Logics of Probability for Quantum Computing and Information

Manas K. Patra
School of Computer Science and Engineering
University of New South Wales
Sydney 2052 Australia
manasp@cse.unsw.edu.au

Supervised by
A. Prof. Ron van der Meyden

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy (Computer Science and Engineering)

Dedicated to the memory of Asish Mahapatra.

Acknowledgement

I wish to thank my supervisor Ron van der Meyden for his help and cooperation. I learned to curb my 'quantum jumps' to conclusions and proceed in a thorough and systematic fashion when doing formal reasoning. Although, a computer scientist by training Ron took the trouble of reading up on quantum theory, especially on the difficult conceptual parts. He made valuable contribution to bring this work to its present form.

I wish to thank my wife Sucharita and son Nishant for bearing with me during some difficult periods. My parents and my brothers and sister have given me constant encouragement and support and I am thankful to them. Thanks also to my sister-in-law Lisa, brother-in-law Bidyut and parents-in-law for their help and support.

Many friends and colleagues have been helpful and encouraging. I can only mention a few. Norman, Kevin, Victor, Steven and Arthur from CSE, UNSW have participated in many discussions during the early phase of this work. I wish to thank them all. Abhaya, Igor, Mehmet, Jason, Bernard and Arindam at Macquarie have been especially encouraging and often helpful. I am thankful to them. I wish to thank Kali and Peter (Brooke) for their encouragement.

Finally, I wish to acknowledge my gratitude to an old friend Asish who is no longer with us. His untimely death left us devastated because he was not only a very good friend but a very decent human being. This work is dedicated to his memory.

Manas Patra

Contents

Chapter 1

Overview

This chapter gives an overview of the rest of the work. In this work we present logics for reasoning about finite dimensional quantum systems. Our approach is operational. Our objective is to formalize the basic probabilistic language of practicing quantum physicists. The examples, which express many of the fundamental notions of quantum theory, including superposition, entanglement, separability and state tomography show that the languages developed here are expressive enough. The main applications are to the new and exciting area of quantum computation and information(QCI).

The role of logic in the study of foundations and structure of mathematics is well established over the last century. Since the emergence of computer science and artificial intelligence as scientific disciplines over the last half century logic has assumed a preeminent role. The importance of the study of formal systems or logics may be understood from two perspectives, both having roots in mathematical logic. First, it is often the case that a computer is abstractly modeled as a device for processing symbols(or strings of symbols). The archetypal Turing machine is a case in point. However, any such machine must be programmed with a definite system of rules so that the next move is unambiguous. We are of course restricting ourselves to deterministic machines. A formal system or *theory* in logic may also be thought of as a rule based system for deriving certain well-defined strings(the for-

mulas) as *theorems*. In fact, the formal system approach(Post) [Smu96] is an abstract formulation of the notion of computation it is equivalent to the other approaches (Church, Turing, Kleene...). This approach is also intimately related to the notion of a *proof*, the object of study of a branch of logic known as **proof theory**. The most familiar form of proof theory is the axiomatic approach that underpins most of mathematics. There are other approaches(sequent calculus [Ebb96] and natural deduction [Dal94]. In this work we follow the axiomatic approach. The second perspective on the pre-eminence of logic may be understood from a simple example. Suppose that we have a single lift in a multi-storeyed building. We would like to optimize energy costs of running the lift. We would also require that no user has to wait for unduly long periods. Besides there may be host of safety issues. How do we tackle a host of possible scenarios? How do we formulate our *specifications* so that we may test them against some protocol for running the lift? How do we formulate and analyze the protocol itself? Like any branch of algebra, say, it would be very advantageous to take a symbolic approach so that the state of the lift may be represented in some symbolic way, well defined sets of symbols representing possible abstract states of the lift and users, the action of the user (or/and the lift operator) defined as operations on the symbols representing the state and so on. We are constructing a *formal model* of the system lift+users. Naturally, we cannot expect to incorporate *everything*. For example, the full dynamics of the lift! We are abstracting some idealized scenarios from the complex behaviour. What we keep will depend upon the aspects we are interested in. To reason about the lift+users system in this restricted sense we must of course, have some well-defined correspondence among the various symbols and concrete states and behaviours. That is, we have a *semantics* for our formal system. The study of semantics of logical systems is as old as mathematical logic. This is the main thrust of *model theory*.

The simplistic example above does not do justice to the importance and

power of the formal approach which was motivated by the study of foundations of mathematical reasoning. Since the computer and information revolution the formal approach has been applied with great success to hardware verification, analysis of programmes, protocols to name a few. The present work aims to develop and apply these techniques to Quantum Computing and Quantum Information(QCI). In the following paragraphs I give an overview of each of the subsequent chapters.

Quantum computing overlaps both physics and computing. The present work is addressed to computer scientists and physicists. To provide them with background material for the two disciplines I have incorporated *two* introductory chapters. The first gives an introduction to logic and some of its applications in computer science. I also give a short synopsis of *complexity theory*. We start with the simplest and the most basic of all logics: propositional logic. Next we consider its extensions. First order logic or predicate calculus has a special place in mathematical theories since most mathematical concepts are formulated in the language of first order logic. Moreover, since the logics developed in the later chapters of this work are extensions of first order logic I devote a section to it. This is followed by a review of some *probabilistic* logics. Here the primary focus is on the logics developed by Fagin, Halpern, Meggedio and Tuttle [FHM90, HT93]. The reason is that I share their view that a probabilistic theory must deal with real numbers and polynomial expressions(contrast the linear case in the references cited above) to do justice to the rich concepts in mathematical probability theory-for example, conditional probability and Bayesian inference. It is an often repeated cliché that quantum theory is inherently probabilistic. Therefore in the spirit of the works cited the logics presented here extend a first order theory. However, unlike the classical case it is not a theory of reals that is extended but a theory of complex field considered as an extension(in the algebraic sense) of real (closed)fields. The other logics discussed in the next chapter are modal logics which may be considered as another exten-

sion of propositional logic. I mainly discuss logics which are of importance in compute science. After a brief introduction to general modal logic I give short sketches of temporal, dynamic and epistemic logics. I only indicate some of their applications in various areas of computer science- hardware verification, protocol analysis and programme correctness- citing some of the vast literature for more material. In the next section I discuss the other approaches to formalizing quantum computation and protocols. The most elegant approach among these is the categorical semantics of Abramsky and Coecke [AC04a]. Some other approaches like the dynamic logic approach [BS04], the logics of Mateus and Serandas [MS04b] and their relation to the present work are also discussed. I also give a sketch of so-called *quantum logic* which is quite different from the approaches mentioned. It is the oldest approach [BvN36] toward understanding the formal structure of quantum theory.

The Chapter 3 gives a brief introduction to Quantum Theory. This is addressed to the readers from computer science. I only state the relevant mathematical structure emphasizing the probabilistic aspects. There is very little physics in the chapter as there are many excellent texts on quantum theory. However, I follow the pragmatic approach in the interpretation of the mathematical formalism. This means that the "observable" probabilities are given primary importance. I also discuss composite systems and their tensor product structure. Measurement is an important issue in quantum theory. I discuss only a restricted class of measurements (projective measurements). This is sufficient for our purposes. In the spirit of quantum computation *unitary* operators or matrices are discussed at some length. These are quantum analogues of classical (logic) gates. Note that we deal only with finite dimensional quantum systems which are essentially the ones used for building quantum gates.

The Chapter 4 lays the foundation of the logics developed in this work [MP03b]. As I have mentioned, the latter are all extension of a first order

theory. That first order theory is called $\mathbb{RC}$. It is a theory of algebraically closed fields considered as an algebraic extension of a real closed field. In presenting quantum theory we require real numbers for probabilities and expectation values *and* complex numbers because the probabilities are functions of states which are elements of some *complex* Hilbert space. The theory $\mathbb{RC}$ is dealt with in detail. It is theory with equality which has predicate R for *real* terms, and for such terms there is predicate $<$ which is a linear order. I prove two of its most important properties: quantifier elimination and completeness. I also establish an algorithm for efficient translation of a formula in $\mathbb{RC}$ to a formula in the real closed field such that one is satisfiable if and only if the other is. It is convenient to introduce some defined symbols for square root and complex conjugation since many of the formulas in quantum circuits and protocols involve square roots. The importance of the theory $\mathbb{RC}$ lies in the fact that our strategy for verification and synthesis of quantum circuits is to reduce them to equivalent formulas in $\mathbb{RC}$. There are two advantages in doing this. First we use the properties of $\mathbb{RC}$ in deducing those of the logics developed for quantum circuits and protocols. Second it gives us a concrete algorithm for verification and synthesis of the latter.

