\ \texttt{y} : \texttt{real}.\ \texttt{if}\ \texttt{x} \geq \texttt{y}\ \texttt{then}\ \texttt{x}\ \texttt{else}\ \texttt{y}$$

The *if* statement is part of the HOL which allows to choose between two alternatives according to a given condition. For such a function, at calling time, it appears as `f_max a b`, which returns the maximum of given parameters `a` and `b`. If we specify the type of `f_max` in this expression, it will be $\texttt{real} \longrightarrow \texttt{real} \longrightarrow \texttt{real}$; however, the HOL Light type engine can infer it.

Operator	Symbol
Conjunction	$\wedge$
Disjunction	$\vee$
Logical negation	$\sim$
Logical implication	$\Rightarrow$
Logical equivalence (if and only if)	$\Leftrightarrow$
Universal quantification	$\forall$
Existential quantification	$\exists$
Choice operator	@ s
Lambda abstraction (required for functions definition)	λ
Number to Real Type casting operator	&
Real and complex power	`pow`
Scalar multiplication	%
Arithmetic negation	$--$
Operator multiplication	$**$
Lists	$[a; b; ..]$
Specify type operator	`A:real`
Function composition	`f o g`
Domain to Codomain	`A` $\longrightarrow$ `B`
Vectors lambda	`lambda x. v x`
Vector indexing operator	$

Table 2.1: HOL Light Symbols

Now, given the definition of `f_max`, one might be in favor of proving the transitivity of such a function, which can be expressed as a HOL theorem as follows:

$$\forall \texttt{a b c. f\_max a b} = \texttt{b} \wedge \texttt{f\_max b c} = \texttt{c} \Rightarrow \texttt{f\_max a c} = \texttt{c}$$

We can revisit the same example using a different approach. This time we will define a predicate which receives two real-value parameters, and returns true if the first is greater than or equal to the second one:

$$\texttt{is\_gt x y} \Leftrightarrow \texttt{if x} > \texttt{y then True else False}$$

Note the use of the equivalence symbol $\Leftrightarrow$ to define `is_gt` since it is a predicate (i.e., the return value is Boolean) not a function (typically returns types other than

Boolean). In this case, the type of `is_gt` is `real` $\rightarrow$ `real` $\rightarrow$ `bool`. This style of definition is a relational, and we usually use it when the concrete implementation of a function is not available but rather its specifications. For this definition, the transitivity theorem is expressed as follows:

$$\forall \texttt{a b c}.\ \texttt{is\_gt c b} \wedge \texttt{is\_gt b a} \Rightarrow \texttt{is\_gt c a}$$

This concludes the preliminary chapter. In the next chapters will investigate in detail the HOL formalization of many concepts that were introduced here. In the rest of the thesis, whatever appears under "`Theorem xx`" or "`Definition xx`" is the HOL implementation of the regular quantum optics theories as taken from reference textbooks (similar to what we have presented in Sections 2.1-2.3). Sometimes these definitions differ from the original mathematical representations. This is due to the replacement of regular mathematical operations with the corresponding HOL Light symbols (see Table 2.1), and the use of developed HOL definitions, which in most cases start with "`cfun_`", "`cop_`" and "`is_`". Whenever possible, after presenting a new HOL definition or theorem, we refer to the corresponding mathematical formula that is typically introduced in this chapter (Sections 2.1-2.3).

Chapter 3

Formalization of Complex-Valued Functions

In Chapter 2, we gave an introduction to quantum theory that shows how important and crucial complex-valued functions and related algebraic notions are for the mathematical formalization of quantum mechanics. This chapter covers in detail the higher-order formalization of the mathematical foundation of quantum theory, namely: finite/infinite complex-valued functions linear subspaces, the inner product over complex-valued functions, linear and self-adjoint transformations, limit and infinite summation of complex-valued functions. In the last part of this chapter, we discuss the implementation of a number of HOL tactics that we used to speed up theorems proving process.

3.1 Finite/Infinite Complex-Valued Functions Spaces

In order to consider both infinite and finite dimension complex linear spaces, we take the *function space* of an arbitrary set to `complex`. This is expressed by the type `cfun` = `A` $\longrightarrow$ `complex`, where `A` is a type variable (`cfun` stands for *c*omplex *fun*ction). This representation allows for both infinite-dimension linear spaces (by taking, e.g., `num` or `real` for `A`), and finite-dimension ones (by taking for `A` any type with a finite extension).

Recall that a linear space V over Field F is closed under the operations *Addition* $(+) : V \longrightarrow V \longrightarrow V$, and *Scalar Multiplication* $(\%) : F \longrightarrow V \longrightarrow V$. These two operations must satisfy certain properties, e.g., addition commutativity and associativity, multiplication distributivity over addition. Accordingly, we define these two operations for `cfun` as follows (note that in the case of `cfun`, $F = \mathbb{C}$):

Definition 3.1 (`cfun` arithmetic)**.**

`cfun_add` (`v`$_1$: `cfun`) (`v`$_2$: `cfun`) : `cfun` = λ`x` : `A`. `v`$_1$ `x` + `v`$_2$ `x`

`cfun_smul` (`a` : `complex`) (`v` : `cfun`) : `cfun` = λ`x` : `A`. `a` $*$ `v` `x`

Note that in our formalization, all definitions related to complex functions are prefixed with the term `cfun`. Using lambda calculus, `cfun_add` of functions `v`$_1$ and `v`$_2$ is defined as a new function that returns, at a given point `x`, the complex addition of `v`$_1$ `x` and `v`$_2$ `x`. Similarly, `cfun_smul` is defined; however, in this case we have only one function as an input and a complex number, and the complex multiplication is used instead. For convenience, we also define the commonly used operations of negation, subtraction and conjugation, as well as the null function:

Definition 3.2.

`cfun_neg` (`v` : `cfun`) : `cfun` = `cfun_smul` (−`Cx`(&1)) `v`

`cfun_sub` $(\mathtt{v}_1 : \mathtt{cfun})$ $(\mathtt{v}_2 : \mathtt{cfun})$ $: \mathtt{cfun} = \mathtt{cfun\_add}\ \mathtt{v}_1\ (\mathtt{cfun\_neg}\ \mathtt{v}_2)$

`cfun_cnj` $(\mathtt{v} : \mathtt{cfun})$ $: \mathtt{cfun} = \lambda \mathtt{x} : \mathtt{A}.\ \mathtt{cnj}\ (\mathtt{v}\ \mathtt{x})$

`cfun_zero` $= \lambda \mathtt{x} : \mathtt{A}.\ \mathtt{Cx}(\&0)$

where `&` is the HOL-Light function injecting natural numbers into reals, and `Cx` injects real numbers into complex numbers. Based on these definitions, we prove that they satisfy the usual axioms of a linear space (see Table 3.1).

Property	HOL Theorem
SYMMETRY	$\forall \mathtt{x}\ \mathtt{y} : \mathtt{cfun}.\ \mathtt{x} + \mathtt{y} = \mathtt{y} + \mathtt{x}$
ASSOCIATIVITY	$\forall \mathtt{x}\ \mathtt{y}\ \mathtt{z} : \mathtt{cfun}.\ (\mathtt{x} + \mathtt{y}) + \mathtt{z} = \mathtt{x} + \mathtt{y} + \mathtt{z}$
DISTRIBUTIVITY 1	$\forall (\mathtt{a} : \mathtt{complex})\ \mathtt{x}\ \mathtt{y}.\ \mathtt{a}\ \% (\mathtt{x} + \mathtt{y}) = \mathtt{a}\ \% \mathtt{x} + \mathtt{a}\ \% \mathtt{y}$
DISTRIBUTIVITY 2	$\forall (\mathtt{a}\ \mathtt{b} : \mathtt{complex})\ \mathtt{x}.\ (\mathtt{a} + \mathtt{b})\ \% \mathtt{x} = \mathtt{a}\ \% \mathtt{x} + \mathtt{b}\ \% \mathtt{x}$
DISTRIBUTIVITY 3	$\forall (\mathtt{a}\ \mathtt{b} : \mathtt{complex})\ \mathtt{x} : .\ \mathtt{a}\ \% (\mathtt{b}\ \% \mathtt{x}) = (\mathtt{a} * \mathtt{b})\ \% \mathtt{x}$
IDENTITY ADDITION	$\forall (\mathtt{x} : \mathtt{cfun}).\ \mathtt{x} + \mathtt{cfun\_zero} = \mathtt{x}$
ADDITIVE INVERSE	$\forall (\mathtt{x} : \mathtt{cfun}).\ \mathtt{x} - \mathtt{x} = \mathtt{cfun\_zero}$

Table 3.1: `cfun_add` and `cfun_smul` Properties

Accordingly, we can define the notion of sub-linear space of `cfun` as follows:

Definition 3.3.

$\mathtt{is\_cfun\_subspace}\ (\mathtt{spc} : \mathtt{cfun} \to \mathtt{bool}) \Leftrightarrow$

$\quad \mathtt{cfun\_zero\ IN\ spc}\ \wedge \forall \mathtt{x}.\ \mathtt{x\ IN\ spc} \Rightarrow$

$\quad\quad (\forall \mathtt{a}.\ \mathtt{a}\ \%\ \mathtt{x\ IN\ spc}) \wedge \forall \mathtt{y}.\ \mathtt{y\ IN\ spc} \Rightarrow \mathtt{x} + \mathtt{y\ IN\ spc}$

The above predicate identifies a set of complex-valued functions as a linear subspace iff it contains the identity element `cfun_zero` and closed under addition and scalar multiplication, as explained earlier.

Now, we have accomplished the building of the first block in the complex-valued functions formalization. The next step is to define operators, in particular linear transformations, over a function linear subspace. These will serve later as quantum operators (see Chapter 2).

3.1.1 Linear Operators

A very important notion is the one of transformation between vector spaces. Such a transformation is called an *operator*. In the context of complex-valued functions `cfun`, an arbitrary operator between two different spaces has the type
$\texttt{cop} = (\texttt{A} \longrightarrow \texttt{complex}) \longrightarrow (\texttt{B} \longrightarrow \texttt{complex})$, for which we define the following standard operations:

Definition 3.4.

$\texttt{cop\_add}\ (\texttt{op}_1 : \texttt{cop})\ (\texttt{op}_2 : \texttt{cop})\ : \texttt{cop} = \lambda \texttt{x}.\ \texttt{op}_1\ \texttt{x} + \texttt{op}_2\ \texttt{x}$

$\texttt{cop\_smul}\ (\texttt{a} : \texttt{complex})\ (\texttt{op} : \texttt{cop})\ : \texttt{cop} = \lambda \texttt{x}.\ \texttt{a}\ \%\ \texttt{op}\ \texttt{x}$

$\texttt{cop\_neg}\ (\texttt{v} : \texttt{cop})\ : \texttt{cop} = \texttt{cop\_smul}\ (-\texttt{Cx}(\&1))\ \texttt{v}$

$\texttt{cop\_sub}\ (\texttt{v}_1 : \texttt{cop})\ (\texttt{v}_2 : \texttt{cop})\ : \texttt{cop} = \texttt{cop\_add}\ \texttt{v}_1\ (\texttt{cop\_neg}\ \texttt{v}_2)$

The above is very similar to the `cfun` operators defined in the previous section but with different types. An essential aspect of operators that do not have the `cfun` counterpart, is the fact that we can *multiply* two `cop` operators. This multiplication is simply functions composition:

Definition 3.5.

$\texttt{cop\_mul}\ (\texttt{op}_1 : (\texttt{A} \longrightarrow \texttt{complex}) \longrightarrow (\texttt{B} \longrightarrow \texttt{complex}))$
$\qquad (\texttt{op}_2 : (\texttt{C} \longrightarrow \texttt{complex}) \longrightarrow (\texttt{A} \longrightarrow \texttt{complex})) = \lambda \texttt{x}.\ \texttt{op}_1\ (\texttt{op}_2\ \texttt{x})$

Note that the domain of $\texttt{op}_1$ and the codomain of $\texttt{op}_2$ must be the same. Following the conventions applied in HOL-Light for matrix multiplication, this operation is denoted with the infix $**$. Indeed, one can recognize that, when the operator is linear, then the operators amount to matrices in finite dimension.

This multiplication has unusual properties, starting with the fact that it is not commutative. It follows that many results that are intuitively true in other contexts are actually false here. For instance, multiplication is only right-distributive over addition, i.e., the following holds:

Theorem 3.1. $\forall \mathtt{op}_1\ \mathtt{op}_2\ \mathtt{op}_3.\ (\mathtt{op}_1 + \mathtt{op}_2)\ **\ \mathtt{op}_3 = \mathtt{op}_1\ **\ \mathtt{op}_3 + \mathtt{op}_2\ **\ \mathtt{op}_3$

But the following does not:

$$\forall \mathtt{op}_1\ \mathtt{op}_2\ \mathtt{op}_3.\ \mathtt{op}_3\ **\ (\mathtt{op}_1 + \mathtt{op}_2) = \mathtt{op}_3\ **\ \mathtt{op}_1 + \mathtt{op}_3\ **\ \mathtt{op}_2$$

Another interesting operation defined over `cop` is the exponentiation, which is equivalent to applying the operator n times:

Definition 3.6.

$\mathtt{cop\_pow}\ (\mathtt{op} : \mathtt{cfun} \longrightarrow \mathtt{cfun})\ \mathtt{0}\ =\ \mathtt{I} \wedge$

$\mathtt{cop\_pow}\ \mathtt{op}\ (\mathtt{SUC}\ \mathtt{n}) = \mathtt{op}\ **\ (\mathtt{cop\_pow}\ \mathtt{op}\ \mathtt{n})$

where `I` is the identity operator, i.e., `I x = x`, and `SUC n` is equal to $n + 1$. The exponentiation (or `cop_pow`) is defined recursively: the base case zero means applying the operator zero times, which is equivalent to identity operator that has no effect. Note that the domain and codomain of `op` should be the same.

We proved numerous theorems for the operations defined for the type `cop`. We list here some examples in the following table:

Linear operators are of particular interest in our work since quantum operators are linear. They correspond, in the finite-dimension case, to matrices. A `cop` operator is called linear iff it satisfies the following two properties:

Property	HOL Theorem
COP_MUL_LID	$\forall$ op : cop. op ** I = op
COP_ADD_RDISTRIB	$\forall$ (a b : complex) op : cop.(a + b) % op = a % op + b % op
COP_POW_I	$\forall$ n. I cop_pow n = I
COP_SMUL_SYM	$\forall$(a b : complex) op : cop. a % (b % op) = b % (a % op)
COP_POW_COMMUTE_N	$\forall$ op1 op2. op1 ** op2 = op2 ** op1 $\Rightarrow$ op1 ** op2 cop_pow n = op2 pow n * op1‘

Table 3.2: Theorems Examples for the Type cop

Definition 3.7.

is_linear_cop (op : cop) $\Leftrightarrow$

$\forall$x y. op (x + y) = op x + op y $\wedge \forall$a. op (a % x) = a % (op x)

Actually, linear operators are very powerful and have a great effect on many theorems. For instance, in the case of operators multiplication, we mentioned earlier that the left-distributivity does not hold. However, for linear operators it holds:

Theorem 3.2.

$\forall$op$_1$ op$_2$ op$_3$. is_linear_cop op$_3$ $\Rightarrow$

op$_3$ ** (op$_1$ + op$_2$) = op$_3$ ** op$_1$ + op$_3$ ** op$_2$

So does the associativity of scalar multiplication on the right of a multiplication:

Theorem 3.3.

$\forall$z op$_1$ op$_2$. is_linear_cop op$_1$ $\Rightarrow$

op$_1$ ** (z % op$_2$) = z % (op$_1$ ** op$_2$)

In practice, one often has to prove that a given operator is linear. For this purpose, many congruence results are very useful and indeed have to be proved. We gathered some examples of them in the following theorem:

Theorem 3.4.

$\forall$op$_1$ op$_2$. is_linear_cop op$_1$ $\wedge$ is_linear_cop op$_2$ $\Rightarrow$

is_linear_cop (op$_1$ + op$_2$) $\wedge$ is_linear_cop (op$_1$ * op$_2$) $\wedge$

$\texttt{is\_linear\_cop}\ (\texttt{op}_2 - \texttt{op}_1) \wedge \forall \texttt{a}.\ \texttt{is\_linear\_cop}\ (\texttt{a}\ \%\ \texttt{op}_1)$

Together, these theorems allow to prove the most frequently seen situations dealing with linearity.

Now, we have developed the infinite/finite complex-valued functions space `cfun` and linear transformation of over such a space, and proved numerous `cfun`/`cop` related theorems. In the next section, we will show how to build an inner product space out of a `cfun` space, and present some interesting linear operators that are defined based on the notion of inner-product, e.g., self-adjoint operator.

3.1.2 Inner Product Space

An inner product space is a linear space augmented with a function, called an inner product, that satisfies certain properties. The domain of such a function is the linear space and its codomain is $\mathbb{C}$. Intuitively, the definition of the inner product changes depending on the underlying space. Since our linear space is somehow abstract (due to the type `cfun` depends on a type variable `A`), we do not provide a concrete implementation of the inner product; instead, we provide a general axiomatic definition which is valid with every possible instantiation of `A`. We thus introduce a predicate asserting whether a given function indeed satisfies the axioms of an inner product. We first define a type for inner product spaces: the type `inner_space` is defined as $(\texttt{cfun} \longrightarrow \texttt{bool}) \times (\texttt{cfun} \longrightarrow \texttt{cfun} \longrightarrow \texttt{complex})$. Then, we define the inner product space as follows:

Definition 3.8.

$\texttt{is\_inner\_space}\ ((\texttt{s}, \texttt{inprod}) : \texttt{inner\_space}) \Leftrightarrow$

$\quad \texttt{is\_cfun\_subspace s} \wedge$

$\forall$x. x $\in$ s $\Rightarrow$

1 real (inprod x x) $\wedge$ 0 $\leq$ real_of_complex (inprod x x) $\wedge$

2 (inprod x x = 0 $\Leftrightarrow$ x = cfun_zero) $\wedge$

3 $\forall$y. y $\in$ s $\Rightarrow$

4 cnj (inprod y x) = inprod x y $\wedge$

5 ($\forall$a. inprod x (a%y) = a $*$ (inprod x y)) $\wedge$

6 $\forall$z. z $\in$ s $\Rightarrow$ inprod (x + y) z = inprod x z + inprod y z

where inprod is the product function, real is a predicate of complex numbers that do not have imaginary parts, and real_of_complex is a function casting such a complex number into a real one. The definition lists the necessary conditions of an inner product: positive-definiteness (Lines 1 and 2), conjugate symmetry (Line 4) and linearity (Lines 5 and 6). Based on this definition, many theorems are proved for inner spaces. Table 3.3 contains examples of those theorems.

Property	HOL Theorem
INPROD_LSMUL	inprod (a%x) y = cnj a $*$ inprod x y
INPROD_LNEG	$\forall$inprod ($-$ x) y = $-$ inprod x y
INPROD_SUB_RDIST	inprod (x $-$ y) z = inprod x z $-$ inprod y z
INPROD_ADD_LDIST	inprod z (x + y) = inprod z x + inprod z y

Table 3.3: Examples of Inner Product Theorems

The interesting thing about Definition 3.8 is that each time the type A is instantiated and associated with the corresponding inner product (e.g., Lebesgue integral in case A is substituted with real and double integral if it is substituted with real^2) all proved theorems are ported for the new instantiated space without the need to reprove any single theorem.

Later in the thesis, we will present the formalization of the *uncertainty principle* which is considered a pillar of quantum mechanics. This notion requires other concepts and

theorems that utilize the inner spaces, e.g., orthogonality and the Cauchy-Schwarz Inequality. Orthogonality is commonly used in quantum mechanics: the basis (i.e., the span set) of a quantum states space are orthogonal. Two vectors are called orthogonal if their respective inner product is equal to zero:

Definition 3.9.

$$\texttt{are\_orthogonal}\ (\texttt{s}, \texttt{inprod})\ \texttt{u}\ \texttt{v} \Leftrightarrow$$
$$\texttt{is\_inner\_space}\ (\texttt{s}, \texttt{inprod}) \Rightarrow \texttt{inprod}\ \texttt{u}\ \texttt{v} = 0$$

Based on this definition, we can prove a couple of interesting theorems that are very helpful in the development of the Cauchy-Schwarz Inequality. First is the Pythagorean theorem, which states that the inner product of the sum of two vectors is equal to the sum of the squared norm of each vector separately (the norm of a vector v is equal to $\sqrt{\texttt{inprod v v}}$):

Theorem 3.5 (Pythagorean)**.**

$$\forall\ \texttt{s inprod u v}.\ \texttt{is\_inner\_space}\ (\texttt{s}, \texttt{inprod}) \wedge\ \texttt{are\_orthogonal}\ (\texttt{s}, \texttt{inprod})\ \texttt{u}\ \texttt{v} \Rightarrow$$
$$\texttt{inprod}\ (\texttt{s}, \texttt{inprod})\ (\texttt{u} + \texttt{v})\ (\texttt{u} + \texttt{v}) = \texttt{inprod}\ \texttt{u}\ \texttt{u} + \texttt{inprod}\ \texttt{v}\ \texttt{v}$$

the second theorem is Decomposition, which states that for any two vectors we can create a new vector out of them that is orthogonal to one of them. The theorem below shows the steps to create such a vector (see Lines 1-3):

Theorem 3.6 (Decomposition)**.**

$$\forall\ \texttt{s inprod u v}.\ \texttt{is\_inner\_space}\ (\texttt{s}, \texttt{inprod}) \Rightarrow$$
1 $\texttt{let proj\_v} = \frac{\texttt{inprod v u}}{\texttt{inprod v v}}\ \texttt{in}$

2 $\texttt{let orthogonal\_component} = \texttt{u} - \texttt{proj\_v}\ \%\ \texttt{v}\ \texttt{in}$

3 $\texttt{u} = \texttt{proj\_v}\ \%\ \texttt{v} + \texttt{orthogonal\_component} \wedge$
$\texttt{are\_orthogonal inprod v orthogonal\_component}$

Finally, there is the theorem of the Cauchy-Schwarz Inequality, which is very popular

in many engineering domains, e.g., information theory [39]. The theorem states that the norm of the inner product of two vectors is less or equal to the multiplication of the norm of each vector:

Theorem 3.7 (Cauchy-Schwarz Inequality)**.**

$\forall$ x y s inprod. is_inner_space (s, inprod) $\Rightarrow$

  norm (inprod x y) pow 2 $\leq$

    real_of_complex (inprod x x) $*$ real_of_complex (inprod y y)

where norm here denotes the norm of a complex number. Recall that, $\forall$ x.inprod (x, x) is a real value.

3.1.3 Hermitian Operators

In the before mentioned development, we have built all the mathematical foundation required to formalize quantum states. In the following, we will implement Hermitian and self-adjoint notions, with the help of linear operators and inner space, which allow the development of quantum operators, the second pillar of quantum theory.

A very useful notion of linear operators is that of the Hermitian adjoint. This operation generalizes the one of conjugate transpose in the finite-dimension case, and we formalize it as follows:

Definition 3.10.

is_hermitian $\mathtt{op}_1$ $\mathtt{op}_2$ (s, inprod) $\Leftrightarrow$

  is_inprod (s, inprod) $\Rightarrow$

    is_linear_cop $\mathtt{op}_1$ $\wedge$ is_linear_cop $\mathtt{op}_2$ $\wedge$

    $\forall$ x y. inprod x ($\mathtt{op}_1$ y) = inprod ($\mathtt{op}_2$ x) y

The relation is_hermitian $\mathtt{op}_1$ $\mathtt{op}_2$ holds iff $\mathtt{op}_2$ is the Hermitian adjoint of $\mathtt{op}_1$. We

use a relation to express the Hermitian instead of a function because the existence of a Hermitian operator cannot be proved in a general way: it depends a lot on the underlying space. In particular, this highlights a big difference between the finite and the infinite dimension case: in the finite dimension, one can just take the conjugate transpose of the underlying matrix to obtain the Hermitian. But in the infinite dimension, this is not as simple: there is indeed a notion of a transpose operator, but it yields an operator in the *dual space* of the original vector space. If there is an isomorphism between this dual space and the original vector space, then one can obtain a satisfying definition of the Hermitian. However, in the infinite dimension, there is not always such an isomorphism. In any case, if there is a Hermitian operator, then it is unique, as proved by the following theorem:

Theorem 3.8.

$\forall \mathtt{op_1}\ \mathtt{op_2}\ \mathtt{op_3}\ \mathtt{s}\ \mathtt{inprod}.$

$(\mathtt{is\_hermitian}\ \mathtt{op_1}\ \mathtt{op_2}\ (\mathtt{s}, \mathtt{inprod}) \wedge \mathtt{is\_hermitian}\ \mathtt{op_1}\ \mathtt{op_3}\ (\mathtt{s}, \mathtt{inprod}))$

$\Rightarrow (\forall\ \mathtt{x}.\ \mathtt{x} \in \mathtt{s} \Rightarrow \mathtt{op_2}\ \mathtt{x} = \mathtt{op_3}\ \mathtt{x})$

Note that the operators $\mathtt{op_2}$ and $\mathtt{op_3}$ are equal up to the inner space `(s,inpord)`, since we do not know how they behave outside the space. We also proved some other properties of the Hermitian, such as the symmetry of its relation:

Theorem 3.9.

$\forall \mathtt{s}\ \mathtt{inprod}\ \mathtt{op_1}\ \mathtt{op_2}.$

$\mathtt{is\_hermitian}\ \mathtt{op_1}\ \mathtt{op_2}\ (\mathtt{s}, \mathtt{inprod}) \Leftrightarrow \mathtt{is\_hermitian}\ \mathtt{op_2}\ \mathtt{op_1}\ (\mathtt{s}, \mathtt{inprod})$

Finally, we prove some congruence theorems which allow to prove, in many cases, that a given operator is the Hermitian of another:

Theorem 3.10.

$\forall$ s inprod op_1 op_2 op_3 op_4 a.

is_hermitian op_1 op_2 (s, inprod) $\wedge$ is_hermitian op_3 op_4 (s, inprod) $\Rightarrow$

is_hermitian $(op_1 + op_3)$ $(op_2 + op_4)$ (s, inprod) $\wedge$

is_hermitian $(op_1 - op_3)$ $(op_2 - op_4)$ (s, inprod) $\wedge$

is_hermitian $(op_1 * op_3)$ $(op_4 * op_2)$ (s, inprod) $\wedge$

is_hermitian (a % op_1) (cnj a % op_2) (s, inprod)

We also provide a more "computational" version of these congruence theorems:

Theorem 3.11.

$\forall$ a b s inprod op_1 op_2 op_3 op_4 op_5.

is_hermitian op_1 op_2 (s, inprod) $\wedge$ is_hermitian op_3 op_4 (s, inprod) $\wedge$

is_hermitian (a % op_1 + b % op_3) op_5 (s, inprod) $\Rightarrow$

op_5 = cnj a % op_2 + cnj b % op_4

Self-Adjoint Operators

A highly coupled notion to Hermitian relation is self-adjointness, which typically represents quantum operators. A self-adjoint operator denotes operators which are their own Hermitian adjoint:

Definition 3.11.

is_self_adjoint op (s, inprod) $\Leftrightarrow$ is_hermitian op op (s, inprod)

Once again, we have proved many congruence theorems allowing to deal with most self-adjoint operators that are encountered in proofs. Most of them are similar to the ones for the Hermitian, only the case of scalar multiplication should be handled with care, since we require that the scalar is a real number:

Theorem 3.12.

$\forall$ s inprod op a. is_inprod (s, inprod) $\wedge$ real a

$\wedge$ is_self_adjoint op (s, inprod) $\Rightarrow$ is_self_adjoint(a % op) (s, inprod)

Some other results are less obvious and very useful, for instance:

Theorem 3.13.

$\forall$ s inprod op x y.

is_inprod (s, inprod $\wedge$ is_linear_op op $\wedge$

inprod (op x) y = $-$(inprod x (op y)))

$\Rightarrow$ is_self_adjoint (ii % op) (s, inprod

Eigenvalues and Eigenvectors

The eigenvalues and eigenvectors of observables (or self-adjoint operators) are of high interest in quantum mechanics. For instance, fock states are eigenvectors of the photon number operators, and coherent states are eigenvectors of annihilator operators, for an optical beam (see Section 2.3). Eigenvectors (or alternatively eigenstates) of such operators form the basis of quantum states spaces, i.e., any quantum optical state can be represented in terms of them. Another important aspect of eigenstates is their deterministic measurement nature, which is in contrast to regular quantum states which are known to be probabilistic. Therefore, we have to consider the formalization of eigenvectors and eigenvalues. We define a pair of eigenvalue and eigenvector (v, μ) of a linear operator op in the context of complex-valued functions as follows:

Definition 3.12.

is_eigen_pair op (v, μ) $\Leftrightarrow$

is_linear_cop op $\Rightarrow$ op v = μ% v $\wedge$ (v $\neq$ cfun_zero)

Note that an eigenvector should not be the zero vector (i.e., `cfun_zero`).

For an eigenvalue/eigenvector pair, we have proved a couple of interesting theorems. For instance, the sum of two eigenvectors, with the same eigenvalue, is an eigenvector with the same eigenvalue:

Theorem 3.14.

$\forall$op μ u v.

$$\texttt{is\_eigen\_pair op}\ (\texttt{v}, \mu) \wedge \texttt{is\_eigen\_pair op}\ (\texttt{u}, \mu)$$

$$\wedge\ (\texttt{u} + \texttt{v} = \texttt{cfun\_zero}) \Rightarrow \texttt{is\_eigen\_pair op}\ (\texttt{u} + \texttt{v}, \mu)$$

Another important result property is the orthogonality of eigenstates of a quantum operator (i.e., self-adjoint) `op`, with different eigenvalues:

Theorem 3.15.

$\forall$qs inprod op.

$$\texttt{is\_qspace}\ (\texttt{qs}, \texttt{inprod}) \wedge \texttt{is\_observable}\ (\texttt{qs}, \texttt{inprod})\ \texttt{op}\ \Rightarrow$$

$$\forall \texttt{u v}\ \nu\ \mu\ \texttt{u}\ \in\ \texttt{s} \wedge \texttt{v}\ \in\ \texttt{s} \wedge (\texttt{nu} \neq \texttt{mu}) \quad \wedge \texttt{is\_eigen\_pairop}(\texttt{u}, \texttt{nu})$$

$$\wedge\ \texttt{is\_eigen\_pair op}\ (\texttt{v}, \texttt{mu}) \Rightarrow \texttt{are\_orthogonal}\ (\texttt{qs}, \texttt{inprod})\ \texttt{u v}$$

Now, we have developed all the mathematical ingredients needed to formalize quantum preliminaries notions. However, to tackle advanced quantum concepts (e.g., the representation of mixed-state as infinite summation of eignestates), devices and circuits (e.g., beam splitters and quantum gates), we need to go further by implementing functional analysis concepts, which by nature are more difficult. In the following, we will tackle the formalization of limit and finite/infinite summation over `cfun`.

3.2 Formalization of Infinite Summation

In this section, we formalize the notion of infinite/finite summation over `cfun`. Being inspired by Harrison's formalization of summation over finite Euclidian vector spaces [30], we develop ours for infinite dimensional complex-valued functions spaces `cfun`. The summation formalization goes through three major steps: 1) define the finite summation, 2) define the limit notion, then 3) extend the finite one to the infinite summation by applying the notion of limit.

3.2.1 Finite Summation

HOL Light supports the `iterate` function that accepts an operation and finite set of elements, then repeatedly applies the operation on the elements belonging to the set. Hence, `iterate` is the best way to define the finite summation:

Definition 3.13.

`cfun_sum` = `iterate cfun_add`

Recall that `cfun_add` is the addition operation between two `cfun` functions. Now, `cfun_sum` is a new operation that accepts two parameters: a finite indexing set `s` (typically, but not limited to, a subset of natural numbers $\mathbb{N}$) and a function `f` : `s` $\longrightarrow$ `cfun`.

In order to prove useful properties about `cfun_sum`, we first need to provide the following essential theorem, *sum clauses*:

Theorem 3.16.

(∀`f`. `cfun_sum` {} `f` = `cfun_zero`)∧

  (∀`f n m`. `FINITE s` ⇒

$$\texttt{cfun\_sum}\ (\texttt{n..m})\ \texttt{f} = \texttt{f(n)} + \texttt{cfun\_sum(n} + \texttt{1..m)}\ \texttt{f)}$$

The theorem classifies the `cfun_sum` into two cases: either the indexing set is empty then the summation is trivial and it is equal `cfun_zero`. Or, given a set of natural numbers $\{x : x \geq n \wedge x \leq m\}$ then the summation can be divided into two terms, see the third line of Theorem 3.16. We can then prove many interesting results based on this theorem, such as *sum of constant*:

Theorem 3.17.

$\forall$`c s. FINITE s` $\Rightarrow$ `cfun_sum s` $(\lambda$`n. c`$)$ = `(CARD s)`%`c`

where `CARD s`, i.e., cardinality of `s`, returns the number of elements in `s`. Theorem 3.17 simply shows that a finite summation turns into a scalar multiplication whenever `f` is a constant function. The next theorem is about closure under `cfun_sum`:

Theorem 3.18.

$\forall$`g spc. is_cfun_subspace spc` $\wedge$ $(\forall$`n. g n IN spc`$) \Rightarrow$

$\forall$`s. FINITE s` $\Rightarrow$ `cfun_sum s g IN spc`

Given a set of complex-valued functions `cfun` (or alternatively vectors) which is a subset of a subspace `spc`, the resulting sum over those vectors indeed belongs to the same subspace `spc`, and hence it is also a vector. We conclude about finite summation with such an important theorem which describes the relationship between linear operators and finite summation:

Theorem 3.19.

$\forall$`f g s.is_linear_cop f` $\wedge$ `FINITE s` $\Rightarrow$ `(f(cfun_sum s g) = cfun_sum s (f o g))`

The theorem clearly shows that linear operators are interchangeable with the finite summation. A known application of this theorem is exchanging the integration function with the summation.

3.2.2 Infinite Summation

The infinite summation can be extended from the finite one using the notion of limit. The latter is tightly coupled with the existence of a normed-space, i.e., a linear space augmented with a norm function. In the context of quantum state spaces (or inner product of `cfun`), a normed-space can be obtained by evaluating the square root of the inner product function of a vector and itself, which in turn yields the norm operation. Formally, we can write the norm of a `cfun` as `cfun_norm inprod x` $= \sqrt{\texttt{inprod x x}}$. Accordingly, the notion of limit can be implemented for quantum spaces as follows:

Theorem 3.20.

`cfun_lim` $(\texttt{s}, \texttt{inprod})$ `f l net` $\Leftrightarrow$

`is_inner_space` $(\texttt{s}, \texttt{inprod}) \wedge$ `l IN s`$/$ $(\forall$`x`. $($`f x`$)$ `IN s`$)\wedge$

$(\forall$`e`. `0` $\leq$ `e` $\Rightarrow$ `eventually`$(\lambda$ `x`. `cfun_dist inprod` $($`f x`$)$ `l` $<$ `e`$)$ `net`$)$

where `cfun_dist inprod x y` = `cfun_norm inprod` $(\texttt{x} - \texttt{y})$. The definition starts with the guarding antecedents which assure that we have an inner space and all the elements we are dealing with are inside this space. Then, it ensures that the difference (or `cfun_dist`) between a vector `f x` and the limit vector `l` is getting smaller, while `x` changes according to the `net`. An example of `net`s is a sequential net for which the parameter `x` starts from 0 and increases gradually until infinity. This definition alone is not enough to reason about the important properties of limit, e.g., linearity. It requires in addition the key theorem of *uniqueness*:

Theorem 3.21.

$\forall$ `net f l l'' innerspc`.

`cfun_lim innerspc f l net` $\wedge$`cfun_lim innerspc f l'' net` $\Rightarrow$ $($`l` $=$ `l''`$)$

By uniqueness, we mean that if it happens that a function `f : A ⟶ cfun` limits to a vector `l:cfun`, and at the same time to vector a `l':cfun`, then `l` should be equal to `l'`. All the proved properties and theorems for the notion of limit have counterparts in the infinite summation. Since quantum theory is in direct contact with infinite summation rather than the notion of limit, we will present these theorems in the context of infinite summation to avoid repetition.

Given the limit definition, we can then extend the finite summation to define the infinite summation of `cfun` as follows:

Definition 3.14.

```
cfun_sums innerspc f l s ⇔
  cfun_lim innerspc (λn. cfun_sum (s INTER (0..n)) f) l sequentially
```

where `INTER` is the sets intersection operator. In order to easily understand the definition, let us assume `s` is equal to the set of natural numbers. Consequently, `(s INTER (0..n))= 0..n`. Then, the definition states that while n increases, the finite summation `cfun_sum` coincides with (or is limited to) `l`. However, this predicate definition does not help much in usual mathematical manipulation. Therefore, we develop another functional definition:

Definition 3.15.

```
cfun_infsum innerspc s f = @l. cfun_sums innerspc f l s
```

Here, the definition uses the Hilbert choice operator `@` to get randomly a vector that satisfies the `cfun_sums` predicate. Since the `cfun_sums` relation satisfies the uniqueness (similar to the notion of limit), then the choice operator always returns the same vector.