In section 4.3 the syntax and semantics of the basic language is presented. We note that in this chapter we deal with quantum systems in a finite dimension n. Thus the language is called $\mathcal{L}_n(P, \mathbf{m})$. The language has three types of distinguished symbols:

1. **Basis symbols.** We use symbols $\mathtt{b}, \mathtt{c}, \mathtt{d}$ sometimes with subscripts to represent (orthonormal) bases in a Hilbert space H_n of dimension n. Associated with each basis symbol $\mathtt{b}$ are n *basis variables* written as $\mathtt{b}_0, \ldots, \mathtt{b}_{n-1}$. The basis variable $\mathtt{b}_i$ is interpreted as the projector on to the subspace generated by the i^{th} vector in the basis corresponding to $\mathtt{b}$. We allow propositional connectives $\vee$ and $\neg$ over basis variables to construct basis formulas. In general, a basis formula corresponds to a subspace in the Hilbert space H_n.

2. **Matrix symbols.** For each pair of bases we have n^2 variables $\{m_{ij}|i,j = 0,\ldots,n-1\}$ of $\mathbb{RC}$. These are interpreted as the $(ij)th$ entry of the unitary matrix connecting the bases corresponding to $\mathtt{b}$ and $\mathtt{c}$.

3. **Probability operator.** For each basis formula $\mathbf{B}$, the expression $P(\mathbf{B})$ is called an atomic probability term. It is interpreted as a real number(more generally as an element of a real closed field). A general term of $\mathcal{L}_n(P,\mathbf{m})$ is obtained by substituting the atomic terms of the form $P(\mathbf{b})$ and $m_{ij}(\mathtt{b},\mathtt{c})$ for variables in a multivariate polynomial in $\mathbb{RC}$. Thus, we admit quite general (nonlinear) expressions in probability and matrix terms. If Φ is such a general term then the atomic formulas of $\mathcal{L}_n(P,\mathbf{m})$ are of the form $\Phi = 0$ or $\Phi > 0$. A general formula is a boolean combination of atomic formulas or quantification over some of its variables (of $\mathbb{RC}$). I emphasize that quantification is allowed over $\mathbb{RC}$ variables only. The interpretation is that a formula like $P(\mathbf{B}) = x$ expresses that the probability of the outcome to be in the subspace corresponding to $\mathbf{B}$ is equal to x.

As stated above the formulas of $\mathcal{L}_n(P,\mathbf{m})$ are interpreted in a complex Hilbert space of dimension n. Actually, they may be interpreted over only n-dimensional vector space with an inner product over some model K of $\mathbb{RC}$. For convenience we call members of K, complex and real numbers(where appropriate). The probability formulas are interpreted in a *state*. A state here means a *pure* state in contrast to the mixed states considered in the next chapter.

Several examples illustrating the use of the logic are presented. Important concepts including superposition and uncertainty principle and state tomography are expressed as formulas in the language. I also write formulas for most single qubit gates. The more complex examples like 2-qubit gates, Grover search algorithm and teleportation are dealt with in the next chapter after the introduction of tensor product.

Next we deal with axiomatization. We call the resulting theory $\mathbf{Ax}_n(P,\mathbf{m})$.

The axiomatization is presented in two parts. In 4.4.2 a fragment of $\mathcal{L}_n(P, \mathbf{m})$ which consists of all formulas without the matrix operator $m_{ij}(\mathsf{b}, \mathsf{c})$. Thus we have only terms containing the probability operator. We call the resulting theory $\mathbf{Ax}_n(P)$. Although, of restricted expressiveness $\mathbf{Ax}_n(P)$ is the closest parallel to the probabilistic logic of Fagin, Halpern *et. al.* mentioned above. It contains only the basis axioms and the probability axioms. Some immediate consequences of the axiomatization are first derived which are useful for reduction algorithms presented later. We have the nontrivial result that any formula of $\mathcal{L}_n(P, \mathbf{m})$ consistent with the probability and the basis axioms is satisfiable. This result is used in proving the completeness of $\mathbf{Ax}_n(P)$ in Theorem 7. Next we show that the decision problem for satisfiability of a formula in $\mathbf{Ax}_n(P)$ is NP-complete. We add next axioms for the transformation matrix operator $m_{ij}(\mathsf{b}, \mathsf{c})$. Essentially these express that transformation relations connecting any two bases in the given Hilbert space form a matrix group, *viz*, the group of unitary matrices. The most complicated axiom is $\mathbf{MP}_k$ which asserts that the probability assignments be consistent with the transformation relations. As we see later this axiom is essential, for there may be probability assignments for different bases and transformation matrices connecting them that are consistent with all other axioms of $\mathbf{Ax}_n(P, \mathbf{m})$ but still the formula is not satisfiable in any quantum state. We prove later that the number of bases k for which we require to check satisfiability is $n^2 - 1$. This is a nontrivial result since the number of possible bases has cardinality $2^{\aleph_0}$ (that of the continuum) and secondly the number $n^2 - 1$ is optimal. As in the case of $\mathcal{L}_n(P)$ I give efficient algorithms for reducing a formula of $\mathcal{L}_n(P, \mathbf{m})$ to an equivalent formula in $\mathbb{RC}$. The equivalence is in the sense of satisfiability. These results serve two purposes. First, they are used to prove the completeness of $\mathcal{L}_n(P, \mathbf{m})$ by reducing to the case of $\mathbb{RC}$. Second, they provide us with concrete methods for settling questions of satisfiability in quantum systems. These algorithms form the basis for the algorithmic verification tools being developed as an extension

to this work. Thirdly, we use the reduction theorems and the known results for complexity in real closed fields to obtain upper bounds on the complexity of the satisfiability problem in finite dimensional quantum systems. The chapter concludes with an alternative language and its semantics for dealing with quantum computation. The language $\mathcal{L}_n(P, \mathbf{m})$ and its extensions in the following chapters are expressive enough to express all known quantum circuits and protocols. But the translation is not "intuitive" in the sense physicists and computer scientists view quantum circuits. Thus a "quantum circuit" like its boolean analogue is function from the input states to the output states. It is usually composed of some basic "gates". That is, we view gates as operations on states. This is the dynamic view- state changing over time- as opposed to the static view where the state remains constant but viewed from changing perspective(the bases!). To facilitate the dynamic view I present another language with new kinds of syntactic operators. Corresponding to a unitary matrix U we have an operator $[U]$ acting on the probability formulas. For many purposes it is sufficient to have only one basis symbol $\mathtt{b}$ and a general formula is obtained by operating probability formulas over $\mathtt{b}$ with operators $[U], [V], \ldots$ -the "gates". We observe some similarities with dynamic logics [Gol92, Har79] which are used to reason about programmes. We see important examples of the formulas in this language in the next chapter. However, the language $\mathcal{L}_n(P, \mathbf{m})$ and its extensions are more expressive and often more convenient to reason with. The equivalence of the two is demonstrated by embedding them in a larger theory. Thus, we may avoid giving the axiomatization for the new languages. The complexity bounds are also same.