In order to proceed with proving theorems related to infinite summation, we have to

first make sure that the series of vectors subject to summation is convergent, i.e., the limit exists. For this purpose, we define the `summable` predicate:

Definition 3.16.

`cfun_summable innerspc s f` $=$ $\exists$`l. cfun_sums innerspc f l s`

Given the before mentioned definitions about infinite summation, we can prove a number of important theorems, e.g., the linearity of `cfun_infsum`, which is expressed in the following two theorems:

Theorem 3.22.

$\forall$ `f g innerspc.`

`cfun_summable innerspc s f` $\wedge$ `cfun_summable innerspc s g` $\Rightarrow$

`cfun_infsum innerspc s`(λ`n.fn + gn`) $=$

`cfun_infsum innerspc s f + cfun_infsum innerspc s g`

The above theorem can be read as the infinite summation of the sum of two functions is equivalent to the sum of the infinite summation of each separately.

Theorem 3.23.

$\forall$ `f innerspc a. cfun_summable innerspc s f` $\Rightarrow$

`cfun_infsum innerspc s`(λ`n.a` % `f n`) $=$ `a`% `cfun_infsum innerspc s f`

This theorem allows to strip out a constant of a scalar multiplication from inside the infinite summation. This theorem is very useful since it eases proving similar results, e.g., the infinite summation of function negation and functions subtraction:

$$\texttt{cfun\_infsum innerspc s}(\lambda\texttt{n. } - \texttt{ f n}) = - \texttt{ cfun\_infsum s f}$$

$$\texttt{cfun\_infsum s}(\lambda\texttt{n.fn} - \texttt{gn}) = \texttt{cfun\_infsum s f} - \texttt{cfun\_infsum s g}$$

Similar to the finite case, we have proved that `cfun_infsum` is interchangeable with linear operators. However, there is an extra condition that a linear operator should

satisfy for this property to hold: it should be bounded. Before we present the theorem itself, let us express the formal definition of boundness:

Definition 3.17.

$$\texttt{is\_bounded}\ (\texttt{s}, \texttt{inprod})\ \texttt{h} \Leftrightarrow\ \texttt{is\_inner\_space}\ (\texttt{s}, \texttt{inprod})$$
$$\Rightarrow\ \texttt{is\_closed\_by s h} \wedge \exists \texttt{B}.\ 0 < \texttt{B} \wedge$$
$$(\forall \texttt{x}.\ \texttt{x IN s} \Rightarrow \texttt{cfun\_norm inprod}\ (\texttt{h x}))) \leq \texttt{B} * \texttt{cfun\_norm inprod x})))$$

Here, a linear operator `h` is bounded if for all `x` the norm of `h x` is less than or equal to the norm of `x` multiplied by a scalar `B`, given that B does not depend on `x`. Accordingly, a bounded linear operator is interchangeable with the `cfun_infsum` as follows:

Theorem 3.24.

$$\forall \texttt{f h s innerspc}.$$
$$\texttt{cfun\_summableinnerspcsf} \wedge \texttt{is\_linear\_cop h} \wedge \texttt{is\_bounded innerspc h}$$
$$\Rightarrow \texttt{cfun\_infsum innerspc s}(\lambda \texttt{n}.\ \texttt{h}(\texttt{f n})) = \texttt{h}(\texttt{cfun\_infsum innerspc s f})$$

This concludes the formalization finite/infinite summation over complex-valued functions. In the following section, we will discuss the implementation of a number of tactics (theorem prover utility function that automate the formal proofs or part of them) that helped in shortening the length of proofs of many theorems that were presented in this chapter.

3.3 Developed Tactics

In this work, we successfully developed several tactics that automatize parts of our proofs, which reduced the length of the proof scripts in many instances (e.g., reducing

part of the code from 300 lines to around 50 lines) and make the proofs easier. Examples of such tactics are `CFUN_ARITH_TAC` and `COP_ARITH_TAC`, which are responsible for handling simple equational theorems. These tactics, which were inspired by the HOL Light tactics `REAL_ARITH_TAC` and `COMPLEX_SIMPLE_ARITH_TAC`, helped in proving intermediate steps rewriting and variables reordering. They mainly convert all variables and expressions, in the goal to be proved, of types `cfun` and `cop` to the complex data type, with the help of all formal definitions we have for those types. For this purpose, we defined the following:

Definition 3.18.

`let CFUN_TO_COMPLEX = CONJS [FUN_MAP_THMS; cfun_defs; CFUN_EQ]`

`let COP_TO_CFUN = CONJS [FUN_MAP_THMS; o_THM; cop_defs; COP_EQ]`

where `cfun_defs` and `cop_defs` contain definitions of arithmetic operations. We then put the goal in a uniform format using the prenex conversion and other lemmas, which bring all quantifiers to the most left side of the goal. Finally, we make a call to `COMPLEX_SIMPLE_ARITH_TAC` or `COMPLEX_FIELD`, which handle the proof at the level of the type complex. We have proved no less than 100 theorems with these two tactics only.

Another tactic is `LINEARITY_TAC`, which is mainly responsible for proving the linearity of a certain `cop` operator based on the `is_linear_cop` definition. The tactic is based on three foundations: 1) The list `linearity_thms` that contains all theorems related to linearity. This list is updatable, i.e., each time a new linear operator or linearity theorem is proved, it can be added to this list using the function `add_linearity_thms`; 2) The Linearity Loop, which goes over `linearity_thms` iteratively, and for each theorem in the list, tries to prove the goal using this theorem with the help of rewriting/simplification tactics or modus pones matching tactics. As long as there is a

change in the goal after each loop, the Linearity Loop continues working until the goal is proved; 3) The tactic fails when no change can be brought to the goal, or if the goal is not about linearity: We parse the goal to make sure that it contains the word `is_linear_cop` with the correct data type to ensure that we are dealing with a correct goal, otherwise it fails. Here is an example of a theorem about theperverseness of linearity under commutator which can be proved using `LINEARITY_TAC`:

Theorem 3.25. $\forall$ `op1 op2.` `is_linear_cop op1` $\wedge$ `is_linear_cop op2`

$\Rightarrow$ `is_linear_cop (commutator op1 op2)`

Similarly, we implemented two other tactics: `SELF_ADJOINT_TAC`, which proves the self-adjointness of a `cop` operator, and `REAL_TAC`, which proves, for a given variable of complex type, that it is a real number, i.e., its imaginary part is equal to zero. These two tactics follow the same technique as `LINEARITY_TAC`, where we use `selfadjoint_thms` and `add_selfadjoint_thms`, and `real_thms` and `add_real_thms`. We also make sure that the goal contains the correct predicate, i.e., `is_self_adjoint` in case of `SELF_ADJOINT_TAC` and `real` in case of `REAL_TAC`. The major difference between `SELF_ADJOINT_TAC` and the other two tactics is that it internally calls `LINEARITY_TAC` and `REAL_TAC` since the definition of self-adjointness is based on linearity. Here is another example of a theorem about the perverseness of self-adjointness under the negation that can be proved using these tactics:

Theorem 3.26. $\forall$ `op.` `is_linear_cop op1` $\wedge$ `is_self_adjoint is op`

$\Rightarrow$ `is_self_adjoint is` $(- - \texttt{op})$

3.4 Summary

In this chapter, we have presented the formal development of the mathematical foundation for quantum theory. It is basically dealing with complex-valued functions and their corresponding linear spaces. In particular, we have implemented the linear transformation over such linear spaces, and extended such spaces to inner product ones, where quantum states reside. In addition, we have developed some interesting operators, e.g., self-adjoint and Hermitian operators. Moreover, we have tackled a number of functional analysis concepts, namely limit and infinite summation over complex-valued functions. In total, we have formally proved 450 theorems, and defined 50 formal definitions in this development. We also developed five tactics: `CFUN_ARITH_TAC` and `COP_ARITH_TAC`, which are responsible for proving simple arithmetic equational theorems of variables of types `cfun` and `cop`; `LINEARITY_TAC` proves the linearity of the `cop` operator according to `is_linear_cop`, and similarly `SELF_ADJOINT_TAC` which proves the self-adjointness of the `cop` operator; `REAL_TAC` proves, for a given variable of complex type, that it is a real number, i.e., its imaginary part is equal to zero. These tactics helped in reducing the length of proofs of many theorems that were presented in this chapter. Remarkably, this library became part of HOL Light's latest release [27], which shows the usefulness of this library in other domains than quantum mechanics.

Chapter 4

Quantum Optics Formalization

In the previous chapter, we have presented the formalization of functional space. In this chapter, we develop the formalization of quantum mechanics notions, that includes quantum states and quantum systems. In addition, we formally prove a number of interesting theorems, e.g., the uncertainty principle, a pillar of quantum physics. Based on the generic definition of quantum systems, we implement the single-mode fields which mimic simple optical beams. Accordingly, a number of important optics notions are formally developed, e.g., fock states, coherent states, and multi-mode fields.

4.1 Formalization of Quantum Mechanics

In this section, we develop the higher-order logic formalization of the quantum notions presented in Section 2.2, where we utilize the formal developments presented in the previous chapter. We start by defining a type for quantum states as `qstate`, which is typically a type abbreviation of the type `cfun` defined in Section 3.1. It is important

to note that this type contains an abstract type (recall $\texttt{cfun} : \texttt{A} \longrightarrow \mathbb{C}$), which can be instantiated differently depending on the system subject to analysis (e.g., for quantum optical beams, `A` is instantiated as `real`, as we will see later in this chapter). Then, we define the type of the quantum states space as `qspace`, which is again an abbreviation of the type `inner_space`. In the same context, we define `is_qspace` as `is_inner_space`.

Since not all complex-valued functions in the `qspace` are quantum states (i.e., normalized), we then characterize the quantum ones as follows:

Definition 4.1.

$$\texttt{is\_qst}\ (\texttt{qs}, \texttt{inprod})\ \texttt{qst} \Leftrightarrow \texttt{qst} \in \texttt{qs} \wedge \texttt{inprod}\ \texttt{qst}\ \texttt{qst} = \texttt{Cx}(1)$$

where `Cx` is a type casting function that converts numbers of real type to numbers of type complex. In our case, Cx(1) corresponds to a complex number with an imaginary part that is equal to zero and a real part that is equal to one.

For quantum operators, i.e., observables, we define the type $\texttt{qop} : \texttt{qstate} \longrightarrow \texttt{qstate}$, which is a special case of the general $\texttt{cop} : \texttt{cfunB} \longrightarrow \texttt{cfunC}$, with the same domain and codomain. Typically, an observable is a linear self-adjoint operator, thus it can be formally defined as follows:

Definition 4.2.

$$\texttt{is\_observable}\ (\texttt{op} : \texttt{qstate} \longrightarrow \texttt{qstate})\ (\texttt{qs}, \texttt{inprod}) \Leftrightarrow$$
$$\texttt{is\_qspace}\ (\texttt{qs}, \texttt{inprod}) \Rightarrow \texttt{is\_self\_adjoint}\ (\texttt{qs}, \texttt{inprod})\ \texttt{op}$$

where `qs` stands for the sets of quantum states.

The remaining ingredient of a quantum system is the definition of canonical coordinates. For this purpose, we define the type $\texttt{coords} = \texttt{qop}\ \texttt{list}$, and define the valid coordinates according to the following predicate:

Definition 4.3.

```
are_canonical_coords (can_cords : coords) ⇔
let n = (LENGTH can_cords) DIV 2 in
1  (LENGTH can_cords) MOD 2 = 0
2  ∧ ∀.i ji < 2 * n ∧ j < 2 * n ⇒
3  commutator (EL i can_cords) (EL j can_cords)
4  = if j − i = n then (ii * Cx planck) % I else cop_zero
```

where `LENGTH` is a function that returns the length of a given list, `(EL i l)` returns the i^{th} element of `l`. The above definition ensures firstly that the length is even (see Line 1), since each operator (or coordinate) should have its own canonical. Note that the canonical list `can_cords` starts with the coordinates themselves, then their respective canonicals, i.e., the canonical of the i^{th} coordinate is at $i + n$. Thus, the commutator of two coordinates in the list is equal $i\hbar$ if they are canonical, i.e., the difference in position is equal to n; otherwise, it is equal to zero (see Lines 3 and 4).

Now, we have all the materials needed to define a *quantum system*: a states space, a list of canonical coordinates, and a Hamiltonian function that describes the evolution of the system state (typically the energy function). Accordingly, a system has the type `qsys : qspace × coords × qop`. Similar to quantum states and operators, we define a predicate that describes valid quantum systems:

Definition 4.4.

```
is_qsys (qs, cs) ⇔ is_qspace qs∧  are_canonical_coords cs
```

where `qs` stands for quantum states space and `cs` for coordinates.

Using these definitions, we can prove interesting results about quantum systems. In

particular, we prove that the measurement of eigenstates is deterministic, in contrast to the probabilistic nature of quantum states, as well as the *uncertainty principle* [23]. In the following, we explain the physical meaning of these results and present their formalization.

4.1.1 Eigenstates

Despite the probabilistic nature of the measurement process, it is still deterministic for some special quantum states (which are called *eigenstates*). From their name, we can gather that such states are related to the existence of a linear operator, in particular a quantum operator. Mathematically, a deterministic state means that the variance of measurement vanishes for such a quantum state. The following theorems present it formally:

Theorem 4.1.

$\forall$ `qs inprod op`.

$$\texttt{is\_qspace}\ (\texttt{qs}, \texttt{inprod}) \wedge \texttt{is\_observable}\ (\texttt{qs}, \texttt{inprod})\ \texttt{op} \Rightarrow$$

$$\forall \mu\ \texttt{qst}.\ \texttt{is\_qst}\ (\texttt{qs}, \texttt{inprod})\ \texttt{qst} \wedge \texttt{is\_eigen\_pair op qst}\ \mu \Rightarrow$$

$$\texttt{variance inprod qst op} = \texttt{Cx}(0)$$

where `op` is a quantum operator and `qst` is an eigenstate of the observable `op`, and μ is the corresponding eigenvalue. According to Section 2.2, the `variance` is defined in terms of expectation as follows:

Definition 4.5.

$$\texttt{variance inprod v op} = \texttt{expectation inprod v}\ (\texttt{op} - \texttt{expectation inprod v op})^2$$

$$\texttt{expectation inprod v op} = \texttt{inprod v}\ (\texttt{op v})$$

Since the variance vanishes as proved in Theorem 4.1, the measurement is always

equal to the expectation. In the following theorem, we prove that the measurement expectation of a given observable `op` at an eigenstate `qst` is equal to the corresponding eigenvalue:

Theorem 4.2.

$\forall$`qs inprod op`.

$$\texttt{is\_qspace}\ (\texttt{qs}, \texttt{inprod}) \wedge \texttt{is\_linear\_cop}\ \texttt{op} \Rightarrow$$

$$\forall \mu\ \texttt{v}.\ \texttt{is\_qst}\ (\texttt{qs}, \texttt{inprod})\ \texttt{v} \wedge \texttt{is\_eigen\_pair}\ \texttt{op}\ \texttt{v}\ \mu \Rightarrow$$

$$\texttt{expectation in\_prod v op} = \mu$$

Another important property about the eigenstates of a quantum operator `op` is that they can form a basis (or span set) of the quantum states space to which they belong. This property can be equivalently formalized by proving that any two eigenstates of an observable `op` with different eigenvalues are orthogonal:

Theorem 4.3.

$\forall$`qs inprod op`.

$$\texttt{is\_qspace}\ (\texttt{qs}, \texttt{inprod}) \wedge \texttt{is\_observable}\ (\texttt{qs}, \texttt{inprod})\ \texttt{op}\ \Rightarrow$$

$$\forall \texttt{u}\ \texttt{v}\ \nu\ \mu\ \texttt{u}\ \in\ \texttt{s} \wedge \texttt{v}\ \in\ \texttt{s} \wedge (\texttt{nu} \neq \texttt{mu}) \quad \wedge \texttt{is\_eigen\_pairop}(\texttt{u}, \texttt{nu})$$

$$\wedge\ \texttt{is\_eigen\_pair op}\ (\texttt{v}, \texttt{mu}) \Rightarrow \texttt{are\_orthogonal}\ (\texttt{qs}, \texttt{inprod})\ \texttt{u}\ \texttt{v}$$

Recall that a set of orthogonal vectors are linearly independent [34], which is typically a span set.

4.1.2 Uncertainty Principle

The uncertainty principle is considered one of the most important quantum notions, and is used in the definition of *coherent states* and *squeezed states*, which are commonly used in the development of quantum computers [66] (later in this chapter, we will see

the coherent state's development). The principle declares that we cannot measure two observables simultaneously with high accuracy. A very popular example of this principal is that we cannot measure exactly the position and momentum of an electron at the same time. In other words, the measurement accuracy of one observable is at the expense of the other's observable accuracy. This can be formally written as follows:

Theorem 4.4.

$\forall \texttt{obs}_1\ \texttt{obs}_2\ \texttt{spc}\ \texttt{inprod}\ \texttt{t}\ \texttt{qst}.$

1 $\texttt{is\_observable}\ \texttt{obs}_1\ (\texttt{spc}, \texttt{inprod}) \wedge \texttt{is\_observable}\ \texttt{obs}_2\ (\texttt{spc}, \texttt{inprod}) \wedge$

2 $\texttt{is\_qspace}\ (\texttt{spc}, \texttt{inprod}) \wedge \texttt{is\_qst}\ (\texttt{spc}, \texttt{inprod})\ \texttt{qst} \Rightarrow$

3 $\left(\frac{\texttt{expectation inprod qst (commutator obs}_1\ \texttt{obs}_2\texttt{)}}{\texttt{Cx(2)*i}}\right)\ \texttt{pow 2}$

4 $\leq \texttt{real\_of\_complex}\ (\texttt{variance inprod qst obs}_1\texttt{)}$

5 $* \texttt{real\_of\_complex}\ (\texttt{variance inprod qst obs}_2\texttt{)}$

Here, `i` is the imaginary number and $\texttt{obs}_\texttt{x}$ abbreviates $\texttt{observable}_\texttt{x}$. Recall that $\texttt{commutator obs}_1\ \texttt{obs}_2 = \texttt{obs}_1\ \texttt{obs}_2\ - \texttt{obs}_4\ \texttt{obs}_1$. Lines 1 and 2 are antecedents, which ensure that we are working on the appropriate parameters. The principle itself is expressed in Lines 3-5, which show that the variances of two non-commuting observables (i.e., commutator is greater than zero) are inversely proportional: the variances multiplication (Lines 4 and 5) is upper bounded by the amount in Line 3, thus any attempt to enhance the accuracy of an operator measurement (i.e., decrease of the corresponding variance) implies an increase in the variance of the other operator (i.e., lowering the accuracy of its measurement). Recall that the variance is an indication of the measurement accuracy. The proof of this theorem is highly dependent on the Schwartz inequality which was presented in Section 3.1.2.

This concludes the formalization of quantum mechanics. In the next section, we

will present the quantum optics formalization, in which we tackle the quantization of single-mode fields. Then, we generalize it to prove results for the Multi-Mode case by considering the notion of tensor product.

4.2 Formalization of Quantum Optics

In the previous section, we defined quantum rules that apply for all quantum systems, where we considered a general system with an abstract type `qstate` : `A` $\longrightarrow$ `complex` to express its quantum state. In this section, we cover a particular system, namely optical beams. Accordingly, we instantiate `A` to be of type `real`, since the coordinate of an optical beam is the amount of charges q which is typically of type `real` (see Section 2.3). Thus, the optical quantum state is of type `bqs` : `real` $\longrightarrow \mathbb{C}$, and the corresponding inner-product function is the Lebesgue integration:

$$\texttt{real\_integration} : \texttt{bqs} \longrightarrow \texttt{bqs} \longrightarrow \mathbb{C}$$

Before we move forward to the formalization of quantum optics notions, i.e., single-mode, we have first to prove that this new set of square Lebesgue integrable functions forms a quantum states space (i.e., inner-product space).

We start by formally defining the notion of the set of square integrable complex-valued functions, namely `sq_integrable`:

Definition 4.6.

`new_specification` [“`sq_integrable`”]

$\forall$`f`. `f IN sq_integrable` $\Leftrightarrow$

1 `f complex_measurable_on` (: `real`) $\wedge$

2 (λ`x`. ||`f x`|| ^{2}) `real_integrable_on` (: `real`)

where `new_specification` is a HOL Light function that allows the definition of a constant that satisfies a certain condition (or predicate). Note that the square of a complex-valued function `f` is equal to the multiplication of `f(x)` by its conjugate $\texttt{f(x)}^*$. This is equivalent to the norm square of the complex value `f(x)`, as presented in Line 2. In order to make the set `sq_integrable` form an inner-product subspace, the functions $\texttt{f} \in \texttt{sq\_integrable}$ must satisfy another condition, which is the complex measurability [44]:

Definition 4.7.

```
f complex_measurable_on s ⇔
  (λx. Re (f x)) real_measurable_on s ∧
    (λx.Im (f x)) real_measurable_on s
```

Note here that the measurability and integrability are over the whole real line (i.e., from $-\infty$ to ∞). Accordingly, we define the inner product function over the elements of space `sq_integrable` as follows:

Definition 4.8.

```
r_inprod f g =
1  complex(real_integral (: real) (λx : real. Re((f x)* * (g x))),
2  real_integral (: real) (λx.Im ((f x)* * (g x))))
```

The above definition states that the inner product of two square integrable functions `f` and `g` is a complex value, whose real part is the Lebesgue integral of the real part of $\texttt{f} * \texttt{g}$ (see Line 1), and its imaginary part is the Lebesgue integral of the imaginary part of $\texttt{f} * \texttt{g}$ (see Line 2).

Now, we move to the most crucial part, namely to prove that these definitions form a linear space and the associated `r_inprod` function is its inner product. Formally, we need to prove the following theorem, which is an instantiation of Definition 3.8 (see

Section 3.1.2):

Theorem 4.5.

$$\texttt{is\_cfun\_subspace sq\_integrable} \wedge \forall \texttt{x}.\ \texttt{x} \in \texttt{sq\_integrable} \Rightarrow$$

$$\texttt{real}\ (\texttt{r\_inprod x x}) \wedge 0 \leq \texttt{real\_of\_complex}\ (\texttt{r\_inprod x x}) \wedge$$

$$(\texttt{r\_inprod x x} = \texttt{Cx}(0) \Leftrightarrow \texttt{x} = \texttt{cfun\_zero}) \wedge$$

$$\forall \texttt{y}.\ \texttt{y} \in \texttt{sq\_integrable} \Rightarrow \texttt{cnj}\ (\texttt{r}_\texttt{i}\texttt{nprod y x}) = \texttt{r\_inprod x y} \wedge$$

$$(\forall \texttt{a}.\ \texttt{r\_inprod x}\ (\texttt{a}\%\texttt{y}) = \texttt{a} * (\texttt{r\_inprod x y})) \wedge$$

$$\forall \texttt{z}.\ \texttt{z} \in \texttt{sq\_integrable}\ \Rightarrow \texttt{r\_inprod}\ (\texttt{x} + \texttt{y})\ \texttt{z} = \texttt{r\_inprod x z} + \texttt{r\_inprod y z}$$

The proof of these properties is quite long and complex; however, we believe that there are two major lemmas that control most of the proof steps. The first lemma is about deriving the integrability of functions multiplication given some assumptions. This lemma is used in proving many intermediate steps of all the above inner products properties. The lemma follows:

Theorem 4.6.

$$\texttt{Im f real\_measurable\_on}\ (:\texttt{real}) \wedge \texttt{Re f real\_measurable\_on}\ (:\texttt{real})$$

$$\wedge\ \texttt{Im g real\_measurable\_on}\ (:\texttt{real}) \wedge \texttt{Re g real\_measurable\_on}\ (:\texttt{real})$$

$$\wedge\ (\texttt{Im f})^2 + (\texttt{Re f})^2\texttt{real\_integrable\_on}\ (:\texttt{real})$$

$$\wedge\ (\texttt{Im g})^2 + (\texttt{Re g})^2\texttt{real\_integrable\_on}\ (:\texttt{real})$$

$$\Rightarrow 2 * (\texttt{Im f}) * (\texttt{Re f})\texttt{real\_integrable\_on}\ (:\texttt{real})$$

The lemma states that for two square integrable measurable functions, all possible multiplications between the functions' imaginary and real parts are real integrable. We will discuss the proof of only one example; however, the other possibilities are the same. Note that the last line of the above lemma can be the multiplication of any two functions of $\{(\texttt{Im f}), (\texttt{Re f}), (\texttt{Im g}), (\texttt{Re g})\}$, e.g., $(\texttt{Im f}) * (\texttt{Im g})$.

Theorem 4.6 can be proved with the help of the following property:

Theorem 4.7.

$\forall$k l s. k real_measurable_on s $\wedge$ l real_integrable_on s

$\wedge$ ($\forall$x. x $\in$ s $\Rightarrow$ abs(k x) $\leq$ l x) $\Rightarrow$ k real_integrable_on s

According to Theorem 4.7, we need to prove that (Im f) $*$ (Re f) is measurable, which is easy since the multiplication of measurable functions is also measurable. Then, we find an integrable function l that is always greater than the absolute value of (Im f) $*$ (Re f) (see the third conjunction in Theorem 4.7). For this purpose, we select l = ((Im f)^2 + (Re f)^2)+ ((Im g)^2 + (Re g)^2) (which is the addition of the functions in the last two conjunctions of Theorem 4.6). This function l is intuitively integrable since the addition of integrable functions is also integrable. By proving the following simple algebraic property, we can then conclude Theorem 4.6:

Theorem 4.8.

$\forall$x. abs(2 $*$ (Im f x) $*$ (Re f x)) $\leq$ ((Im f)^2 + (Re f)^2) + ((Im g)^2 + (Re g)^2) x)

The second major lemma is about proving that r_inprod f f = Cx(0) $\Rightarrow$ f = cfun_zero. To this aim, we start by proving that a zero integrable positive function over the whole real line is zero integrable at any closed subinterval:

Theorem 4.9.

$\forall$f.real_integral(: real)f = 0 $\wedge$ ($\forall$x. 0 $\leq$ f x)

$\Rightarrow$ $\forall$a b. real_integral(real_interval [a, b])f = 0

where real_interval [a,b] is the real line from a to b.

Applying the fundamental theorem of calculus [44],

Theorem 4.10.

$\forall$f a b.

f real_integrable_on real_interval [a, b]

$\Rightarrow \exists$k. real_negligible k $\wedge \forall$x. x $\in$ real_interval [a, b] difference k

$\Rightarrow$ ((λx. real_integral(real_interval[a, x]) f) has_real_derivative f(x))

which proves that the composition of the derivative and integration operation acts as an identity operator to Theorem 4.9, which yields the following interesting result:

Theorem 4.11.

$\forall$f. $\exists$k. real_negligible k $\wedge \forall$x. (x $\in$ k) $\Rightarrow$ f(x)) = 0

where real_negligible k in Lebesgue and measure theories means that k has a very small number of elements (e.g., a finite set or empty set) that we can always neglect in any integration process. The problem here is that we are looking for a pure zero function to satisfy the inner product properties. For this purpose, mathematicians developed a new notion of "zero almost everywhere" to overcome this problem, in the case of space of square integrable complex-valued functions [44]:

cfun_almost_zero = $\exists$k. real_negligible k $\wedge \forall$x. (x $\in$ k) $\Rightarrow$ f(x) = Cx(0)

Hence, they update the property to be r_inprod f f = Cx(0) $\Rightarrow$ f = cfun_almost_zero Thus, we are finally able to prove the following quite important result, which proves that sq_integrable forms an inner product space, and hence a quantum states space:

Theorem 4.12.

is_inner_space (sq_integrable, r_inprod)

In the next section, we will use the above developed states space to build the basic

building block of a quantum optical beam.

4.2.1 Single Mode

Recall that the first step towards quantum optics formalization is implementing electromagnetic field quantization, which is classified according to the number of resonance frequencies. Thus, a single-mode field possesses single resonance frequency, which typically represents single-input/single-output optical devices, and a multi-mode field possesses a higher number of frequencies, which typically represents multi-input/multi-output optical devices. According to Section 2.3, we can then formally define a single mode as follows:

Definition 4.9.

$$\begin{aligned}
&\texttt{is_sm}\ (((\texttt{sq_integrable}, \texttt{r_prod}), \texttt{cs}, \texttt{H}), \texttt{w}, \texttt{vac}) \Leftrightarrow \\
&\quad \texttt{is_qsys}\ ((\texttt{sq_integrable}, \texttt{r_inprod}), \texttt{cs}, \texttt{H}) \wedge 0 < \texttt{w} \wedge \exists \texttt{q}\ \texttt{p}.\ \texttt{cs} = [\texttt{q}; \texttt{p}] \\
&\quad \wedge\ \texttt{is_observable}\ (\texttt{sq_integrable}, \texttt{r_inprod})\ (\texttt{p}) \wedge \\
&\quad \texttt{is_observable}\ (\texttt{sq_integrable}, \texttt{r_inprod})(\texttt{q}) \\
&\quad \wedge\ \texttt{H t} = \tfrac{\texttt{w}^2}{2}\%((\texttt{q t})\ \texttt{pow}\ 2) + \quad \tfrac{1}{2}\%((\texttt{p t})\ \texttt{pow}\ 2) \\
&\quad \wedge \texttt{is_qst}\ (\texttt{sq_integrable}, \texttt{r_prod})\ \texttt{vac} \wedge \texttt{is_eigen_pair}\ (\texttt{H t})\ (\texttt{vac}, \tfrac{\texttt{planck}*\texttt{w}}{2})
\end{aligned}$$

A single-mode field is characterized by five elements: the optical quantum states space `(sq_integrable,r_inprod)`, the list of the canonical observables of the mode `cs` (typically contains charges q and flux p), the energy operator `H`, the resonance frequency `w`, and finally the vacuum state `vac`. The predicate asserts that the system (i.e., the optical beam) should indeed be a valid system, that the frequency should be positive, and `vac` is the eigenstate of the energy operator `H` with the eigenvalue $\frac{\texttt{planck}*\texttt{w}}{2}$.

Based on Definition 4.9 and with the help of the quantum theory developed earlier, we can prove a number of elementary results. For instance, we can evaluate the commutator of our canonical coordinates $[q, p]$:

Theorem 4.13.

$\forall$sm. is_sm sm $\Rightarrow$

  commutator (q_of_sm sm) (p_of_sm sm) = (ii $*$ Cx planck) %I

We also prove that the energy operator H is indeed an observable (i.e., self-adjoint), based on the fact that q and p are also observables:

Theorem 4.14. $\forall$sp cs H ω. is_sm ((sp, cs, H), ω) $\Rightarrow$ is_observable sp H

Zero Point Energy

Now, we will revisit the formal version of the minimum energy theorem informally presented earlier in Section 2.3. Recall that we need to show that an optical field always contains energy greater than or equal to $\frac{\hbar\omega}{2}$, where $\hbar$ is the Planck constant, even though no charge or flux exist. We start by defining the creator and annihilator according to Equations (2.6) and (2.5), respectively, as follows:

Definition 4.10.

anh_sm (sm_sys : sm) = a1_sm sm_sys + ii % a2_sm sm_sys

cr_sm (sm_sys : sm) = a1_sm sm_sys − ii % a2$_s$m sm_sys

Accordingly, we re-express the energy $\hat{H}$ in terms of creation and annihilation operators. This rewriting step plays an important role since it helps in defining the concept of photons, as we will see later:

Theorem 4.15.

$\forall$ vac w coord H.

  let sm = (((sq_integable, r_inprod), coord, H), w, vac) in

$0 < \texttt{planck} \wedge \texttt{is\_sm sm} \Rightarrow$

$\texttt{H} = \texttt{Cx}(\texttt{planck} * \omega)\ \%\ (\texttt{cr\_smsm}\ **\ \texttt{anh\_smsm} + \texttt{Cx}(\frac{1}{2})\ \%\ \texttt{I})$

The first lines define an abbreviation for the long construct of the single-mode `sm`, since it is frequently used in the body of the theorem. For this definition of energy, we then prove the minimum energy theorem as follows:

Theorem 4.16.

$\forall \texttt{w coord H qst}.$

$\texttt{let sm} = (\omega, ((\texttt{sq\_integable}, \texttt{r\_inprod}), \texttt{w}, \texttt{coord}, \texttt{H}))\ \texttt{in}$

$0 < \texttt{planck} \wedge \texttt{qst}\ \in\ \texttt{sq\_integable} \wedge \texttt{is\_sm sm}$

$\Rightarrow \frac{\texttt{planck}*\omega}{2} \leq \texttt{real\_of\_complex}\ (\texttt{expectation inprod qst H})$

The last line shows the lower bound, i.e., the minimum energy, of the expectation of the energy `H`. However, we mentioned earlier that the energy itself is lower bounded, not its expectation. We can explain this as follows: The states at which the expectation of energy is equal to the minimum energy are the eigenstates of the energy operator `H`. And according to Theorem 4.1, measurements at the eigenstates are deterministic, thus they are equal to the expectation. Note that the state that typically satisfies this theorem is the vacuum state `vac`.

Annihilation and Creation Operators

The energy spectrum in quantum optics not only has a minimum regardless of the field states, it also has a discrete nature. This is in contrast to classical theory, which considers it a continuous one. This phenomenon can be explained by studying the effect of the "annihilation operator" and "creation operator" on the system energy. The following two theorems provide such an effect and give a justification for naming

such two operators with these names. First, the effect of `anh_sm`:

Theorem 4.17.

$\forall$ `vac w coord H qst en`.

`let sm` $=$ `(((sq_integable, r_inprod), coord, H), w, vac) in`

$0 <$ `planck` $\wedge$ `qst` $\in$ `sq_integable` $\wedge$ `is_sm sm`

`is_eigen_pair H qst en`

$\Rightarrow$ `is_eigen_pair H (anh_sm sm qst) (en` $- \hbar\omega$`)`

The last line of the theorem shows that the resulting state (called the demoted state) `(a qst)` is an eigenstate of `H`, and its energy is decreased by $h\omega$. Similarly, the "creation operator" increases the energy of the excited state by $\hbar\omega$:

Theorem 4.18.

$\forall$ `vac w coord H qst en`.

`let sm` $=$ `(((sq_integable, r_inprod), coord, H), w, vac) in`

$0 <$ `planck` $\wedge$ `qst` $\in$ `sq_integable` $\wedge$ `is_sm sm`

`is_eigen_pair H qst en`

$\Rightarrow$ `is_eigen_pair H (cr_sm sm qst) (en` $+ \hbar\omega$`)`

It is important to mention here that it is not necessary for the excited state (i.e., `herm_a qst`) or the demoted state (i.e., `a qst`) to be a quantum state; if the norm of the new state is one, then it is a quantum state; otherwise, the normalized version is the new quantum state. By combining the minimum energy theorem and the effect of creation and annihilation operators we can prove that the amount of energy inside a field follows this form: $0.5\hbar\omega$, $1.5\hbar\omega$, $2.5\hbar\omega$, $3.5\hbar\omega$, etc; if we allow any intermediate value then applying the annihilation operator repeatedly would yield an energy less than $0.5\hbar\omega$, which violates the minimum energy theorem.

4.2.2 Fock States

Given the discrete spectrum of the energy contained in an optical beam, it can be deduced that there are particles of constant energy $\hbar\omega$ inside the field, and if the number of such particles increases or decreases by n, then the whole amount of energy inside the field is affected by $n * \hbar\omega$. This is typically the notion of photons, each of which has energy of $\hbar\omega$. The number of photons inside an optical beam can be observed via the quantum operator `N`, and is formally defined as follows:

Definition 4.11.

$$\texttt{phn\_sm sm\_sys} = \texttt{cr\_sm sm\_sys} * \texttt{anh\_sm sm\_sys}$$

Recall that $\texttt{cr\_sm sm\_sys} * \texttt{anh\_sm sm\_sys} = \frac{\texttt{H}}{\texttt{Cx}(\hbar * \omega)} - \frac{1}{2}$ (see Theorem 4.15), which exactly calculates the number of photons. Accordingly, we have proved some essential properties for the photons number operator, e.g., self-adjointness, the formal theorem is as follows:

Theorem 4.19.

$\forall$ `vac w coord H`.

$$\texttt{let sm} = (((\texttt{sq\_integable}, \texttt{r\_inprod}), \texttt{coord}, \texttt{H}), \texttt{w}, \texttt{vac}) \texttt{ in}$$

$$0 < \texttt{planck} \wedge \texttt{is\_sm sm}$$

$$\texttt{is\_sm sm} \Rightarrow \texttt{is\_self\_adjoint } (\texttt{sq\_integable}, \texttt{r\_inprod}) \texttt{ phn\_sm sm}$$

Indeed, the photon number operator `N` has a strong relation with the energy operator `H`. The following theorem shows that an eigenstate of `H` is also an eigenstate of `N`, but typically with different eigenvalues since they are measuring (or observing) different quantities:

Theorem 4.20.