The Chapter 5 extends the logics in the preceding chapters by adding new operators for tensor product and measurement [Pat05]. Given two vector spaces S and T of dimension m and n respectively, over the same field one can define the tensor product $S \times T$. It is a vector space of dimension mn. If S and T also have a scalar product defined, making them Hilbert

spaces then there is natural extension of the scalar product to $S \otimes T$. These constructions from linear algebra are briefly sketched in the chapter on quantum theory 3. The important point is, that if a S and T are the state spaces of two quantum systems then $S \otimes T$ is the state space of the *joint* composite system. Even in classical computing we have to consider the states of such composite system. For example the set of states of two bits is the *cartesian* product $B \times B$ where $B = \{0, 1\}$ is the boolean algebra of two elements. Since the set of states of a quantum system must be linear vector space we have the tensor product in this case. As a consequence, besides the states which behave like classical product states we have *entangled* states which are linear combinations of product states. Entanglement is one of the intriguing features of quantum systems and a rich source for powerful practical application. From a logical point of view the problem we are faced with is the following. We start with the languages $\mathcal{L}_m(P, \mathbf{m})$ and $\mathcal{L}_n(P, \mathbf{m})$ in dimension m and n respectively. Suppose we do not restrict the number copies of the corresponding quantum systems. This implies that we take the tensor product of arbitrary copies of H_n and H_m. In general, a formula could have subformulas referring to more than one dimension. This will clearly cause problems in the interpretation. Moreover, we will also have ordering issues since the tensor product is not commutative. The first problem is solved by stipulating that the probability formulas, whose interpretation depends upon the state, refer to a fixed dimension, say n. But the basis formulas over which the probability operators act may be tensor product of bases of lower dimension. Hence, all bases occurring in the formula are required to have dimensions which are factors of n. With these restrictions we define the language $\mathcal{L}_n(P, \mathbf{m}, t, M, S)$ where t is the tensor product symbol, M and S are symbols for two types of measurement.

Section 5.1 gives the syntax and semantics of the logic for the anguage $\mathcal{L}_n(P, \mathbf{m}, t, M, S)$. The t-operator is introduced as the basis terms appearing in the formulas are classified as *irreducible* and product bases. The

t-operator is defined for any two basis terms. We recall that associated to basis symbols in dimension n there are n basis variables. In the case of the product basis the ordering of the basis variables are, by definition, inherited from that of its factors. This is defined unambiguously and is important in subsequent sections. Next, I introduce two operators for measurement: the first one is the standard formalization of projective measurement as introduced by von Neumann [BvN36]. The second is a special kind of measurement called selective measurement. Here the measuring agent selects a particular outcome (if recorded) others being discarded. This operator can be dispensed with by expressing it in terms of other operators. But we retain it because certain formulas appear more intuitive by its use. The semantics of $\mathcal{L}(P, \mathbf{m}, t, M, S)$ is trickier than the language $\mathcal{L}_n(P, \mathbf{m})$ considered in the previous chapter. The reason is, because of the tensor product we may have bases of *different* dimensions in the same formula. Probability terms and formulas with measurement operator are the terms whose semantics depends on the state which is usually assumed to be given in some fixed dimension, possibly unknown. Therefore, by definition all probability terms are in some fixed dimension n and matrix terms are unrestricted. Thus in matrix terms we may have bases in dimensions different from n. They are interpreted accordingly. The terms $t(\mathsf{b}^{n_1}, \mathsf{c}^{n_2})$ is interpreted as the tensor product of the bases corresponding to b^{n_1} and c^{n_2}. Instead of arbitrary vector space we interpret the formulas in $\mathbb{C}^n$ with the standard tensor product. With this stipulation the interpretation of matrix terms is almost identical to that in the Chapter 4. However, the notion of state is broadened to include "mixed" states or density matrices. Informally, a "mixed" state reflects our degree of uncertainty about the quantum state. I explain this concept briefly in Chapter 3. The probability terms are evaluated in mixed states. The reasons for this generalization is explained in 5.1. Next, I give the semantics of measurement operators which have some similarities to Kripke [Gol92] semantics of modal logics.

In section 5.2 some examples of formulas in $\mathcal{L}_n(P, \mathbf{m}, t, M, S)$ are given. These are some of the important gates in quantum circuits. The more complex examples are deferred till the next chapter. Next we present an axiomatization of the for the language $\mathcal{L}_n(P, \mathbf{m}, t, M, S)$. The resulting theory is called $\mathbf{Ax}_n(P, \mathbf{m}, t)$. The new axioms pertain to the tensor operator and the measurement operator. The axioms for the t-operator are somewhat complicated. This is because there is no natural isomorphism relating tensor product of the same spaces but in different ordering. But as we fix our interpretation in $\mathbb{C}^n$, $\mathbf{Ax}_n(P, \mathbf{m}, t)$ it inherits some features typical to these spaces and these are formalized in the axiomatization. As in 4 we prove some important model theoretic and complexity results. We essentially adopt the same strategy as in the preceding chapter to accomplish this. Thus we reduce a formula of $\mathcal{L}_n(P, \mathbf{m}, t, M, S)$ to an equivalent (in the sense of satisfiability) formula of $\mathbb{RC}$. I prove a theorem which gives an efficient algorithm for this reduction. This result is then used to prove completeness and decidability of $\mathbf{Ax}_n(P, \mathbf{m}, t)$. It is also instrumental in deriving upper bounds for the complexity of the decision problem of satisfiability. The algorithm may also be used for higher-level simulation of quantum circuits and protocols.

In Chapter 6 we discuss alternative applications. In particular, I include three of the best known quantum algorithms: teleportation, Grover search [Gro96], and phase estimation (Shore algorithm). I start with an alternative language $\mathcal{L}_n(P, t, M, S, \mathbf{U})$ which which is intuitively better related to quantum circuits and protocols. We have already discussed this in the context of $\mathcal{L}_n(P, \mathbf{m})$. The teleportation and Grover algorithm are written as formulas. It is shown that the Grover algorithm may be written as formulas but also the question of their *existence* in some fixed dimension. The phase estimation algorithm requires introduction of a defined predicate for n^{th} roots of unity. With the use of the reduction algorithm 12 it can be shown that the equivalent $\mathbb{RC}$ formula Grover algorithm formula $\mathbf{G}$ is $O(|\mathbf{G}|^2)$. Hence, as

each of the basic matrix operation in the algorithm can be done in fixed time the classical simulation of $\mathbf{G}$ requires $O(n)$ steps, a well known result. However, the phase estimation algorithm (which is the key step in Shor factorization algorithm) is a different matter. However, it can be verified (classically) if the number of qubits is not too large. The chapter concludes with a discussion on implementation issues.

In the final chapter I make some concluding remarks. I discuss some work regarding knowledge in quantum systems [MP03a] which was not included in this work. The directions of future work and other issues are discussed.

Chapter 2

Introduction

This chapter gives an introduction to several logics connected with the current investigation. Many of these logics will be very familiar to computer scientists but readers from physics background may be unfamiliar with some. The subject of logic is a broad one. From the philosophers of antiquity to the modern revival initiated by mathematicians like Boole, Cantor, Frege, Peano and philosophers like Russel and Wittgenstein the issues and methods of logic span a wide spectrum. However, they are all concerned with the central theme: "what assertions may be reasonably considered to be **true** and how to arrive at such truths". Here we must distinguish between empirical truths that are the concern of experimental sciences and formal truths which are generally relational. This roughly means that given some assertions which are accepted to be true what are we seek/deduce the relations amongst these which are also true. Thus the statement "the orbits of the planets in the solar system are elliptical" is an accepted empirical truth but the assertion "the orbits of the planets are elliptical or they are *not* elliptical" is a logical truth. Here the significant object is the connective "or" which combines two assertions and the negation "not". It does not matter what the individual assertions are and we could replace them by any symbol say, **a** and the statement "**a** or not **a**" is a logical truth. Logic is concerned with the truth of the *relations* among the individual or

atomic statements. Clearly, we need a more precise language than a natural language like English in which phrases like "or", "not", "necessity", "for all" are unambiguously defined. An essential part of any language is the alphabet- the set of symbols- and the formation rules (the grammar) which tell us which strings of alphabet constitute legitimate objects (e.g terms, sentences) of the language. These formation rules must be such that we can always decide in finite number of steps whether a given string is a specific type of object of the language. The language and the distinguished sets of objects constitute the syntax of the logic. We usually reason with some class of syntactic objects which may be assigned truth values. These are called propositions, sentences, formulas, statements etc. in different contexts. Abstractly, we say that we have a distinguished class of strings- which we call formulas- for definiteness and whose formation rules are unambiguously given. The object of a logical theory is the study of these formulas.

There are two aspects to the study of formulas in a logic. The first is purely syntactic or structural. Among the symbols of a language there are some distinguished ones called logical constants. First there are the propositional connectives $\wedge, \vee, \Rightarrow$, and $\neg$. Then there are the quantifiers $\forall$ and $\exists$. Among the set of formulas we define a subset called the axioms. We also have a set of *inference rules.* Further, the axioms are divided into two groups *logical axioms* and *nonlogical axioms.* The logical axioms and *all* inference rules capture our intuitive understanding of the logical constants. All theories have the logical axioms and rules in common. It is in the treatment of the logical axioms and rules that classical approach (the most common approach) and the *intuitionistic* approach differ. First let us briefly review the classical approach (also called formal approach). By definition all axioms are theorems of the logic. Then one uses the rules to derive new theorems from the ones already proved . Any formula obtained by applying these rules is a theorem and the steps in the derivation is a *proof* of the theorem. We note that a proof is an effective procedure in the sense

that given the axioms, rules and the steps in the proof we can verify the purported proof is indeed one. Implicit in this statement is the fact that a proof involves finitary reasoning. In other words, a proof may in principle be verified by purely mechanical means, for example by a Turing machine. Note that in this approach we are not concerned with the *meaning* of the formulas or sentences. We treat them as purely symbolic objects. This branch of logic is called proof theory. The particular approach to proof theory with axioms and proof rules is called the axiomatic method [Chu56]. It is the most prevalent approach in modern mathematics.

However, there are other approaches to proof theory. The closely related systems of natural deduction and sequent calculus were influenced by the ideas of the intuitionistic school [Dru77]. The difference between the intuitionistic and the classical school is in their views on what constitutes a proof. According to the intuitionists a proof must be *constructive* in the following sense. A proof of $A \vee B$ is anything that is either a proof of A or proof of B. In other words, we must have an effective means of demonstrating a proof of A or proof of B. Thus, the formula $A \vee \neg A$ (the law of excluded middle), which is a theorem in classical mathematics, cannot be taken for granted unless we have an effective means of proving A or $\neg A$. For example, if $A(n)$ is the statement that n is a prime then $A(n) \vee \neg A(n)$ is acceptable as a theorem in the theory of natural numbers for given any n we have effective means showing that it is either a prime or not. Similarly, a proof of $\exists x B(x)$ if we have an effective means which will yield an individual n such that there is proof $B(n)$. By this criterion many proofs of classical mathematics are not acceptable. The most well known among them are- the two related systems natural deduction [Dal94] and sequent calculus [Ebb96]. I quote Drummet (cited above) on natural deduction:

> *A natural deduction system* is a formalization of logic in which no formulas are axiomatically assumed as valid, but there are only rules of inference. To compensate for lack of axioms it is

permitted to introduce any formula as hypothesis at any stage.

The inference rules of these proof systems are different for classical and intuitionistic logics. In this work we will be primarily concerned with the axiomatic approach.

The second apect of logic is to give interpretation or meaning to the objects of the language. The language of the logic is called the object language. Clearly, we cannot interpret the object language in itself. The interpretation is an abstract process involving an element of intuition. We may interpret in some metalanguage, for example, a natural language like English. Or we could interpret one object language in another. In mathematical logic the interpretation is done in the language of sets. Once interpreted the sentences of the object language can be given truth values. Evidently, we must demand that the axioms must be true and that the rules of derivation be such that any sentence derived from true sentences must itself be true. This branch of logic is called model theory [Hod97] or semantics. Unlike a proof the truth of a sentence may not be finitely decidable.

The imprecise discussion above is intended to give a general overview of the study of logic. In the following sections we discuss several logics which illustrate the general discussion and also give background to the logics developed in subsequent chapters. I also discuss other approaches to formal reasoning about quantum systems which is the primary concern of this work. The structure of this chapter is as follows. In the first section I discuss the simplest and the most basic of all logics, namely, propositional or boolean logic. A brief overview of the syntax and semantics is given along with an axiomatization. Then I discuss two possible extensions. The first is known as predicate calculus or first-order logic. It is the most important logic for mathematics. Most(but not all) mathematical concepts in mathematical logic can be expressed as sentences in first-order logic. Moreover, the logics presented in this work are interpreted in a first-order theory. I review the proof theory and model theory of first-order logic. Most results are stated

without proof. The reader may consult standard textbooks like Shoenfield [Sho67] for proofs. We also discuss decidability and complexity of the basic decision problems like satisfiability and validity. Fundamental concepts like soundness, completeness and consistency are also discussed. I also include subsections on two other possible extensions of propositional logic, namely, modal logic and quantum logic. Modal logic is included because a particular approach to reasoning about quantum computation is the dynamic logic approach and the latter is a type of modal logic. Quantum logic was developed to formalize the declarative content of quantum theory so that it is consistent with the predictions of the latter. It is a generalization of ordinary propositional logic so as to capture the peculiarity of quantum systems. Next, we discuss some applications in hardware and protocol analysis. These wide-ranging applications of the logical method was one of the main motivations for this work.

In the next section we review probabilistic logics. The primary focus is on the logic developed by Fagin, Halpern, and others [FHM90, HT93]. In this we work we adopt the basic philosophy of this approach- that probabilities are real numbers and reasoning about probabilities should include reasoning about real numbers. The first order theory of real closed fields plays an important role here. The probabilities that arise in quantum theory have more structure, as they are given by complex "amplitudes". We discuss the probabilistic logic of Fagin et. al. on several occasions. The reason is to bring out the similarities and differences in the reasoning of classical and quantum probabilities.

The next section is devoted to a survey of some of the alternative approaches to logical formulations for reasoning about quantum systems. The three approaches that are discussed in some detail are the categorical semantics of Abramsky and Coecke [AC04a], the dynamic logic approach of Baltag and Smets [BS04], and the logic developed by Mateus and Sernades. The last approach is close to the present one. We also discuss some of the

advantages and disadvantages of the current approach.

The last section gives a short review of computational complexity theory. The first question we ask about a function on integers is whether it *can* be computed. Then for the class of computable functions we may try to quantify the *cost* or *complexity* of computing a member of this class. This is important from a theoretical as well as a practical point of view. The characterization of complexity of functions and predicates is the subject of computational complexity theory.

2.1 Classical Logics

What we call "classical logic" consists of propositional logic and first order logic. These are the most widely used logics and form a major part of what is called mathematical logic. Propositional logic, also known as Boolean logic, sentential logic or propositional calculus is the simplest. We present it first.

2.1.1 Propositional Logic

The symbols of the logic consist of the following.

1. Propositional variables: The symbols p, q, r etc. will denote propositional variables. Thus, we assume we have set $\mathbf{S}$ of propositional or Boolean variables.

2. Logical Symbols: We have two logical symbols $\neg$ (negation) and $\vee$(disjunction). We also use two other logical symbols, $\wedge$(conjunction) and $\Rightarrow$(implication), but these are defined in terms of the first two.

The formulas of the logic are defined recursively as follows.

1. Any propositional variable p is a formula.

2. If $\mathbf{A}$ and $\mathbf{B}$ are formulas then so are $\neg\mathbf{A}$ and $\mathbf{A} \vee \mathbf{B}$.

3. Any formula is obtained by the above operations.

A formula such as p is called an atomic formula. We also call formulas propositions. The prescription for the construction of a formula may be succinctly expressed in the BNF formalism as follows

$$\mathbf{A} :: p|\neg\mathbf{A}|\mathbf{A} \vee \mathbf{B}$$

This simply means that an expression $\mathbf{A}$ is a formula if it is a propositional atom denoted by p, or it is a negation of a formula already constructed or disjunction of two formulas.