$\forall$ vac w coord H qst en.

  let sm = (((sq_integable, r_inprod), coord, H), w, vac) in

  0 < planck $\wedge$ qst $\in$ sq_integable $\wedge$ is_sm sm

1 is_eigen_pair H qst en $\Leftrightarrow$

2 is_eigen_pair (phn_sm sm)) qst ((Cx($\frac{1}{\text{planck}*\text{w}}$)) $*$ en $-$ Cx$\frac{1}{2}$)

Note that we can prove, based on the above theorem, that the vacuum state vac is an eigenvector of the photon number operator N with zero photons. This theorem extends the effect of the creator and annihilator to the photon number eigenstates. However, in this cases they increase (decrease) the number of photons by one.

The photon number eigenstates are called fock states, which are quite important in quantum optics since they form the span set of the optical quantum states space (see Section 2.3). Moreover, it is widely used in the development of single-photon devices which have direct applications in quantum cryptography. Recall that a fock state $|n\rangle$ describes an optical beam of exactly n photons. Thus, with the help of the vacuum state (i.e., a fock state with zero photons) and the effect of the creator on photon number eigenstates, we formally define a fock state as follows:

Definition 4.12.

fock sm 0 = vac $\wedge$ fock sm (SUC n) =

  get_qst r_inprod (cr_sm sm (fock sm n)))

As shown, it is recursively defined with vac state as the base case. Then, we can get any higher fock state by applying the creation operator. The function get_qst returns the normalized version of a vector, i.e., by dividing the state by the norm of the vector itself. This is to ensure that the norm of the resulting state is equal to one.

For this definition, we have proved that a fock state is indeed a quantum state, i.e., normalized and belongs to sq_integrable:

Theorem 4.21.

$\forall$ vac w coord H.

let sm = (((sq_integable, r_inprod), coord, H), w, vac) in

is_sm sm $\Rightarrow \forall$n. is_qst ((sq_integable, r_inprod))(fock sm n)

The following two theorems express the effect of the creator/annhilator on the fock states as discussed. Concretely, they correspond to Equations (2.14) and (2.15):

Theorem 4.22.

$\forall$n sm.

is_sm sm $\Rightarrow$ (anh_sm sm) (fock sm (SUC n)) $= \sqrt{\text{SUC n}}$ % fock sm n

Since the state number in the left hand side is SUC n, then the theorem is valid for all fock states except at zero, i.e., the vac state. However, the following theorem is valid for any state including the vac state:

Theorem 4.23.

$\forall$n sm.

is_sm sm $\Rightarrow$ (cr_sm sm) (fock sm n) $= \sqrt{\text{SUC n}}$ % fock sm (SUC n)

The above formulas are recurrence relations, and by solving any of them, we can get a non-recursive definition of the fock state. The following provides the solution of the recurrence relation of Theorem 4.23:

Theorem 4.24.

$\forall$ vac w coord H.

let sm = (((sq_integable, r_inprod), coord, H), w, vac) in

is_sm sm

$\Rightarrow \forall$m. fock sm m $= \frac{1}{\sqrt{\text{m!}}}$ % (cr_sm sm pow m) vac

In the following, we will utilize the formal development of fock states in the formalization of the coherent states.

4.2.3 Coherent States

We described earlier the *uncertainty principle* and how it is important in the development of many quantum concepts, in particular coherent states. The principle admits that performing measurements of a quantum observable affects the measurements accuracy of other observables. In 1926, Schrödinger discovered coherent states that achieve a minimal measurement error for both observables [60]. In other words, the measurement variance of two non-commuting observables at a coherent state is equal.

Recall the definition of coherent states in Equation (2.13). We can formally develop it in terms of fock states and infinite summation as follows:

Definition 4.13.

$$\texttt{coherent sm } \alpha =$$
$$\exp(-\tfrac{|\alpha|^2)}{2})) \ \% \ \texttt{cfun\_infsum } (\texttt{sq\_integrable}, \texttt{r\_inprod}) \ (\texttt{from 0})$$
$$(\lambda \texttt{n}. \tfrac{\alpha^{\texttt{n}}}{\sqrt{\texttt{n}!}} \% (\texttt{fock sm n}))$$

where α is the state parameter. Similar to any definition that involves infinite sets, we need to make sure that it converges. We have handled a similar situation in Section 3.2 by defining the `summable` predicate:

Definition 4.14.

$$\texttt{coherent\_summable sm } \alpha \Leftrightarrow$$
$$\texttt{cfun\_summable } (\texttt{sq\_integrable}, \texttt{r\_inprod}) \ (\texttt{from 0}) \ (\lambda \texttt{n}. \tfrac{\alpha^{\texttt{n}}}{\sqrt{\texttt{n}!}} \% (\texttt{fock sm n}))$$

Next, we prove the essential property of coherent states, that they are eigenstates

of the annihilation operator. Theorem 4.22 plays a crucial role in proving such a relation. However, this theorem is only valid for fock states greater than zero (i.e., vac state). Consequently, we have to rewrite the coherent definition in a way that allows the application of Theorem 4.22:

Theorem 4.25.

$\forall$ vac w coord H.

let sm = (((sq_integable, r_inprod), coord, H), w, vac) in

coherent_summable sm $\alpha \Rightarrow$

coherent sm a = $\exp(-\frac{|\alpha|^2)}{2})$) %(vac+

cfun_infsum (sq_integrable, r_inprod) (from 0) (λn.$\frac{\alpha^{(\text{SUC n})}}{\sqrt{(\text{SUC n})!}}$%(fock sm (SUC n))))

It is important to mention here that vac is a coherent state with $\alpha = 0$. This can be proved by showing that the vac state is an eigenvector of the annihilator:

Theorem 4.26.

$\forall$ vac w coord H.

let sm = (((sq_integable, r_inprod), coord, H), w, vac) in

is_sm sm $\Rightarrow$ is_eigen_pair (anh_sm sm) (vac, Cx(0))

We can appreciate the importance of the vac state since it acts as a coherent and a fock state at the same time. Fortunately, this allows us to use the properties of both notions, which is very helpful.

Now, we can prove that coherent states are eigenvectors of the annihilation operator, with eigenvalue α based on Theorems 4.22 and 4.25 as follows:

Theorem 4.27.

$\forall$sm α.

is_sm sm $\wedge$ coherent_summable sm $\alpha \wedge$

$\wedge$is_bounded (sq_integrable, r_inprod) (anh_sm sm)

$\Rightarrow$ is_eigen_pair(cr_sm sm) (coherent sm α, α)

Note that the annihilator should be a bounded operator in order to swap it with the infinite summation (see Theorem 3.24). Such a swapping happens in many instances throughout the proof.

According to the above relation between coherent states and an annhilator, we can prove an interesting property that shows how a complicated expression that involves operator exponentiation can be turned into a simple scalar multiplication:

Theorem 4.28.

$\forall$sm α.

is_sm sm $\wedge$ coherent_summable sm $\alpha \wedge$

$\wedge$is_bounded (sq_integrable, r_inprod) (anh_sm sm)

$\Rightarrow \forall$ n.((anh_sm sm) pow n) coherent sm $\alpha =$

(α pow n) % coherent sm α

Now, we conclude the formalization of coherent states which will be extended in the next chapter for the sake of building quantum gates, where they are being presented in terms of *displacement operators*.

4.2.4 Multi-Mode Formalization

Up to this point, all development is serving optical systems of single-input/single-output, which is not the practical case. In fact more complicated optical devices and circuits, e.g., quantum gates, are based on multi-mode fields. As described in Section 2.3.2, the core idea of the multi-mode formalization is based on the development of the tensor product. Before we present the general definition of the quantum states tensor product, we will show an example of the tensor product of only two states.

Recall that a quantum state has a probabilistic nature, i.e., for an optical beam it provides the probability density function of the number of photons inside the optical beam. Now, if we have two beams described with states $|n_1\rangle$ and $|n_2\rangle$, the function that describes the joint probability of the two beams is the point-wise multiplication of $|n_1\rangle$ and $|n_2\rangle$. Hence, we define the tensor product of two quantum states as follows: $\lambda y_1\ y_2.\ |n_1\rangle\ y_1 * |n_1\rangle\ y_2$. To generalize for n beams, we define the tensor product recursively as follows:

Definition 4.15.

$\texttt{tensor 0 (modes : bqs}^{\texttt{N}}\texttt{)} = \texttt{K(Cx(1))} \wedge$

$\quad \texttt{tensor (SUC n) (modes)} =$

$\quad\quad (\lambda \texttt{y} : \texttt{A}^{\texttt{N}}.((\texttt{tensor n modes})\ \texttt{y}) * (\texttt{modes\$(SUC n)})\ (\texttt{y\$(SUC n)}))$

where `modes` is a vector of size n that contains n modes. The base case of the zero mode is a trivial case; it only guarantees a terminating definition. We then define the tensor product of operators as follows:

Definition 4.16.

$\texttt{is\_tensor (tens : cops}^{\texttt{N}} \longrightarrow (\texttt{real}^{\texttt{N}} \longrightarrow \texttt{complex}) \longrightarrow (\texttt{real}^{\texttt{N}} \longrightarrow \texttt{complex})) \Rightarrow$

$\quad \forall(\texttt{ops} : (\texttt{bqs} \longrightarrow \texttt{bqs})^{\texttt{N}})\ (\texttt{modes} : \texttt{bqs}^{\texttt{N}})\ \texttt{n.}\ \texttt{is\_linear\_cop (tens ops)} \wedge$

$\quad \texttt{tens ops (tensor n modes)} = \texttt{tensor n(lambda i.(ops\$i) (modes\$i))}$

where `ops` is a vector of the operators defined on the single-modes, and `tens ops` is the tensor product. Note that the resulting new operator is only applicable to the tensor product of states. That is why we define it in a predicate form in order to restrict its functionality. For this definition, we prove the following crucial property of the operators tensor product, associativity:

Theorem 4.29.

$\forall$ `ten ops1 ops2 n modes`.

`is_tensor ten` $\Rightarrow$ `ten ops2(ten ops1 (tensor nmodes))` $=$

`ten ((lambda i. (ops2$i) o (ops1$i))) (tensor n modes)`

As we will see later, an optical quantum circuit accepts single-modes as inputs; however, the circuit operation itself runs in multi-mode. Thus, we need to develop a function to embed (or express) a single-mode operator in a multi-mode fashion. For this purpose, we define the following function:

Definition 4.17.

`pos (tens : cops`$^{\mathtt{N}}$ $\longrightarrow$ `(A`$^{\mathtt{N}}$ $\longrightarrow$ `complex)` $\longrightarrow$ `(A`$^{\mathtt{N}}$ $\longrightarrow$ `complex)) (op : cops) m` $=$

`tens` (λ `i. if i` $=$ `m then op else I`)

The concept of `pos` (or positioning) is to place a given operator in a specific mode (based on its order in the input list) and leave the other modes with the identity operator.

By the development of multi-mode, we have finished the formal foundation of quantum optics that allows the formal modeling, analysis and verification of optical quantum circuits, in particular quantum gates.

4.3 Summary

In this chapter, we have covered the formalization of crucial notions of quantum optics for the sake of the modeling and verification of quantum computing circuits. Firstly, we built the general quantum mechanics rules based on the `cfun` library. In particular, we defined the concepts of quantum states, operators and systems and

proved the uncertainty principle, a pillar of quantum theory. We then customized the general rules for optical beams by defining a specific type `bqs`, which in turn leads to the development of the square integrable space L^2, where optical quantum states reside. Having such a concrete foundation, we formalized single-mode fields which mimic the single-input/single-output optical systems. We then proved several theorems, in particular the interesting one of minimum energy. The fock states are then implemented and their relation with the photon number operator was proved. Since they form the basis of the optical quantum states space L^2, they were utilized to develop the coherent states which are of special interest in the development of quantum computers. Finally, we generalized our work by formalizing multi-mode fields, which allows practicable applications to be tackled, as will be illustrated in the next chapter. The whole development of this library amounts to 1200 lines of HOL scripts including 140 theorems and 20 definitions.

Chapter 5

Applications

This chapter is considered the tangible product of the thesis, where all the before mentioned formal development is utilized to build formal models of optical devices and circuits, and reason about them. In particular, we have implemented four optical elements: the *coherent light displacer*, *optical phase shifter*, *beam splitters*, and *mirrors*. Based on these elements, we have formally verified the behavior of three quantum computing circuits: *Mach-Zehender Interferometer*, *Flip gate* and *Controlled NOT gate*. These elements and circuits cover both single-mode and multi-mode cases.

5.1 Coherent Light Displacer

One of the possible presentations of coherent light is using the *displacement operator* $D(\alpha)$:

$$|\alpha\rangle = D(\alpha)|0\rangle \tag{5.1}$$

where, $D(\alpha) = e^{\alpha\hat{a}^\dagger}\ e^{-\alpha^*\hat{a}}\ e^{[\alpha\hat{a}^\dagger, -\alpha^*\hat{a}]}$. Recall that α^* is the complex conjugate of α, $**$ denotes the multiplication between quantum operators, and $[op_1, op_2] = op_1 *$

$* \; op_2 - op_2 \; ** \; op_1$. Note the use of exponentiation *over operators*, which is defined as follows:

$$e^{\hat{O}} = \sum_{i=0} \frac{\hat{O}^i}{i!} \tag{5.2}$$

The idea behind such representation is that the displacement operator D is a real optical element that has interesting properties. For a coherent beam $|\alpha\rangle$ that passes through an optical displacer $D(\beta)$ [9], its coherence degree α is displaced by β, which results in producing a coherent beam $|\alpha+\beta\rangle$. It is clear that the formalization of such a device requires the development of *operator Exponentiation*, as defined in Equation 5.2.

Operator Exponentiation Formalization

This formalization is completely dependent on the infinite summation over `cfun`, which was presented in Section 3.2. We start by defining the infinite summation over quantum operators, which is a pointwise infinite summation over complex functions:

Definition 5.1.

`cop_sums (s, inprod) f l set` $\Leftrightarrow$ `∀x. x IN s` $\Rightarrow$

`cfun_sums (s, inprod) (λn.(f n) x) (l x) set`

Note that this definition is an adaptation of the `cfun` case: the only differences are the types of `f`, `l`, and `set`, and the fact that the pointwise definition is restricted to functions that belong to the inner space.

Similarly to `cfun_infsum` and `cfun_summable`, we then define `cop_infsum` and `cop_summable`:

Definition 5.2.

`cop_infsum innerspc s f` = `@l. cop_sums innerspc f l s`

`cop_summable innerspc s f` = `∃l. cop_sums innerspc f l s`

Finally, we can use `cop_infsum` to define quantum operator exponentiation according to Equation (5.2):

Definition 5.3.

$\texttt{cop\_exp innerspc (op : cfun} \longrightarrow \texttt{cfun)} \Leftrightarrow$

$\quad \texttt{cop\_infsum innerspc (from 0) } (\lambda \texttt{n}. \frac{1}{!\texttt{n}} \ \% \ (\texttt{op pow n}))$

where `from 0` denotes the set of natural numbers $\mathbb{N}$. We proved a number of properties about the exponentiation, and we mention here the important ones that are used in the development of our optical circuits.

The following theorem is about proving that `cop_exp (cop_zero) = I`, the scalar counterpart of which is $e^0 = 1$:

Theorem 5.1.

$\forall \texttt{s inprod x}.$

$\quad \texttt{x IN s} \wedge \texttt{is\_inner\_space(s, inprod)} \Rightarrow$

$\qquad \texttt{cop\_exp (s, inprod) cop\_zero x} = \texttt{x}$

where $\texttt{cop\_zero} = \lambda \texttt{x} : \texttt{cfun}.$ `cfun_zero` is the operator. Note that we did not prove the explicit `cop_exp (cop_zero) = I`, where we cannot reason about the behavior of `cop_exp (cop_zero)` outside the `s`. Thus, we restrict the theorem inside `s`. Recall that the existence of `cop_exp` is coupled with the existence of an inner space `(s,inprod)`.

We also prove the interesting result about linearity preservation: for a linear operator `op`, the exponentiation `cop_exp op` is linear too:

Theorem 5.2.

$\forall$ s inprod op.

cop_summable innerspc (from 0) (λn.$\frac{1}{!n}$ % (op pow n)) $\wedge$ is_linear_cop op $\Rightarrow$

is_linear_cop (cop_exp (s, inprod) op)

This property is essential in the development of the flip gate as we will see later: it allows the generalization of the effect of the gate (typically a cascade of quantum operator exponentiations) on the basis states $|\psi_1\rangle$ and $|\psi_2\rangle$ to any mixed state $c_1 |\psi_1\rangle + c_2|\psi_2\rangle$.

Accordingly, an optical displacer can be formalized as cop multiplication of three cop_exp as follows:

Definition 5.4.

disp sm α =

(cop_exp (sq_integrable, r_inprod) (α % cr_sm sm) **

cop_exp (sq_integrable, r_inprod) (−(cnj v) % anh sm) **

cop_exp (sq_integrable, r_inprod) ((v % cr_sm sm) com ((cnj v) % anh_sm sm))

Recall that op1 com op2 = op1 ** op2 - op2 ** op1 (called the *commutator* of op1 and op2), and cr_sm and anh_sm are the creator and annihilator, respectively. We then proved the typical behavior of such a device: when it receives vacuum at the input port it generates a coherent beam $|\alpha\rangle$:

Theorem 5.3.

$\forall$ vac w coord H.

let sm = (((sq_integable, r_inprod), coord, H), w, vac) in

is_sm sm $\wedge$ exp_summable (sq_integrable, r_inprod) (cnj($-\alpha$) %anh_sm sm)

$\wedge$ cfun_summable (sq_integrable, r_inprod) (from 0)(λn.$\frac{\alpha \text{ pow n}}{\sqrt{!n}}$ % fock sm n)

is_sm sm $\wedge$ exp_summable (sq_integrable, r_inprod) (α cr_sm sm)

$\Rightarrow$ (disp sm α) vac = coherent sm α

The last two conjunctions ensure the existence of a convergent sequence that generates such a coherent state. The above theorem represents the formal version of Equation (5.1).

5.2 Optical Phase Shifter

A phase shifter typically causes a phase shift to a fock beam that passes through it, while keeping the same fock state, i.e., changes the beam directions but keeps the number of photons as is [48]. An optical phase shifter is expressed mathematically as follows:

$$U(\theta) = e^{i\theta\ \hat{N}} \tag{5.3}$$

where θ is the shifting angle, $\hat{N}$ is the photon number operator, and $U(\theta)\ |n\rangle = e^{i\theta}\ |n\rangle$. Accordingly, we formally define an optical shifter as follows:

Definition 5.5.

shifter sm θ =

  cop_exp (sq_integrable, r_inprod) (iθ % phn_sm sm)

and the following theorem shows the effect of the shifter on a fock beam:

Theorem 5.4.