Let $\mathbf{Fm}$ be the set of formulas. Let $\mathbf{Pv}$ be the set of propositional variables. Any mapping $V : \mathbf{Pv} \to F_2$, where F_2 is a set with 2 elements, is called a valuation. For definiteness we take $F_2 = \{0, 1\}$. One may consider all the propositions mapped to 1 as "true". We extend now the valuation function V to all formulas as follows by induction on the length of the formulas

$$V(\neg(\mathbf{A})) = 1 \text{ iff } V(\mathbf{A}) = 0$$

$$V(\mathbf{A} \vee \mathbf{B}) = 1 \text{ iff at least one of } V(\mathbf{A}), V(\mathbf{B}) \text{ is } 1.$$

Note that this definition gives an effective way of evaluating the value of any formula(for a given valuation). In fact, given a formula we can obtain its truth value for all possible valuations since a formula is of finite length and has finitely many propositional variables. For each possible assignment the formula has a unique truth value. Here is a simple example. Let $\mathbf{F} \equiv p \vee \neg q \vee \neg r$, then its truth table is

p	q	r	**F**
0	0	0	1
0	0	1	1
0	1	0	1
1	0	0	1
1	0	1	1
1	1	0	1
0	1	1	0

A formula which takes the value 1 for all possible valuations is called a tautology. Informally, it means that it is true for all valuations. The formula $p \vee \neg p$ is a tautology. If a formula evaluates to true for some valuations of the truth values it is called *satisfiable.* Thus a formula is unsatisfiable if and only if it evaluates to 0 for all valuations of its variables. The two basic problems of propositional logic are satisfiability and validity. These two problems are dual in the sense that a formula $\mathbf{A}$ is valid if and only if $\neg\mathbf{A}$ is unsatisfiable. A valuation V and the rules for valuation of any formula essentially constitute the semantics of propositional logic. We may also *interpret* the logic in any nonempty set S. Thus, an interpretation π is a map from $\mathbf{Pv} \to 2^S$, the set of subsets of S. This map is extended to the set of formulas $\mathbf{Fm}$ as follows. Assuming $\mathbf{A}$ and $\mathbf{B}$ are already defined define $\pi(\mathbf{A} \vee \mathbf{B}) = \pi(\mathbf{A}) \cup \pi(\mathbf{B})$ and $\pi(\neg\mathbf{A}) = S - \pi(\mathbf{A})$(the complement of $\pi(\mathbf{A})$ in S). An interpretation defines a valuation at each point $x \in S$. For let $\chi_T : S \to F_2$ denote the *characteristic* function of $T \subset S$, defined by $\chi_T(x) = 1$ iff $x \in T$. Then for a fixed $x \in S$, $p \to \chi_{\pi(p)}(x)$ is a valuation. Each such point x is a world and p is true in this world iff $x \in \pi(p)$. If the cardinality of the set S is greater than or equal to that of $\mathbf{Pv}$ then all valuations can be obtained as interpretations. We note that the set of formulas of a propositional logic form a Boolean algebra if we add the two propositional constants $\top$ and $\bot$ which evaluate to 1(true) and 0(false) respectively and identify any two formulas $\mathbf{A}$ and $\mathbf{B}$ for which $\mathbf{A} \Leftrightarrow \mathbf{B}$ is valid. The operations of Boolean algebra are defined by $\mathbf{A} \vee \mathbf{B}$, $\neg\mathbf{A}$, and $\mathbf{A} \wedge \mathbf{B} \equiv \neg(\neg\mathbf{A} \wedge \neg\mathbf{B})$ [BS69]. This is the Lindenbaum algebra of(classical) propositional logic. The connective $\wedge$ is defined in terms of the primary connectives $\vee$ and $\neg$. To make $\mathbf{Fm}$ a Boolean algebra we have to state some axioms for the connectives. Informally, they state that the two binary operations $\vee$ and $\wedge$ are commutative, associative and distributive over each other, both are idempotent, the elements $\top(\bot)$ satisfy $\top \wedge a = a(\bot \vee a = a)$ and $a \vee \neg a = \top(a \wedge \neg a = \bot)$. Thus, a Boolean algebra is a set with two

binary operations $\vee$, $\wedge$ and unary operation $'$(we use the standard notation for set complement) such that

1. $a \vee b = b \vee a \quad a \wedge b = b \wedge a$ (commutative)
2. $a \vee (b \vee c) = (a \vee b) \vee c \quad a \wedge (b \wedge c) = (a \wedge b) \wedge c$ (associative)
3. $a \vee (b \wedge c) = (a \wedge b) \vee (a \wedge c) \quad a \wedge (b \vee c) = (a \vee b) \wedge (a \vee c)$ (distributive)
4. $a \wedge a = a \vee a = a$ (idempotent)
5. $a \wedge \bot = 0,\ a \wedge \top = a, \quad a \vee \bot = a,\ a \vee \top = \top$ (identity)
6. $a \vee a' = \top,\ a \wedge a' = \bot$ (complements)

This axiomatization is a variant of Stone's axiomatization. In the language of Boolean algebras $\vee$ is called the join and $\wedge$ the meet of two elements. In the literature, $\top$ and $\bot$ are often written as 1 and 0 respectively. Quantum logic, which deals with quantum propositions is *not* a Boolean algebra. The axiom that fails is the distributive axiom as we will see. But I mention two important examples.

1. The set of subsets of a set $\mathcal{S}$. The join is given by set union and the meet by set intersection. The constant $\bot$ corresponds to the empty set and $\top$ to the whole set. If $A \subset S$, then A' is the complement of A.
2. Let $F_2 \equiv \{0,1\}$ be the two element set mentioned above. Let $1 \wedge 1 = 1 \vee 0 = 1, 1 \vee 1 = 1,\ 1 \wedge 0 = 0 \wedge 0 = 0$ and $1' = 0, 0' = 1$. The other operations are defined by commutative property. It is easily verified that F_2 is a Boolean algebra. An important fact is that any Boolean algebra $\mathcal{B}$ can be homomorphically mapped to F_2 [BS69]. A homomorphism is a mapping H that preserves all the operations, that is, $H(a \vee b) = H(a) \vee H(b), H(a \wedge b) = H(a) \wedge H(b)$, and $H(a') = H(a)'$. Let us call this the F_2-property. Any such mapping from the Boolean algebra of formulas is precisely a valuation.

We have had a discussion about the *semantics*, including the algebraic aspects, of propositional logic. Let us now focus our attention on the axiomatics. In any logical theory an axiomatic approach involves the following: set of formulas called the axioms of the theory *and* some rules of deduction. Recall that a formula is a member of subset **Fm** of strings, from the set of all strings of the alphabet. An axiom usually defines a set of strings of **Fm** with some specific *structure*. A rule on the other hand prescribes how new formulas(called theorems) may be obtained from old ones. Thus, informally, rules give us the allowed structural manipulations of formulas. We have already seen an example of axioms above in the definition of Boolean algebra. We now consider the axiomatization of propositional logic. First, define some new connectives in terms of the old ones.

$$p \Rightarrow q \equiv \neg p \vee q \quad p \wedge q \equiv \neg(\neg p \vee \neg q)$$

The connective $\Rightarrow$ is called the implication. Its truth table is

p	q	p⇒ q
0	0	1
0	1	1
1	1	1
1	0	0

We also define the logical equivalence $p \Leftrightarrow q \equiv (p \Rightarrow q) \wedge (q \Rightarrow p)$ and the exclusive "or", $\oplus$, $p \oplus q \equiv (p \wedge \neg q) \vee (\neg p \vee q)$. The exclusive "or" is same as addition modulo 2. A possible axiomatization of propositional logic is to modify the axioms of Boolean algebras, replacing $=$ with $\Leftrightarrow$. However, I adopt the axiomatization of Shoenfield [Sho67] since these will form a part of the first order theory. The axioms are all instances of the formulas

Prop1 $\neg \mathbf{A} \vee \mathbf{A}$

Prop2 $\mathbf{A} \Rightarrow \mathbf{B} \vee \mathbf{A}$

Prop3 $\mathbf{A} \vee \mathbf{A} \Rightarrow \mathbf{A}$

Prop4 $\mathbf{A} \vee (\mathbf{B} \vee \mathbf{C}) \Rightarrow (\mathbf{A} \vee \mathbf{B}) \vee \mathbf{C}$

Prop5 $(\mathbf{A} \Rightarrow \mathbf{C}) \Rightarrow ((\neg\mathbf{A} \Rightarrow \mathbf{B}) \Rightarrow \mathbf{C} \vee \mathbf{B})$

We also have one inference rule.

Modus Ponens Infer $\mathbf{B}$ from $\mathbf{A}$ and $\mathbf{A} \Rightarrow \mathbf{B}$

These axioms and inference rules are not identical to those of Shoenfield, but the later can be derived from them. I have avoided the contentious "cut" rule [Dal94]. A theorem of propositional logic is either an axiom or a formula which is derived from other theorems using the inference rule/s.

Definition 1 *An axiomatization of a logic is said to be sound if every theorem is valid.*

We also note that these are properties which relate the axiomatization with semantics. Hence these notions cannot be expressed in the language of the logical theory, the *object* language, itself. It must be stated in a metalanguage like English. See Church [Chu56] for a good discussion of this point. We come back to these and other related features after we introduce first order logic. We only mention that the axiomatization of propositional logic presented above is sound and complete.

2.1.2 Modal Logics

The first extensions of propositional logic that we consider are called *modal logics*. A modal logic is an extensions of propositional logic [Gol92]. Hence the language L of a modal logic has all the symbols of propositional logic discussed above. Further we have a new logical symbol $\Box$, called a modal operator. A formula is defined recursively as follows:

1. All propositional variables are formulas. These are the atomic formulas.
2. If $\mathbf{A}$ and $\mathbf{B}$ are formulas then so are $\mathbf{A} \vee \mathbf{B}$, $\neg\mathbf{A}$, and $\Box\mathbf{A}$.
3. Any formula is constructed as above.

The operator $\Box$ was motivated by the desire to formalize linguistic notions such as "necessity", "perpetuity" etc. Informally, we say that $\Box\mathbf{A}$ is true means that $\mathbf{A}$ is necessarily true. However, in computer science, the modal operator has different interpretations. We will see some examples of these. First, the semantics of modal logic. Let $\mathcal{F} = \{S, R\}$ be a pair such that S is a set(the set of *worlds or states*) and R is binary relation, called the transition relation, on S. The pair $\mathcal{F}$ is called a . Let $\mathbf{Fm}_L$ be the set of formulas of L. A function $V : \mathbf{Fm}_L \times S \to F_2$ is called a valuation. Intuitively, a formula $\mathbf{A}$ is true at a state $s \in S$ iff $V(\mathbf{A}, s) = 1$. We write this as

$$s \models \mathbf{A}$$

The valuation V is not arbitrary except on $\mathbf{Pv}$, the set of *atomic* formulas. Given an arbitrary function $V : A \times S \to F_2$, we extend it to a function $V : \mathbf{Fm}_L \times S \to F_2$ as follows.

$$V(\mathbf{A} \vee \mathbf{B}, s) = \max(V(\mathbf{A}, s), V(\mathbf{B}, s))$$

$$V(\neg\mathbf{A}, s) = 1 - V(\mathbf{A}, s)$$

$$V(\Box\mathbf{A}, s) = \prod_{\{t|\ sRt\}} V(\mathbf{A}, t)\ (\textstyle\prod \text{ denotes the product})$$

In other words, $\mathbf{A} \vee \mathbf{B}$ is true at a state s iff at least one of $\mathbf{A}$ and $\mathbf{B}$ is true, $\neg\mathbf{A}$ is true at s iff $\mathbf{A}$ is false there and $\Box\mathbf{A}$ is true at s iff it is true at all the states related to s, that is all $\{t|\ (s,t) \in R\}$. This semantics was proposed by Kripke [Kri59, Kri63]. We see that only the truth value of boxed formulas are dependent on the worlds that are different from but related to the current state. Compare it with the valuation of formulas in propositional logic. The truth value of a proposition depends only upon the current state. Properties of modal logics depend strongly upon the relation R. We mention two important cases.

1. R is an equivalence relation. The corresponding modal logic is called S_5. An important subclass of this class are the epistemic logics or

logics of knowledge. In this case the set S has well-defined structure capturing the states of the system under consideration.

2. R is the graph of a *function* $R : S \to S$. Then the corresponding frame is called functional. We will encounter an example of functional frame later.

A formula $\mathbf{A}$ is valid in a class $\mathcal{C}$ of frames if it is true for all valuations and frames in $\mathcal{C}$. It is possible to axiomatize most modal logics. The axioms often capture the frame property. For example, for S_5, the frames are based on equivalence relations. First define the operator

$$\Diamond \mathbf{A} = \neg \Box \neg \mathbf{A}$$

Note that $\Diamond A$ is true at a state s if there is state s' such that sRs' and $s' \models \mathbf{A}$. Now the axioms for S_5 are

1. $$\Box \mathbf{A} \Rightarrow \mathbf{A} \text{ reflexivity}$$

2. $$\mathbf{A} \Rightarrow \Box \Diamond \mathbf{A} \text{ symmetry}$$

3. $$\Box \mathbf{A} \Rightarrow \Box \Box \mathbf{A} \text{ transitivity}$$

Thus, for example, any formula of the form $\Box \mathbf{A} \Rightarrow \mathbf{A}$ is valid in all reflexive frames, and conversely if we require that all such formulas be valid then the frame must be reflexive. Similarly functional frames are characterized by the schema $\Diamond \mathbf{A} \Leftrightarrow \Box \mathbf{A}$.

Modal logics have proved very useful in the formal modeling of sequential hardware, program verification, and protocol analysis among others. I discuss some simple examples. First, we observe that it is easy to extend the definitions above so we may include several modal operators, $\Box_1, \Box_2, \cdots$ with respective relations $R_l, R_2, \cdots$.

2.1.3 Circuits and Hardware

A classical or Boolean circuit is a combination of logic elements or gates [Vol98]. It is simply a function of Boolean variables and may therefore be adequately described by a Boolean or propositional formula. However, if a circuit contains timing elements("clocks") or storage elements which have "memory" like flipflops then a Boolean formula no longer describes such sequential circuits. We need a more expressive language. Modal logics such as **CTL**(computational tree logic) and **LTL**(linear time logic) can handle such circuits efficiently. Such circuits can be thought of as transition systems. Informally, a transition system gives the possible next state of the system from the present state. Typically, the state of the system is given by the value of all the registers, that is, a set of Boolean or propositional variables. There may be more than one next state. A typical modal operator is denoted by X, the next time operator. A formula Xp is true in the current world if p is true in all possible "next worlds". As an example,consider the modulo-8 counter which is expressible in a language with at least three propositional variables, $\{p_0, p_1, p_2\}$. Then, the formula

$$Xp_0 \Leftrightarrow \neg p_0 \wedge Xp_1 \Leftrightarrow p_1 \oplus p_0 \wedge Xp_2 \Leftrightarrow p_2 \oplus (p_1 \wedge p_0)$$

is a constraint describing the *ideal* behaviour of the counter at any point of time. The reader may refer [CBG+92] or [KG99] for more on formal verification of hardware. I have emphasized the word ideal because in any real system there is nonzero probability that the components behaviour deviates from the ideal. How do we incorporate these probabilities in a language? One possibility is to use the probabilistic extension of **CTL** called **PCTL**. Roughly, **PCTL** formulas are **CTL** formulas with probabilities attached [BKR94]. We do not go into the details because we are soon going to discuss an alternative logic for dealing with probabilities.

2.1.4 Program Verification

A simple program consists of declaration, assignment and control statements. The control statements may be branching type or iterative. In propositional dynamic logic(**PDL**) [HK00], there are two kinds of expressions besides the logical symbols: formulas and programmes. For each programme α, $[\alpha]$ is modal operator acting on formulas. Thus, if A is formula then so is $[\alpha]A$. Intuitively, this means that after the programme α, the formula A holds. The programmes such as α have a structure too, built from atomic formulas [HK00, Gol92]. The language consists of two distinct components: programmes and formulas. The programmes are built from simple atomic programmes *and* conditional test $A?$, which means informally "test the formula A" for truth. The formulas of the language are the usual propositional formulas and expressions of the form $[\alpha]A$, where α is programme and A is formula. The intended meaning is the following. The formula $[\alpha]A$ is true iff after the execution of the programme α, A is true. Again, we will not go into the details since it will take us too far afield. We only observe that one of the logics developed has some resemblance to **PDL**, where various quantum operators play the role of programmes.