$\forall$ vac w coord H.

  let sm = (((sq_integable, r_inprod), coord, H), w, vac) in

  is_sm sm $\wedge$ exp_summable (sq_integable, r_inprod) (iθ % n_of_sm sm)

    $\Rightarrow$ shifter sm θ (fock sm n) =

      (cexp (iθ)) % (fock sm n)

Note that we have to confirm the convergence of the shifter operator using `exp_summable`.

Since fock states are basis states, and any mixed states can be expressed in terms of such a basis, then the optical shifter has an effect similar to other optical states in general. In the following, we show an interesting result of applying the optical phase shifter to coherent light. By selecting the shifting angle θ to be π, the shifter will operate as a mirror (typically called a *phase conjugating mirror*):

Definition 5.6.

$\texttt{ph\_mirror sm} =$

$\quad \texttt{cop\_exp (sq\_integrable, r\_inprod) (i}\pi \ \% \ \texttt{n\_of\_sm sm)}$

For a coherent beam that passes through the phase conjugating mirror, it is reflected in the reverse direction, as is shown in the next theorem:

Theorem 5.5.

$\forall \ \texttt{vac w coord H}.$

$\quad \texttt{let sm} = (((\texttt{sq\_integable, r\_inprod}), \texttt{coord, H}), \texttt{w, vac}) \ \texttt{in}$

$\quad \texttt{is\_sm sm} \wedge \texttt{cfun\_summable (sq\_integable, r\_inprod) (from0)} \ (\lambda \texttt{n}. \frac{\alpha^{\texttt{n}}}{\sqrt{!\texttt{n}}} \ \% \ \texttt{fock sm n})$

$\quad \wedge \ \texttt{ph\_mirror\_summable sm} \wedge \texttt{is\_bounded (sq\_integable, r\_inprod) (mirror sm)}$

$\quad\quad \Rightarrow \texttt{ph\_mirror sm (coherent sm } \alpha) = \texttt{coherent sm } (-\alpha)$

where `ph_mirror_summable` is similar to the summable notions defined before: we define a new predicate only for simplicity.

5.3 Quantum Flip Gate

A quantum bit is (or *qbit*) a quantum system with two basis states, $|0\rangle$ and $|1\rangle$. However, contrary to its classical counterpart, the state of a qbit is not only $|0\rangle$ or $|1\rangle$,

but can be a mix thereof. Indeed, such a state can be expressed as $|\psi\rangle = \beta|0\rangle + \gamma|1\rangle$, where $|\beta|^2 + |\gamma|^2 = 1$ (according to Equation (2.11)).

In order to compute with qbits, one needs operators applied to them. As for classical circuits, this is achieved through *gates*. The quantum computer model is made of nine such gates, e.g., *quantum flip gate, Controlled NOT gate, Swap gate and Phase-shift gate* [35]. For instance the quantum flip gate, which is equivalent to the classical NOT gate, converts $|0\rangle$ to $|1\rangle$ and vice versa. However, due to its quantum nature, it is capable of much more: for any β and γ, $\beta|0\rangle + \gamma|1\rangle$ is turned into $\gamma|1\rangle + \beta|0\rangle$.

We mentioned earlier in Chapter 1 that quantum computers can be implemented in different technologies. The major difference among these implementations is how the qbits are realized. In this application, we focus on the coherent light quantum bits, where the states $|0\rangle$ and $|1\rangle$ are realized by $|vac\rangle$ and $|\alpha\rangle$, respectively. In this context, the specification of a flip gate is that it converts $\beta\ |vac\rangle + \gamma\ |\alpha\rangle$ into $\gamma\ |\alpha\rangle + \beta\ |vac\rangle$.

The intended implementation of the gate consists of a displacer $D(-\alpha)$, followed by a phase conjugating mirror (see Figure 5.1).

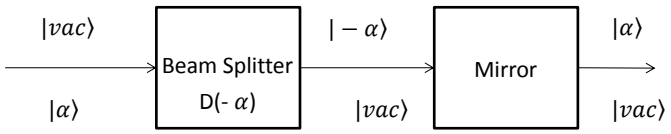

Figure 5.1: Optical Quantum Flip Gate

We start by demonstrating the effect of the proposed optical flip gate on each optical qbit separately. Then, we generalize the result to any mixed qbit by using the linearity of the quantum operator exponentiation. In a case in which `vac` is the input, according

to Theorem 5.3, the displacer will generate a coherent light beam $|-\alpha\rangle$. This beam then hits the mirror and is reflected back to generate $|\alpha\rangle$ according to Theorem 5.5. In case $|\alpha\rangle$ is the input, the displacer completely destructs the optical beam and generates `vac`, the formal theorem follows:

Theorem 5.6.

$\forall$ `vac w coord H`.

`let sm` $=$ `(((sq_integable, r_inprod), coord, H), w, vac) in`

`is_sm sm` $\wedge$ ($\forall$`b.exp_summable (sq_integable, r_inprod) (b%a_of_sm sm`))

$\wedge$`coherent_summable sm` α

$\wedge$ `exp_summable (sq_integable, r_inprod)` (α `cr_sm sm`)

$\wedge$ `is_bounded (sq_integable, r_inprod) (anh_sm sm`))

$\wedge$ ($\forall$`x op. is_linear_cop op` $\wedge$ `x IN s` $\Rightarrow$

(`cop_exp (s, inprod) (`$-$`op)` $**$ `cop_exp (s, inprod) (op)) x = x`)

$\Rightarrow$ `disp sm` ($-\alpha$) (`coherent sm` α) $=$ `vac`

The last conjunction in the premises shows an assumed property about the exponentiation of quantum operators. Such a property requires the proof of the general theorem of Baker-Campbell-Hausdorff [26] [1]. The mirror then does not have any effect on `vac`, and keeps it as is (see Theorem 5.5). Recall that `vac = coherent sm 0`.

Now, we have all the ingredients to construct the flip gate and verify its behavior. The formal definition of the flip gate is made through the cascading of the phase conjugating mirror and the displacer. This can be defined as an operators' multiplication (i.e., function composition):

[1]The proof of the Baker-Campbell-Hausdorff theorem is very complex and requires a lot of prerequisites that are not available in HOL Light, and is hence outside of the scope of this work

Definition 5.7.

$\texttt{flip\_gate}\ \alpha\ \texttt{sm} = (\texttt{ph\_mirror sm})\ **\ (\texttt{disp sm}\ (-\alpha))$

Based on the above definition and using Theorems 5.5 and 5.6, we prove the correctness of the gate behavior in one single theorem as follows:

Theorem 5.7.

$\forall\ \texttt{vac w coord H}.$

$\texttt{let sm} = (((\texttt{sq\_integable}, \texttt{r\_inprod}), \texttt{coord}, \texttt{H}), \texttt{w}, \texttt{vac})\ \texttt{in}$

$\texttt{is\_sm sm} \wedge \texttt{exp\_summable}\ (\forall \texttt{b}.\ (\texttt{sq\_integable}, \texttt{r\_inprod})\ (\texttt{b\%a\_of\_sm sm})$

$\wedge\ (\forall \texttt{b}.\ \texttt{coherent\_summable sm b})$

$\wedge\ (\forall \texttt{c}.\ \texttt{cfun\_summable}\ (\texttt{sq\_integable}, \texttt{r\_inprod})\ (\texttt{from 0})\ (\lambda \texttt{n}.(\frac{\texttt{c}^{\texttt{n}}}{\sqrt{!\texttt{n}}})\%\texttt{fock sm n}))$

$\wedge\ (\forall \texttt{d}.\ \texttt{exp\_summable}\ (\texttt{sq\_integable}, \texttt{r\_inprod})\ (\dot{\%}\texttt{creat\_of\_sm sm}\ (0)))$

$\wedge\ \texttt{is\_bounded}\ (\texttt{sq\_integable}, \texttt{r\_inprod})\ (\texttt{anh\_sm sm})$

$\wedge\ (\texttt{cop\_exp}\ (\texttt{s}, \texttt{inprod})\ (-\texttt{op}) ** \texttt{cop\_exp}\ (\texttt{s}, \texttt{inprod})\ (\texttt{op}))\ \texttt{x} = \texttt{x})$

$\wedge\ \texttt{ph\_mirror\_summable sm} \wedge \texttt{is\_bounded}\ (\texttt{sq\_integable}, \texttt{r\_inprod})\ (\texttt{ph\_mirror sm})$

$\Rightarrow\ (\texttt{flip\_gate}\ \alpha\ \texttt{sm})\ (\texttt{coherent sm}\ \alpha) = \texttt{vac}$

$\wedge\ (\texttt{flip\_gate}\ \alpha\ \texttt{sm})\ \texttt{vac} = \texttt{coherent sm}\ \alpha$

In a nutshell, Theorem 5.7 proves that a coherent beam $|\alpha\rangle$ ($|vac\rangle$) passes through a beam splitter, which in turn generates $|vac\rangle$ ($|-\alpha\rangle$), then the beam encounters a mirror which reflects it in the opposite direction to generate $|vac\rangle$ ($|\alpha\rangle$). Hence, we have the realization of the quantum flip gate. Note that given the linearity of the optical elements, this result generalizes for any mixed state $c_1 * |\alpha\rangle + c_2 * |vac\rangle$.

5.4 Beam Splitter

A beam splitter is a device that takes a beam of light and partly transmits it and partly reflects it, thus splitting the beam into two beams. The remarkable feature of quantum mechanics is that a *single photon* can be split by a beam splitter.

In its standard definition, a beam splitter consists of two-input/two-output ports [22]. We can recognize each port (or optical mode) by the creator and annihilator operators, as shown in Figure 5.2:

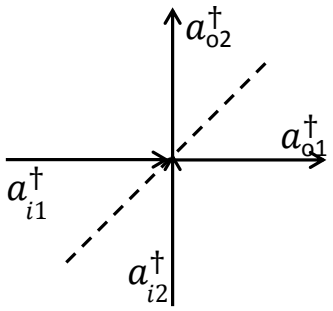

Figure 5.2: Beam Splitter

The beam splitter then relates input modes with the output modes according to the following matrix representation:

$$\begin{pmatrix} \hat{a}^\dagger_{o1} \otimes I \\ I \otimes a^\dagger_{o2} \end{pmatrix} = \begin{pmatrix} \mathbf{T}' & \mathbf{R} \\ \mathbf{R}' & \mathbf{T} \end{pmatrix} \begin{pmatrix} \hat{a}^\dagger_{i1} \otimes I \\ I \otimes a^\dagger_{i2} \end{pmatrix} \tag{5.4}$$

with the following relations between the coefficients :

$$|\mathbf{R}'| = |\mathbf{R}|,\ |\mathbf{T}'| = |\mathbf{T}|,\ |\mathbf{R}|^2 + |\mathbf{T}|^2 = 1,$$

$$\mathbf{R}^*\mathbf{T}' + \mathbf{R}'\mathbf{T}^* = 0, \quad \text{and } \mathbf{R}^*\mathbf{T} + \mathbf{R}'\mathbf{T}'^* = 0.$$

These coefficients are of type complex and represent reflectivity and transitivity in

some sense. We now have the quantum mechanical description of the beam splitter, and thus we can develop its formal version as follows:

Definition 5.8.

1 $\texttt{is\_beam\_splitter}\ (\texttt{p1}, \texttt{p2}, \texttt{p3}, \texttt{p4}, \texttt{ten}, \texttt{i1}, \texttt{m1}, \texttt{i2}, \texttt{m2}, \texttt{o1}, \texttt{m3}, \texttt{o2}, \texttt{m4}) \Leftrightarrow$

2 $\texttt{is\_sm i1} \wedge \texttt{is\_sm i2} \wedge \texttt{is\_sm o1} \wedge \texttt{is\_sm o2}$

3 $\wedge\ \texttt{w i1} = \texttt{w i2} \wedge \texttt{w i2} = \texttt{w o1} \wedge \texttt{w o1} = \texttt{w o2} \wedge$

4 $\texttt{vac i1} = \texttt{vac i2} \wedge \texttt{vac i2} = \texttt{vac o1} \wedge \texttt{vac o1} = \texttt{vac o2} \wedge$

5 $\texttt{pos ten (cr i1) m1} = \texttt{p1}^{*}\% \ \texttt{pos ten (cr o1) m3} + \texttt{p2}^{*}\% \ \texttt{pos ten (cr o2) m4}$

6 $\texttt{pos ten (cr i2) m2} = \texttt{p3}^{*}\% \ \texttt{pos ten (cr o1) m3} + \texttt{p4}^{*}\% \ \texttt{pos ten (cr o2) m4}$

Note that the formal definition of beam splitters relates the inputs operators in terms of the outputs operators (see Lines 5 and 6), in contrast to the theoretical definitions presented earlier in Equation (5.4): This form is practical for the analysis of the circuits, as we will see later, since the goal is to generate the output states from the input states. Thus, the parameters {p1,p2,p3,p4} are the inverse of the matrix presented before. In Line 1, the parameters {m1,m2,m3,m4} define the order of each mode in the whole circuit. In the case of a circuit of only two inputs/two outputs, the possible values of these parameters are 1 and 2. Lines 2 and 3 ensure that the four modes are proper single modes, and working with the same frequency and vacuum state. We have proved that this device is energy-loss less by indicating that the energy at the input ports is equal to the energy at the output ports:

Theorem 5.8.

$\forall\ \texttt{bs}.\ \texttt{is\_beam\_splitter bs} \Rightarrow \texttt{H}_{\texttt{in}_1} + \texttt{H}_{\texttt{in}_2} = \texttt{H}_{\texttt{out}_1} + \texttt{H}_{\texttt{out}_2}$

5.5 Mach-Zehnder Interferometer

The most interesting use of beam splitters is to combine them with mirrors that reflect the incident photon. The configuration shown in Figure 5.3 is called a *Mach-Zehnder* Interferometer [1]. There are two beam splitters labeled BS_1 and BS_2. The grey objects shown are mirrors. The photon is shown as a wavy line. The photon incident at BS_1 is split in the manner we have described above, where each beam splitter is working according to the matrix $\frac{1}{\sqrt{2}}\begin{pmatrix} -i & 1 \\ -1 & i \end{pmatrix}$, and each mirror produces phase shifts of i over the creation operators. Accordingly, we have the following

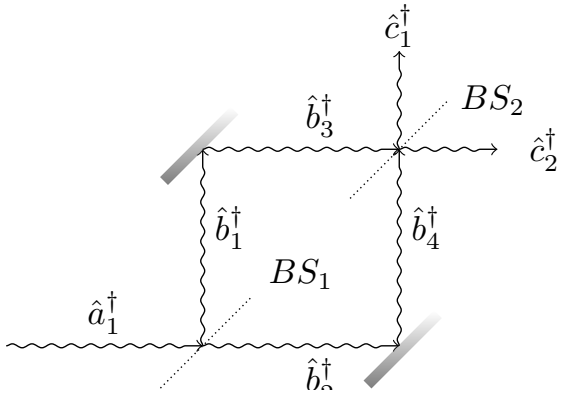

Figure 5.3: Mach-Zehnder Interferometer

transformation between the different creations operators:

$$\mathbf{a}_1^\dagger = \tfrac{1}{\sqrt{2}}(i\mathbf{b}_1^\dagger + \mathbf{b}_2^\dagger)$$
$$\mathbf{b}_1^\dagger = i\mathbf{b}_3^\dagger$$
$$\mathbf{b}_2^\dagger = i\mathbf{b}_4^\dagger$$
$$\mathbf{b}_3^\dagger = \tfrac{1}{\sqrt{2}}(i\mathbf{c}_1^\dagger + \mathbf{c}_2^\dagger)$$
$$\mathbf{b}_4^\dagger = \tfrac{1}{\sqrt{2}}(\mathbf{c}_1^\dagger + i\mathbf{c}_2^\dagger)$$

Given that only one photon incidents at the input mode $\mathbf{a}_1^\dagger$ (see Figure 5.3), then the

state of the input modes is $\left|1\right\rangle \otimes \left|0\right\rangle$ (recall that $\otimes$ is the tensor product of quantum states). According to Equation (2.16), this is equal to $\mathbf{a}_1^\dagger \otimes I(\left|0\right\rangle \otimes \left|0\right\rangle)$. Carrying out the above transformations of the field operators all the way to the end, the output modes state becomes equal to $i\mathbf{c}_1^\dagger \otimes I(\left|0\right\rangle \otimes \left|0\right\rangle)$, i.e., the photon will leave from the vertical port of BS_2 (see Figure 5.3). In the following, we see how to formally prove this result along with the formal definition of the Mach-Zehnder interferometer.

Before we present the theorem that verifies the above result, we have to define the notion of the mirror, similar to what we have for the beam splitters:

Definition 5.9.

$\texttt{mirror}(\texttt{ten}, \texttt{i1}, \texttt{m1}, \texttt{o1}, \texttt{m2}) \Leftrightarrow$

$\quad \texttt{pos ten}(\texttt{cr i1})\ \texttt{m1} = \texttt{i}\ \%\ \texttt{pos ten}\ (\texttt{cr o1})\ \texttt{m2}$

The following theorem shows the formal structure of the above circuit, and proves that if we receive a photon at the horizontal input of the interferometer, then it will leave at the vertical output of the interferometer:

Theorem 5.9.

$\forall \texttt{a b d}.$

$\quad \texttt{is\_tensor ten} \wedge$

1 $\texttt{is\_beam\_splitter}\ (-\sqrt{\tfrac{1}{2}} * \texttt{ii}, \sqrt{\tfrac{1}{2}}), -\sqrt{\tfrac{1}{2}}, \sqrt{\tfrac{1}{2}} * \texttt{ii},$

$\quad \texttt{ten}, \texttt{a\$1}, 1, \texttt{a\$2}, 2, \texttt{b\$1}, 1, \texttt{b\$2}, 2) \wedge$

2 $\texttt{mirror}(\texttt{ten}, \texttt{b\$1}, 1, \texttt{b\$3}, 1) \wedge \texttt{mirror}(\texttt{ten}, \texttt{b\$2}, 2, \texttt{b\$4}, 2) \wedge$

3 $\texttt{is\_beam\_splitter}\ (-\sqrt{\tfrac{1}{2}} * \texttt{ii}, \sqrt{\tfrac{1}{2}}), -\sqrt{\tfrac{1}{2}}, \sqrt{\tfrac{1}{2}} * \texttt{ii},$

$\quad \texttt{ten}, \texttt{b\$3}, 1, \texttt{b\$4}, 2, \texttt{c\$1}, 1, \texttt{c\$2}, 2)$

4 $\texttt{tensor 2}\ (\texttt{lambda i. if i} = 1\ \texttt{then fock}\ (\texttt{a\$1})\ 1\ \texttt{else vac}) =$

5 $\quad \texttt{ii}\%\ \texttt{tensor 2}\ (\texttt{lambda i. ifi} = 1\ \texttt{then fock}\ (\texttt{c\$1})1\ \texttt{else vac})$

Lines (1-4) provide the structure of the circuit in Figure 5.3 with the same modes

naming. Line 4 describes the input modes, where we have one photon at mode $\mathbf{a}_1^\dagger$ and nothing elsewhere. Line 5 provides the corresponding output modes, where we obtain one photon at mode $\mathbf{c}_1^\dagger$ and nothing elsewhere.