There is an alternative and older approach to programme analysis developed by Tony Hoare [Hoa85]. It is based on the notion of predicate transformation. The programme state is given by a first order predicate(see below) and the statements of the logic are in the form $\{P\}S\{Q\}$, where P and Q are predicates pertaining to the programme states respectively *before* and *after* the programme S executes successfully(with termination).

2.1.5 Protocol Analysis

This is a broad and relatively new area. First, the word protocol is used in many different areas, e.g., concurrent computation, networks, and cryptography to name a few. We will just focus on cryptographic protocols where the issue of security is paramount. The first significant application of logic to

formalizing the interactions among agents participating in a cryptographic protocol was the paper [BAN90]. The authors used modal operators to capture notions like "belief". Although it consisted of simple and intuitive rules it could be used to discover serious flaws in key distribution protocols. Cryptographic protocols are often analyzed using the notions of computational hardness. Thus a protocol is secure if it can be shown that it is computationally "hard" to break it. For example, factorizing a large number is computationally hard. However, as Burrows et. al. [BAN90] and others [Low96] showed even in the early days, that there are subtle attacks which can circumvent the computational barrier. These attacks are mostly discovered by a careful *formal* analysis. More recently Abadi [AR02] and Rogaway showed how the two views of security(computational and formal) may be reconciled. In many classes of protocols involving interacting agents the information state or *knowledge* and *belief* of agents plays an important role. Knowledge and belief have been successfully modeled as modal operators [FHMV95]. I mention it here because *quantum cryptographic protocols* is an important branch of Quantum Information Science. There is a crucial difference between knowledge in quantum and classical systems [MP03a]. This area is still being developed and we only touch upon it in the concluding section.

2.2 First order logic

The language of first order logic with equality consists of three disjoint sets, $\mathcal{X}, \mathcal{P}$, and $\mathcal{L}$. Elements of the first set $\mathcal{X}$ are called variables and those of $\mathcal{P}$ are pairs of the form (f, n) or (p, n), n a nonnegative integer. The integer n is called the 'arity' of a function f or predicate p. We further assume that the set of function symbols, denoted by $f, g, h, f_i, \cdots$ and the set of predicate symbols denoted by, $p, q, r, p_i, \cdots$ are disjoint. We also suppress the arity since in most cases it is clear from the context. The variables will be denoted by the letters $x, y, z, y, , X_i \cdots$. The logical symbols are

$\{\vee, \neg, \wedge, \Rightarrow, \Leftrightarrow, \exists, =\}$. We also have the defined symbol $\forall \equiv \neg\exists\neg$. There are two kinds of expressions in the logic, *terms* and *formulas*, defined below.

1. Every variable is a term. If $\mathbf{t}_1, \cdots, \mathbf{t}_n$ are terms and f is a n-ary function symbol then $f(\mathbf{t}_1, \cdots, \mathbf{t}_n)$ is a term. The 0-ary functions are called constants.

2. Let $\mathbf{t}_1, \cdots, \mathbf{t}_n$ be terms. Then any expression of the form $\mathbf{t}_i = \mathbf{t}_j$ is an atomic formula. If p is an n-ary predicate symbol then $p(\mathbf{t}_1, \cdots, \mathbf{t}_n)$ is an atomic formula. We now define composite formulas recursively. If F_1 and F_2 are formulas and x is variable then

$$\neg F_1, F_1 \vee F_2, F_1 \wedge F_2, F_1 \Rightarrow F_2, \exists x F_1, \text{ and }, \forall x F_1$$

are formulas.

We next discuss the semantics of first order logic. The formulas of first order logic are symbolic expressions which acquire meaning only when interpreted in some domain of discourse. The domain of discourse is some *set*. At this stage, we take a set to be a primitive notion without further elaboration. Let $\mathcal{L}$ be denote a first order language. A *structure* $\mathcal{S}$ for $\mathcal{L}$ consists of the following:

1. A nonempty set S called the domain or universe of $\mathcal{S}$. The members of S are called the individuals of the structure $\mathcal{S}$.

2. For each n-ary function symbol f of $\mathcal{L}$ an n-ary function $f_{\mathcal{S}}$ from S to S. Note that it follows from this definition that a constant denotes some fixed individual in a structure.

3. For each n-ary predicate symbol p of $\mathcal{L}$ an n-ary relation $p_{\mathcal{S}}$ in S. We recall that an n-ary relation in a set S is a subset of the set $\underbrace{S \times S \cdots \times S}_{n \text{ factors}}$.

An interpretation π of $\mathcal{L}$ in a structure is a map which assigns to each variable x in $\mathcal{L}$ an individual $\pi(x)$ of S. In most of this work we will assume the set of symbols representing variables is countable. Then an interpretation is simply a sequence of individuals of S. Hence, unless the universe is finite there are uncountably many interpretations. Now we can define the notion of "truth". Once variables are assigned some value(an individual) the formulas are like propositional formulas. We define this recursively starting from atomic formulas. For a formula $\mathbf{A}$ we write

$$\mathcal{S}, \pi \models \mathbf{A} \tag{2.1}$$

to indicate that $\mathbf{A}$ is true or $\mathbf{A}$ holds for the interpretation π. Below, $\mathtt{t}, \mathtt{t}_1, \mathtt{t}_2, \cdots, \mathtt{t}_n$ are terms.

1. $$\mathcal{S}, \pi \models \mathtt{t}_1 = \mathtt{t}_2 \text{ iff } \pi(\mathtt{t}_1) = \pi(\mathtt{t}_2)$$

   That is, $\mathtt{t}_1$ and $\mathtt{t}_2$ denote the same individual. Note that $\pi(\mathtt{t})$ is defined in the structure once the variables are assigned. This follows from the recursive definition of a term and the fact that all function symbols have a fixed meaning in the structure. For example, suppose f is unary function and x a variable, then $\pi(f(x)) \equiv \pi(f)(\pi(x))$.

2. For an n-ary predicate p,

   $$\mathcal{S}, \pi \models p(\mathtt{t}_1, \mathtt{t}_2, \cdots, \mathtt{t}_n) \text{ iff } (\pi(\mathtt{t}_1), \pi(\mathtt{t}_2), \cdots, \pi(\mathtt{t}_n)) \in p_{\mathcal{S}} \tag{2.2}$$

3. $$\mathcal{S}, \pi \models \neg \mathbf{A} \text{ iff not } \mathcal{L}, \pi \models \mathbf{A} \tag{2.3}$$

   That is, it is not the case that $\mathcal{L}, \pi \models \mathbf{A}$.

4. $$\mathcal{S}, \pi \models \mathbf{A}_1 \vee \mathbf{A}_2 \text{ iff} \tag{2.4}$$
   $$\mathcal{S}, \pi \models \mathbf{A}_1 \text{ or } \mathcal{L}, \pi \models \mathbf{A}_2 \text{ or both} \tag{2.5}$$

5.

$$\mathcal{S}, \pi \models \exists x \mathbf{A}(x) \text{ iff } \mathcal{S}, \pi[x \to a] \models \mathbf{A}(x) \text{ for some } a \in S \qquad (2.6)$$

Here $\pi[x \to a]$ is another interpretation which agrees with π for all syntactic variables except possibly x, and $\pi[x \to a](x) = a$. This corresponds to Tarski's semantics and notation [Tar56]. The variables within the scope of a quantifier $\exists$ or $\forall$ are called bound variables, a variable which is not bound is called free. Substitution is an important syntactic concept and we write $\mathbf{A}(\mathtt{t}/x)$ to denote that the term $\mathtt{t}$ is substituted for a free occurrences of a variable x. One must be careful in substitutions because an arbitrary substitution may cause a free variable to be captured. We will assume the renaming of bound variables so that they are always distinct from free variables.

6.

$$\mathcal{S}, \pi \models \forall x \mathbf{A}(x) \text{ iff } \mathcal{S}, \pi[x \to a] \models \mathbf{A}(x) \text{ for all} a \in S \qquad (2.7)$$

I do not give the semantics of the rest of the connectives since they can be defined in terms of the ones already given. A formula is *valid* in a structure if it is true for all interpretations. A formula is valid if it is valid in all structures. A formula is satisfiable in a structure if there is some interpretation π in which it is satisfiable. It is satisfiable if it is satisfiable in some structure. For closed formulas (formulas without free variables) the notions of validity and satisfiability are independent of interpretations. Note also that if x is the only free variable in $\mathbf{A}(x)$ then satisfiability of $\mathbf{A}(x)$ is equivalent to that of $\exists x \mathbf{A}(x)$ and validity to that of $\forall x \mathbf{A}(x)$.