Now, we will move to a more complex circuit, where we will build the formal model of another quantum gate using beam splitters, and reason about it.

5.6 Controlled NOT Gate

In this section, we tackle the formalization of a Controlled NOT (CNOT) gate [35], but using single-photon technology (recall that we build the flip gate using the coherent light technology). A CNOT gate is a two inputs/two outputs gate, namely *control* and *target* signals. The gate semantic is to invert the target bit whenever the control bit is equal to one, and nothing changes as long as the control bit is equal to zero. The control bit is always transmitted as is. In other words: if the possible input is $|\psi_i\rangle = \alpha|00\rangle+\beta|01\rangle+\gamma|10\rangle+\eta|11\rangle$ then the output is $|\psi_o\rangle = \alpha|00\rangle+\beta|01\rangle+\gamma|11\rangle+\eta|10\rangle$.

In quantum optics, this gate can be implemented using five beam splitters [67], as given in Figure 5.4, where each of the control and target qbits is represented using two optical beams, and each of the beam splitters follows the matrix $\begin{pmatrix} \sqrt{\eta} & \sqrt{1-\eta} \\ \sqrt{1-\eta} & -\sqrt{\eta} \end{pmatrix}$. For BS4 and BS5, η is equal to $\frac{1}{2}$, and for the rest it is equal to $\frac{1}{3}$. The encoding of four such beams is as follows: applying a single photon to c_0 is equivalent to setting the control bit to zero, and applying the photon to c_1 is equivalent to setting the control bit to one (the same rule applies for the target bit). In Figure 5.4, *vac* refers to the vacuum state, i.e., we do not apply any photons at these ports. For the output modes, v_0 and v_6 are dummy signals and do not have any semantic.

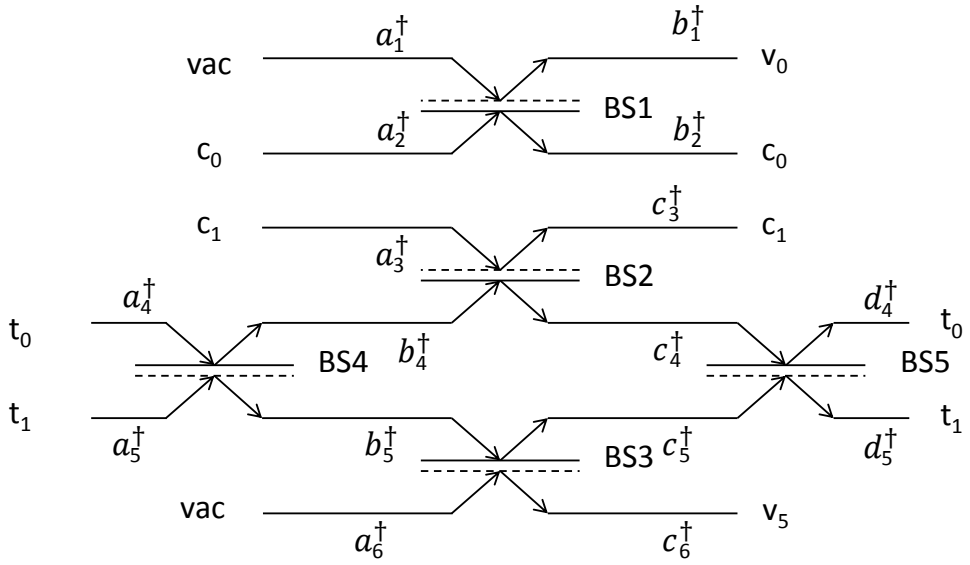

Figure 5.4: Optical Quantum Controlled NOT Gate

Now the formal definition of such a circuit is included in the following theorem:

Theorem 5.10.

$\forall$a b c d.

is_tensor ten $\wedge$

1 is_beam_splitter $(\sqrt{\frac{1}{3}}, \sqrt{\frac{2}{3}}, \sqrt{\frac{2}{3}}, -\sqrt{\frac{1}{3}}, \texttt{ten}, \texttt{a\$2}, 2, \texttt{a\$1}, 1, \texttt{b\$2}, 2, \texttt{b\$1}, 1) \wedge$

2 is_beam_splitter$(\sqrt{\frac{1}{2}}, \sqrt{\frac{1}{2}}, \sqrt{\frac{1}{2}}, -\sqrt{\frac{1}{2}}, \texttt{ten}, \texttt{a\$4}, 4, \texttt{a\$5}, 5, \texttt{b\$4}, 4, \texttt{b\$5}, 5) \wedge$

3 is_beam_splitter$(\sqrt{\frac{1}{3}}, \sqrt{\frac{2}{3}}, \sqrt{\frac{2}{3}}, -\sqrt{\frac{1}{3}}, \texttt{ten}, \texttt{b\$4}, 4, \texttt{a\$3}, 3, \texttt{c\$4}, 4, \texttt{c\$3}, 3) \wedge$

4 is_beam_splitter$(\sqrt{\frac{1}{3}}, \sqrt{\frac{2}{3}}, \sqrt{\frac{2}{3}}, -\sqrt{\frac{1}{3}}, \texttt{ten}, \texttt{b\$5}, 5, \texttt{a\$6}, 6, \texttt{c\$5}, 5, \texttt{c\$6}, 6) \wedge$

5 is_beam_splitter$(\sqrt{\frac{1}{2}}, \sqrt{\frac{1}{2}}, \sqrt{\frac{1}{2}}, -\sqrt{\frac{1}{2}}, \texttt{ten}, \texttt{c\$4}, 4, \texttt{c\$5}, 5, \texttt{d\$4}, 4, \texttt{d\$5}, 5) \Rightarrow$

6 $|010100\rangle = \frac{1}{3} * (\underline{\underline{|010100\rangle}} + \sqrt{2} * |101000\rangle$

7 $+ \sqrt{2} * |100001\rangle + |011000\rangle + |010001\rangle + \sqrt{2} * |100100\rangle)$

Lines (1-5) represent the formal structure of the CNOT gate in Figure 5.4. Note that we used the bra-ket notation [23] in the formal theorem for simplicity, but in the actual code all states are written in the same form as in the Mach-Zehnder example

(see Theorem 5.9). The order of the output bits, on the right hand side of Lines 6 and 7, is $v_0, c_0, c_1, t_0, t_1, v_5$.

According to [67], the output of the circuit in Figure 5.4 is not exactly as desired: As one can notice from Lines 6 and 7, in case the control bit is equal to zero (i.e., $c_0 = 1$ and $c_1 = 1$) and the target bit is equal to zero (i.e., $t_0 = 1$ and $t_1 = 1$). The result on the right hand side contains many possibilities of different probabilities, among them the required (underlined) one with probability $(\frac{1}{3})^2$ (the other unwanted possibilities can be removed by a physics process called coincidence basis [67]). Recall that for a mixed quantum state $\sum |c_i| * |\psi\rangle_i$ that $\sum |c_i|^2$ is equal to one, thus $|c_i|^2$, not $|c_i|$, represents the probability of yielding the state $|\psi\rangle_i$.

We also verify the case where the control gate is equal to zero and the target is equal to one. The result was compatible with the one presented in [67]. Similarly, we verified the case of the control is equal to one. For example in the case of $|001100\rangle$, the following theorem shows the result:

Theorem 5.11.

$$\texttt{tensor}\ |001100\rangle = \tfrac{1}{3} * (|001010\rangle - \sqrt{2} * |002000\rangle - |001001\rangle +$$
$$\sqrt{2} * |000200\rangle + |000101\rangle + |000110\rangle + |000011\rangle)$$

This interesting result concludes our formalization by showing the effectiveness of formal methods.

5.7 Discussion

There are a number of lessons to highlight out of the formalizations presented in this chapter: At the application level, the amount of efforts spent (time and lines of

HOL script) is relatively short, in comparison with the formalization of the theoretical foundations discussed in Chapters 3 and 4. This investment worthwhile as it eases the proof efforts of potential end users, typically verification engineers. Even though all the reported results are similar to the paper-and-pencil analysis (in terms of final conclusion), we have gained more information about the circuits, in particular the full list of assumptions that should be satisfied in order to the circuit to behave correctly. Note that such assumptions are commonly missing in most of the physics books. The operators (e.g., mirrors and shifters) boundness is one such an important assumption that we determined in our work.

The formal analysis of the CNOT gate and Mach-Zehender optical circuits would not have been possible without the development of the following tactic: `MULTI_MODE_DECOMPOSE` which is responsible for passing the creator operator in/out to/from the different modes. As its name suggests, this tactic acts like decomposing multi-modes to many single modes that can be dealt with using the single-mode theorems. The key lemma, on which this tactic is built, is:

Theorem 5.12.

$\forall$`p q f x`.(`p x` $\Rightarrow$ `f x = q`) $\Rightarrow$ (`if p x then q else (f x)`) = `f x`

This lemma typically reduces multi-mode to single-mode, whenever all possible conditions (in the `if` statement) reduce to the same predicate.

Besides above tactic, we have developed `CFUN_FLATTEN`, which takes the whole formula to complex level, at final stage of the proof, to handle some algebraic simplification to finalize the proof. Without these tactics the verification of the Mach-Zehnder interferometer and CNOT gate circuits would be lengthly and complicated.

Through our formalization of the CNOT gate we encountered a problem to generate

the correct answer; the reference [63], we have used for the CNOT circuit has mismatched connections, thanks to theorem proving and our developed tactics, we were able to quickly figure out this problem. Tactics also have another benefit in case one is designing a new circuit from scratch, not like the CNOT case where we knew the final result prior to the formal analysis. In such case, calling the tactics can quickly give a hint about the expected output, then the user can adapt his circuit to achieve the desired result, and recall the tactics (this process is know as the iterative design). The CNOT formalization also showed the scalability of our work since we can tackle circuits with many inputs and many outputs, with a large number of connections and variables. In addition, we were able to perform the analysis in a relatively short time thanks to the developed tactics. Note that the CNOT circuit is working on 6 modes in each step, with the actual number of single modes (including intermediates) equal to 16. For this kind of large circuits, we also gain the trust of the formally produced results, compared to the error-prone paper-and-pencil technique [2].

5.8 Summary

The chapter shows the practicality of the proposed framework in the formal modeling and analysis of optical devices and their applications in quantum computers. We have tackled several applications that vary in terms of size, complexity and functionality. We also covered both single and multi-mode circuits and devices. For instance, we have built the formal models of the coherent light displacer and optical phase shifter, which are basically dependent on the quantum operator exponentiation. Thus, we

[2]It took 2 hours to handel this circuit by paper-and-pencil through several trials. Each time we gave up because of the large number of equations, where we thought to have made a mistake and then we had to start over.

have formalized the notion of exponentiation and proved a number of properties. We then built the flip gate out of these two single-mode optical devices. In this formalization we adopted the coherent light realization of qbits. Next, we addressed the formalization of the beam splitter, an important multi-mode optical element, and proved that it is an energy-lossless device. We then showed the usefulness of this device formalization by providing the formal developments of two circuits that are built solely of the beam splitters, namely the Mach-Zehender interferometer and CNOT gate. The Mach-Zehender interferometer is very common in many quantum computing algorithms. The CNOT gate is another quantum computer gate which involves a large number of optical modes. It operates on 6 modes at each stage, with a total number of 16 modes. In this gate, we adopted the single-photon technology. In total, this library of optical circuits and components amounts to 1000 lines of HOL Light scripts, including 25 theorems and 10 definitions. In addition, we developed two tactics that eased the analysis of the CNOT gate and Mach-Zehender interferometer.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

Quantum optics explores new phenomena and properties of light as a stream of particles called photons. This concept allows a better use of existing optical devices, e.g., beam splitters, and the invention of totally new quantum devices, e.g., single photon devices. These new discoveries are expected to lead to breakthroughs in different fields, especially in quantum information theory. However, the new theory adds complexity to the systems models, which in turn makes their analysis process more complex. The analysis of quantum mechanics systems represents a critical issue. Available techniques, such as simulation in optical laboratories; paper-and-pencil analysis; numerical methods; and computer algebra systems, suffer from a number of problems, including safety, cost, low-expressiveness and soundness. Applying formal methods, more specifically theorem proving, in the area of quantum optics and its application, in particular quantum computing, seems promising since it can deal with some problems that traditional techniques face.

In this multidisciplinary work, we have built a formal analysis framework for quantum optics and its applications. The work provided the HOL formalization of different aspects. The whole framework amounts to about 5000 lines of HOL code with 600 theorems.

From a mathematics perspective, we have formally developed complex-valued function spaces theory, where we have implemented the linear transformation over such linear spaces, and extended such spaces to inner product ones, where quantum states reside. In addition, we have developed some interesting operators, e.g., self-adjoint and Hermitian operators. Moreover, we have tackled a number of functional analysis concepts, namely limit and infinite summation over complex-valued functions.

From a physics perspective, we have covered the formalization of crucial notions of quantum optics. Firstly, we built some general quantum mechanics rules, in particular, we defined the concepts of quantum states, operators. We then customized these rules for optical beams, where we formalized single-mode fields which mimic the single-input/single-output optical systems. The fock states and coherent are then implemented and their relation with the photon number operator was proved. Finally, we generalize our work by formalizing multi-mode fields in order to deal with multi-input/multi-output optical systems.

From an engineering perspective, we showed the practicality of the proposed framework in the formal modeling and analysis of optical devices and their applications in quantum computing. We have tackled different concrete applications, namely the formal models of the coherent light displacer and optical phase shifter. We then built out of these two single-mode optical devices the flip gate. Next, we addressed the formalization of the beam splitter, an important multi-mode optical element. We then utilize this device in the formal developments the Mach-Zehender interferometer and

CNOT gate. In addition, we developed a number of tactics, which eased the formal verification of the applications, such as `MULTI_MODE_DECOMPOSE` which automatically decomposes multi-modes to many single modes.

From a theorem proving perspective, we developed five tactics: `CFUN_ARITH_TAC` and `COP_ARITH_TAC` that are responsible for proving simple arithmetic equational theorems of variables of types `cfun` and `cop`; `LINEARITY_TAC` proves the linearity of the `cop` operator according to `is_linear_cop`, and similarly `SELF_ADJOINT_TAC`, which proves the self-adjointness of the `cop` operator; `REAL_TAC` proves, for a given variable of complex type, that it is a real number, i.e., its imaginary part is equal to zero. In some cases, such tactics reduced our code from more than 300 lines of proof script to around 50. All developed tactics and the HOL Light codes and proof scripts are available at [55].

On the other hand, there were a number of difficulties and challenges in this work. At the beginning, we experienced a problem with finding one clear definition for many quantum concepts. Physics books present the same ideas from different perspectives and each considers some implicit assumptions and approximation, e.g., the idea of quantum space basis (or let us call them span set) and the fact that they might or might not reside in the space. To deal with this problem, we focused our axiomatic definitions on the common ground of the different physics resources available, which was not an easy task. Another problem is the missing of many mathematical assumptions in the formulas available in the reference physics books, which made the formal proof difficult because it adds to the formalization complexity the effort of figuring out these assumptions, including wasted efforts trying to prove things without the respective assumptions. Repetitive proofs steps is another problem which we tackled by developing a number of tactics. The size of large objects, w.r.t the number of the

constructing variables, e.g., the quantum system object `qsys` and single mode object `sm`, raised problems relating to how to access these variables in theorems and definitions: either we define a getter function [1] for each variable (which makes theorems readable but full with `let` statements), or provide as many variables as the number of objects we are dealing with (which makes theorems smaller but not readable).

6.2 Future Work

Quantum theory is represented in many books with different levels of abstraction, with each level serving a specific kind of application. We believe that we presented in this work a flexible formalization that can move among different levels of abstraction: from high abstraction of quantum computing where only two quantum states are of interest, or a mid-level of abstraction like the Dirac abstraction of quantum mechanics, to very concrete implementation where the Lebesgue integral is considered to evaluate the probability distribution in an optics beam. Accordingly, we expect the future extension of this work will take three directions, which complement each other.

Direction 1: The formalization of quantum computers regardless of the realization technology. This is typically the highest level of abstraction, where only two quantum states are of interest. In this level, one can focus more on quantum programming language formalization, and prove their soundness. In addition, this direction would include the verification of quantum algorithms. The major part of this work would be dedicated to the analysis of quantum cryptography protocols which are proved to be more secure and unbreakable against the classical ones.

[1] a getter function of a variable receives the whole object and returns that variable, e.g., `get_v (x,y,z,v) = v`

Direction 2: The formalization of detection theory. An important step that takes place at the end of any quantum circuit computation is the state measurement. In the optics context, this is typically implemented by detecting the photons in an optical beam. The formalization of detection theory is directly related to the framework in this thesis where the formalization of Lebesuge integration will be very helpful in the development of this theory. Then, it can be applied to different optical states, in particular coherent states and squeezed states, where we can prove the high accuracy measurement of such states. In the same direction, one could go further and implement the *Winger functions* [48] (also based on Lebesgue integral). This one is helpful in the analysis of detectors and other complicated optical devices, e.g., parametric amplifiers.

Direction 3: Building fully automated tools for the analysis of quantum gates based on single photon technology. The work we have for the CNOT gate can be extended to other similar universal gates. Thus, it will be valid for any circuit that is built out of those gates. There is a high potential for automating this work: we expect a tool that has a circuit structure as an input, and then generates the formal behavior analysis as an output. It is also possible to apply the same idea to other quantum computer realization technologies, e.g., for coherent states, squeezed states, ion traps, etc. An initial proposal of this tool is presented in [4].

Apart from the quantum theory, the developed complex-valued functions infinite/finite subspaces theory has a wide range of applications in mathematics and engineering, for instance in the area of control theory, where formal methods can have great values due to the safety-critical nature of these kinds of systems in, e.g., aerospace or robotics surgery. The stability of such systems are at stake, and there are many analytical

methods to deal with stability, e.g., Lyapunov function [46] and L^2 input/output stability [28]. The core mathematics of such techniques are the L^2 space, where the input and output signals are modeled as functions of L^2. Whenever the output of a system is not described as an L^2 function then it means that the output is not integrable, i.e., not bounded, and hence the system is not stable.

As another interesting application, decision making is a crucial process that takes place every day in our lives. It becomes more important for the industry where numerous data are available and the principles seek the best decision based on them. This problem is well known in mathematics as the optimization problem [10]. In optimization analysis, all available information are modeled as integrable functions. The more complex the function is the more accurate the decision maker model. The best mathematical tool to describe such optimization functions are the orthogonal independent basis of a L^2 space [54], then we search for a function that is expressed in terms of such orthogonals, and achieves the best maximization or minimization of a certain quantity. Another method is called Pseudo inverse operators. Such operators mimic a certain decision, and the idea is to find the best state (a L^2 function) to which we apply that decision and which yields the best revenue, for example.

The applications are not limited to engineering domains, as complex-valued functions have many applications in mathematics, in particular in functional analysis. A know application in mathematics is Fourier Analysis [72], where fourier transformation is defined as a linear bounded operator over the L^2, and functions subject to transformation are members of the L^2 space.

Bibliography

[1] G. S. Agarwal. *Quantum Optics*. Cambridge University Press, 2012.

[2] E. Ardeshir-Larijani, S. J. Gay, and R. Nagarajan. Equivalence Checking of Quantum Protocols. In *Tools and Algorithms for the Construction and Analysis of Systems*, LNCS, pages 478–492. Springer, 2013.

[3] C. Baier and J. P. Katoen. *Principles of Model Checking*. The MIT Press, 2008.

[4] S. M. Beillahi, M. Y. Mahmoud, and S. Tahar. A Tool for the Formal Verification of Quantum Optical Computing Systems. In *Automated Reasoning Workshop*, pages 25–26, 2015.

[5] J. Blank, P. Exner, and M. Havlíek. *Hilbert Space Operators in Quantum Physics*. Theoretical and Mathematical Physics. AIP Press, 2008.

[6] G. Brassard, C. Crepeau, R. Jozsa, and D. Langlois. A Quantum Bit Commitment Scheme Provably Unbreakable by Both Parties. In *IEEE Annual Symposium on Foundations of Computer Science*, pages 362–371, 1993.

[7] C. E. Brown. *Automated Reasoning in Higher-order Logic: Set Comprehension and Extensionality in Church's Type Theory*. College Publications, 2007.

[8] J. Clarke and F. K. Wilhelm. Superconducting Quantum Bits. *Nature*, 453:1031–1042, 2008.

[9] G. M. D'Ariano, M. G. A. Paris, and M. F. Sacchi. On the Parametric Approximation in Quantum Optics. *Nuovo Cimento B*, 114(quant-ph/9902013):339–354, 1999.

[10] K. Deb. *Optimization for Engineering Design: Algorithms and Examples*. PHI Learning, 2012.

[11] W. Demtrèoder. *Atoms, Molecules and Photons: An Introduction to Atomic-, Molecular- and Quantum Physics*. Graduate texts in physics. Springer, 2010.

[12] P. A. M. Dirac. The Fundamental Equations of Quantum Mechanics. *The Royal Society A: Mathematical, Physical and Engineering Sciences*, 109(752):642–653, 1925.

[13] N. Endou. Algebra of Complex Vector Valued Functions. *Formalized Mathematics*, 12(3):397–401, 2004.

[14] J. M. Feagin. *Quantum Methods with Mathematica*. Springer, 2002.

[15] Y. Feng, E. Hahn, A. Turrini, and L. Zhang. QPMC: A Model Checker for Quantum Programs and Protocols. In *Formal Methods*, volume 9109 of *LNCS*, pages 265–272. Springer, 2015.

[16] R. Feynman. Simulating Physics with Computers. *International Journal of Theoretical Physics*, 21:467–488, 1982. 10.1007/BF02650179.

[17] S. Franke-Arnold, S. J. Gay, and I. V. Puthoor. Quantum Process Calculus for Linear Optical Quantum Computing. In *Reversible Computation*, volume 7948 of *LNCS*, pages 234–246. Springer, 2013.

[18] S. Fujita, K. Itō, and S.V. Godoy. *Quantum Theory of Conducting Matter: Superconductivity*. Springer, 2009.

[19] S. J. Gay and R. Nagarajan. Communicating Quantum Processes. In *ACM SIGPLAN Notices*, volume 40, pages 145–157. ACM, 2005.

[20] S. J. Gay, R. Nagarajan, and N. Papanikolaou. QMC: A Model Checker for Quantum Systems. In *Computer Aided Verification*, volume 5123 of *LNCS*, pages 543–547. Springer, 2008.

[21] R. Gerritsma, G. Kirchmair, F. Zähringer, E. Solano, R. Blatt, and C. F. Roos. Quantum Simulation of the Dirac Equation. *Nature*, 463(7277):68–71, 2010.

[22] C. Gerry and P. Knight. *Introductory Quantum Optics*. Cambridge University Press, 2005.

[23] D. J. Griffiths. *Introduction to Quantum Mechanics*. Pearson Prentice Hall, 2005.

[24] H. Haeffner, C.F. Roos, and R. Blatt. Quantum Computing with Trapped Ions. *Physics Reports*, 469(4):155 – 203, 2008.

[25] P. Haensel, A.Y. Potekhin, and D. G. Yakovlev. *Neutron Stars: Equation of State and Structure*. Astrophysics and Space Science Library. Springer, 2007.

[26] B. Hall. *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction*. Graduate Texts in Mathematics. Springer, 2003.

[27] J. Harisson. HOL-Light Revision r200, `https://code.google.com/p/hol-light/source/detail?r=200`, October 2014.

[28] C. J. Harris and J. M. E. Valenca. *The Stability of Input-Output Dynamical Systems*. Mathematics in Science and Engineering. Elsevier Science, 1983.

[29] J. Harrison. HOL Light: A Tutorial Introduction. In *Formal Methods in Computer-Aided Design*, volume 1166 of LNCS, pages 265–269. Springer, 1996.

[30] J. Harrison. The HOL Light Theory of Euclidean Space. *Journal of Automated Reasoning*, 50(2):173–190, 2013.

[31] O. Hasan and S. Tahar. Formal verification methods. *Encyclopedia of Information Science and Technology, IGI Global Pub.*, pages 7162–7170, 2015.

[32] H. Herencia-Zapana, R. Jobredeaux, S. Owre, P. Garoche, E. Feron, G. Perez, and P. Ascariz. PVS Linear Algebra Libraries for Verification of Control Software Algorithms in C/ACSL. In *NASA Formal Methods*, volume 7226 of *LNCS*, pages 147–161. Springer, 2012.

[33] A. J. G. Hey and P. Walters. *The New Quantum Universe*. Cambridge University Press, 2003.

[34] A. E. Hirst. *Vectors in 2 Or 3 Dimensions*. Modular Mathematics Series. Arnold, 1995.

[35] M. Hirvensalo. *Quantum Computing*. Natural Computing Series. Springer, 2004.

[36] M. Horbatsch. *Quantum Mechanics using Maple*. Springer, 1995.

[37] Institute for Quantum Science and Technology at the University of Calgary, Introduction to an Optical Lab, `http://old.rqc.ru/quantech/memo.php`.

[38] T. Jennewein, M. Barbieri, and A. G. White. Single-Photon Device Requirements for Operating Linear Optics Quantum Computing Outside the Post-Selection Basis. *Journal of Modern Optics*, 58(3-4):276–287, 2011.

[39] M. Kelbert and Y. Suhov. *Information Theory and Coding by Example*. Cambridge University Press, 2013.

[40] S. Khan-Afshar, V. Aravantinos, O. Hasan, and S. Tahar. Formalization of Complex Vectors in Higher-Order Logic. In *Intelligent Computer Mathematics*, volume 8543 of *LNCS*, pages 123–137. Springer, 2014.

[41] S. Khan-Afshar, O. Hasan, and S. Tahar. Formal Analysis of Electromagnetic Optics. In *Novel Optical Systems Design and Optimization XVII*, volume 9193 of SPIE, pages 91930A–91930A. Proceedings of SPIE, 2014.

[42] S. Khan-Afshar, U. Siddique, M. Y. Mahmoud, V. Aravantinos, O. Seddiki, O. Hasan, and S. Tahar. Formal Analysis of Optical Systems. *Mathematics in Computer Science*, 8(1):39–70, 2014.

[43] H. J. Kimble, M. Dagenais, and L. Mandel. Photon Antibunching in Resonance Fluorescence. *Physical Review Letter*, 39:691–695, 1977.

[44] A. N. Kolmogorov, S. V. Fomin, and S. V. Fomin. *Elements of the Theory of Functions and Functional Analysis*. Number v. 2 in Dover books on mathematics. Dover, 1999.

[45] T. D. Ladd, F. Jelezko, R. Laflamme, Y. Nakamura, C. Monroe, and J. L. O'Brien. Quantum Computers. *Nature*, 464:45–53, 2010.

[46] V. Lakshmikantham, V.M. Matrosov, and S. Sivasundaram. *Vector Lyapunov Functions and Stability Analysis of Nonlinear Systems*. Mathematics and Its Applications. Springer, 1991.

[47] U. Leonhardt. Quantum Physics of Simple Optical Instruments. *Reports on Progress in Physics*, 66(7):1207, 2003.

[48] U. Leonhardt. *Essential Quantum Optics: From Quantum Measurements to Black Holes*. Cambridge University Press, 2010.

[49] Y. Li, D. E. Browne, L. C. Kwek, R. Raussendorf, and T. Wei. Thermal States as Universal Resources for Quantum Computation with Always-on Interactions. *Physical Review Letter*, 107:060501, Aug 2011.

[50] S. J. Lomonaco. *Quantum Computation: A Grand Mathematical Challenge for the Twenty-first Century and the Millennium*. American Mathematical Society, 2002.

[51] D. Loss and D. P. DiVincenzo. Quantum Computation with Quantum Dots. *Physical Review A*, 57:120–126, 1998.

[52] B. Lounis and M. Orrit. Single-Photon Sources. *Reports on Progress in Physics*, 68(5):1129, 2005.

[53] D. G. Lucarelli and T.J. Tarn. Holonomic Quantum Computation with Squeezed Coherent States. In *IEEE Decision and Control Conference*, volume 1, pages 452–455, 2002.

[54] D. G. Luenberger. *Optimization by Vector Space Methods*. Professional Series. Wiley, 1997.

[55] M. Y. Mahmoud. Formal Analysis of Quantum Optics: Project Web Page, `http://hvg.ece.concordia.ca/projects/optics/quantumoptics.htm`. 2015.

[56] L. Mandel and E. Wolf. *Optical Coherence and Quantum Optics*. Cambridge University Press, 1995.

[57] The Coq development team. *The Coq Proof Assistant Reference Manual*. LogiCal Project, 2004. Version 8.0.

[58] MetaMath-InnerProduct, http://us.metamath.org/mpegif/mmtheorems168.html#mm16716s, 2009.

[59] R. Milner. *Communicating and Mobile Systems: The Pi Calculus*. Cambridge University Press, 1999.

[60] P. Milonni and M. M. Nieto. Coherent States. In *Compendium of Quantum Physics*, pages 106–108. Springer, 2009.

[61] M. A. Nielsen and I.L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

[62] T. Nipkow, L.C. Paulson, and M. Wenzel. *Isabelle/HOL: A Proof Assistant for Higher-Order Logic*, volume 2283 of LNCS. Springer, 2002.

[63] J. L. O'Brien, G. J. Pryde, A. G. White, T. C. Ralph, and D. Branning. Demonstration of an All-Optical Quantum Controlled-NOT Gate. *Nature*, 426(6964):264–267, 2003.

[64] S. Owre, J. M. Rushby, and N. Shankar. PVS: A Prototype Verification System. In *Automated Deduction*, volume 607 of LNCS, pages 748–752. Springer, 1992.

[65] R. Pashley and M. Karaman. *Applied Colloid and Surface Chemistry*. Wiley InterScience online books. Wiley, 2005.

[66] T. C. Ralph, A. Gilchrist, G. J. Milburn, W. J. Munro, and S. Glancy. Quantum Computation with Optical Coherent States. *Physical Review A*, 68:042319, Oct 2003.

[67] T. C. Ralph, N. K. Langford, T. B. Bell, and A. G. White. Linear Optical Controlled-NOT Gate in the Coincidence Basis. *Physical Review A*, 65:062324, Jun 2002.

[68] F. Schwabl. *Quantum Mechanics*. Springer, 2007.

[69] U. Siddique, Vincent A., and S. Tahar. On the Formal Analysis of Geometrical Optics in HOL. In *Automated Deduction in Geometry*, LNCS, pages 161–180. Springer, 2013.

[70] K. Slind and M. Norrish. A Brief Overview of HOL4. In *Theorem Proving in Higher Order Logics*, volume 5170 of LNCS, pages 28–32. Springer, 2008.

[71] R. M. Smullyan. *First Order Logic*. Dover Publications, 1995.

[72] E. M. Stein and G. L. Weiss. *Introduction to Fourier Analysis on Euclidean Spaces*. Mathematical Series. Princeton University Press, 1971.

[73] J. Stein. LinAlg: A Development of Some Preliminary Linear Algebra, `http://www.lix.polytechnique.fr/coq/pylons/coq/pylons/contribs/view/LinAlg/v8.4`, 2012.

[74] S. M. Tan. A Computational Toolbox for Quantum and Atomic Optics. *Journal of Optics B: Quantum and Semiclassical Optics*, 1(4):424, 1999.

[75] H. Vahlbruch, M. Mehmet, S. Chelkowski, B. Hage, A. Franzen, N. Lastzka, S. Goßler, K. Danzmann, and R. Schnabel. Observation of Squeezed Light with 10-db Quantum-noise Reduction. *Physical Review Letter*, 100:033602, Jan 2008.

[76] J. Y. Vaishnav and C. W. Clark. Observing Zitterbewegung with Ultracold Atoms. *Physical Review Letters*, 100(15):153002, 2008.

[77] F. Wiedijk. Comparing mathematical provers. In *Mathematical Knowledge Management*, volume 2594 of *LNCS*, pages 188–202. Springer, 2003.

[78] S. Yamashita and I. L. Markov. Fast Equivalence-Checking for Quantum Circuits. In *IEEE/ACM International Symposium on Nanoscale Architectures*, pages 23–28. IEEE Press, 2010.

Biography

Education

- **Concordia University**: Montreal, Quebec, Canada

  Ph.D. candidate, Dept. of Electrical & Computer Engineering, (November 2012 - present)

- **Alexandria University**: Alexandria, Egypt

  M.A.Sc. of Engineering Mathematics, (September 2008 - July 2011)

- **Alexandria University**: Alexandria, Egypt

  B.Sc. of Computer and Systems Engineering, (September 2003 - July 2008)

Awards

- Concordia Graduate Accelerator Award, Concordia University, Canada (2015).
- Concordia Graduate Fellowship, Concordia University, Canada (2011-2014).
- Concordia Partial Tuition Award, Concordia University, Canada (2011-2014).

- Faculty of Engineering Graduation Projects Competition (second place), Alexandria University, Egypt (2008).
- Alexandria University Award of Excellence, Alexandria University, Egypt (2003-2008).
- Alexandria Secondary Schools Academic Competition (first place), Alexandria University, Egypt (2003).

Work History

- **Concordia University**: Montreal, Quebec, Canada
  Research Assistant, Dept. of Electrical & Computer Engineering (2011-2015)
  Teaching Assistant, Dept. of Electrical & Computer Engineering (2014-2015)
- **Alexandria University**: Alexandria, Egypt
  Teaching Assistant, Dept. of Engineering Mathematics, (2008 - 2011)
- **Ejada IT Solutions**: Alexandria, Egypt
  Software Engineer, (2008 - 2011)

Publications

- **Journal Papers**
  - **Bio-Jr1** S. K. Afshar, U. Siddique, M.Y. Mahmoud, V. Aravantinos, O. Seddiki, O. Hasan and S. Tahar: Formal Analysis of Optical Systems; *Mathematics in Computer Science*, Vol. 8, No. 1, Springer, May 2014, pp.

39-70.

- **Bio-Jr2** M. Y. Mahmoud, V. Aravantinos and S. Tahar: On the Formalization of Quantum Mechanics in HOL. *Journal of Automated Reasoning*, 26 pages (Under Third Review).

- **Bio-Jr3** M. Y. Mahmoud, S. Tahar: Towards the Formalization of Quantum Computers in HOL. *ACM Transactions of Computer Logic* July 2015, 25 pages (Submitted).

• Refereed Conference Papers

- **Bio-Cf1** M. Y. Mahmoud, P. Panangaden and S. Tahar: On the Formal Verification of Optical Quantum Gates in HOL, In: *Formal Methods for Industrial Critical System* , Lecture Notes in Computer Science Volume 9128, 2015, pp 198-211.

- **Bio-Cf2** S. M. Beillahi, M. Y. Mahmoud, and S. Tahar A Tool for the Formal Verification of Quantum Optical Computing Systems, In: *Automated Reasoning Workshop*, April 2015, pp. 25-26.

- **Bio-Cf3** M. Y. Mahmoud, V. Aravantinos and S. Tahar: Formal Verification of Optical Quantum Flip Gate, In: *Interactive Theorem Proving*, Lecture Notes in Computer Science Volume 8558, 2014, pp 358-373.

- **Bio-Cf4** U. Siddique, M. Y. Mahmoud and S. Tahar: On the Formalization of Z-Transform in HOL, In: *Interactive Theorem Proving*, Lecture Notes in Computer Science Volume 8558, 2014, pp 483-498.

- **Bio-Cf5** M. Y. Mahmoud, and S. Tahar: On the Quantum Formalization of Coherent Light in HOL, In: *NASA Formal Methods*, Lecture Notes in

Computer Science 8430 , Springer Verlag, 2014, pp. 128-142

– **Bio-Cf6** M.Y. Mahmoud, V. Aravantinos, and S. Tahar: Formalization of Infinite Dimension Linear Spaces with Application to Quantum Theory, In: *NASA Formal Methods*, Lecture Notes in Computer Science 7871, Springer, 2013, pp. 413-427.

– **Bio-Cf7** M.Y. Mahmoud, V. Aravantinos and S. Tahar: Towards the Formal Verification of Quantum Optical Systems, In: *International Workshop on Formal Techniques for Safety-Critical Systems*, Kyoto, Japan, November 2012, pp. 147-151.