2.2.1 Axiomatics

The axioms and rules of a first order theory are divided into two classes. The axioms and rules belonging to the first class are called *logical* axioms and rules and are common to all theories. The second class consists of *nonlogical* axioms and actually defines the structure. The theory of groups differs from

the theory of vector spaces in that they have different nonlogical axioms. We state the logical axioms below [Sho67].

1. All the axioms of propositional logic.
2. **Substitution:**$\mathbf{A}(\mathrm{t}/x) \Rightarrow \exists x\mathbf{A}$.
3. **Identity** $x = x$.
4. **Equality**:

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \Rightarrow f(x_1, \cdots, x_n) = f(y_1, \cdots, y_n)$$

$$x_1 = y_1 \wedge \cdots \wedge x_n = y_n \Rightarrow p(x_1, \cdots, x_n) \Rightarrow p(y_1, \cdots, y_n)$$

In this axiom f and p are respectively, any function or predicate symbol.

The rules of inference are given next.

1. **Modus Ponens**: From $\mathbf{A}$ and $\mathbf{A} \Rightarrow \mathbf{B}$ infer $\mathbf{B}$.
2. **$\exists$-introduction**: If x is not free in $\mathbf{A}$, infer $\exists x(\mathbf{B} \Rightarrow \mathbf{A})$ from $(\mathbf{B} \Rightarrow \mathbf{A})$.

There are no nonlogical inference rules. A first order language along with a set of formulas for the nonlogical axioms is called a first order theory. Examples of first order theory are: theory of groups, the theory of fields, the theory of natural numbers. We discuss the theory of real closed fields and algebraically closed fields in section 4.2. The reader may refer to [Sho67] for more examples and details.

A theorem in a first order theory T is either an axiom or a formula derived from existing theorems by the inference rules. A model for T is a structure in which every nonlogical axiom is valid. We note that all the logical axioms are valid in any structure. We have two properties of formulas, derivability as theorem and validity. We should expect that all theorems must be valid.

A theory is called *sound* if all its theorems are valid in every model of T. Note that this may be vacuously true since T may not have *any* model! However we always have

Theorem 1 *If T is a theory then every theorem is valid in every model of T.*

A theory is called consistent if it is not the case that $\mathbf{A}$ and $\neg\mathbf{A}$ are both theorems for some formula $\mathbf{A}$.

Theorem 2 *A theory is consistent if and only if it has a model.*

The proofs of these theorems and some others stated below may be found in [Sho67] or any other text on mathematical logic. The theorem 2 is deep and fundamental theorem of mathematical logic. It is equivalent to the following completeness theorem. A formula may be valid in some models and still not provable in a theory.

Theorem 3 *A formula $\mathbf{A}$ of a theory T is theorem if and only if it is valid in every model.*

The two preceding theorems are equivalent in the sense that one implies the other. They are called completeness theorems. A formula $\mathbf{A}$ in a theory T is *decidable* if $\mathbf{A}$ or $\neg\mathbf{A}$ is a theorem of T. Otherwise it is *undecidable*. The theory T is complete if every *closed* formula in T is decidable. Examples of complete theory are: the theory of algebraically closed fields and the theory of real closed fields. Peano arithmetic is *incomplete* [Sho67]. This is Godel's famous incompleteness theorem. For a theory which is known to be decidable, for example, the theory of real closed fields, a natural question is the *complexity* of the decision procedure. Informally, this means the time or space taken by a Turing machine to execute the procedure. It is expressed as function of some positive integer giving a measure of the size of the formula. The usual measure is its length, defined to be the total number of symbols appearing in the formula.

2.3 Logics for reasoning about probability

2.3.1 Probability Theory

Many classical and almost all quantum systems have inherent uncertainties. Given that the system is in some state the next state may not be deterministically predicted. Often, we can assign positive real numbers ≤ 1 to the various alternatives which are called the probabilities of these alternatives. We will not go into the debate about the subjective/objective nature of the probabilities. For us it suffices to note that operationally we have effective procedures for consistent assignment of the probabilities and that in a large number of trials the relative frequencies of the alternatives tend to the assigned probabilities. The widely accepted form of mathematical probability is the axiomatic method of Kolomgorov [Kol56]. Probability theory is founded on the notion of a measure space. In fact, probability by definition is a measure. Thus a probability space is a triple $\{S, \mathcal{M}, \mu\}$, with the following properties.

1. S is set and $\mathcal{M} \subset 2^S$ is a σ-algebra of sets. A σ-algebra is generalization of Boolean algebra that is closed under countable unions.

2. $\mu : \mathcal{M} \to \mathbb{R}^+$ is a function into nonnegative real numbers which satisfies the following:

   $\mu(S) = 1$ and if $A_1, A_2, \ldots, \in \mathcal{M}$ is a countable collection of pairwise disjoint sets then

   $$\mu(\bigcup_n A_n) = \sum_n \mu(A_n)$$

   The last property is called countable additivity. If it holds for finite collections only then the corresponding property is called finite additivity. We will require finite additivity only.

Note that a μ induces a finitely additive measure on any Boolean subalgebra of $\mathcal{M}$.

Again, I will not go into the details. See [Fel57] for a vintage but still relevant introduction or [Wil91] for a more recent one. However, our primary interest is not probability theory *per se* but formal reasoning about systems which obey probabilistic laws.

In the previous chapter we have seen that formal reasoning helps us represent, analyze and synthesize complex interactive systems, both physical(e.g. hardware) and abstract(e.g. concurrent programs). In such formal modeling we have to often make abstractions or idealizations about the behaviour. The representation of a transistor as a logical operator is such an abstraction. In case the system's development in time or evolution is not completely deterministic there are two possibilities: we assume *nondeterministic* behaviour, that is the various alternative "worlds" or events cannot be assigned probabilities in any reasonable way, or the development is stochastic. It is the second possibility which is relatively more difficult to model, both conceptually and technically, that concerns us in this work. Several authors have attempted to formalize probabilistic reasoning [AH94, Bac90, Car50, Nil86, FHM90]. Only few of the large body of literature is cited and the reader may explore the references in them. Nilsson's paper is perhaps the first one intended for applications. In Nilsson's logic one assigns probabilities to consistent valuations of propositional variables. This is also the semantics of a logic developed in [FHM90]. We briefly present the latter.

Recall from section 2.1.1 that every valuation may be generated by a map π from the set of atomic formulas to a set $\mathcal{N}$ of subsets of some set S. We may restrict $\mathcal{N}$ so that the map is onto. The Boolean algebra $\mathcal{B} \subseteq 2^S$ generated by $\mathcal{N}$ is homomorphic to the Boolean algebra generated by the atomic formulas, that is, the set $\mathbf{Fm}$ of formulas. We assume that there is a probability measure μ defined on $\mathcal{B}$. We only require finite additivity of the measure (in contrast to σ-additivity) since in most cases extension to a full measure is possible [Hal50]. Then, we say that the probability of an atomic

formula q is given by

$$P(q) = \mu(\pi(q))$$

The probability is easily extended to all formulas. A general *atomic* formula is of the form

$$c_1 P(\phi_1) + \cdots + c_n P(\phi_n) \geq k.$$

where the c_i and k are integers. This is satisfied in the probability structure $\mathcal{X} = \{S, \mathcal{M}, \pi, \mu\}$, written as,

$$\mathcal{X} \models \Phi \text{ iff } c_1 \mu(\pi(\phi_1)) + \cdots + c_n \mu(\pi(\phi_n)) \geq k$$

A general probability formula is a Boolean combination of atomic formulas. Note that the measure space $\mathcal{X}$ is quite arbitrary. We will see that in the case of the logic of quantum probabilities this is not the case. This logic is axiomatized as follows.

1. **Taut.** All instances of propositional tautologies
2. **Mod. Pon.** From A and $A \Rightarrow B$ infer B.
3. All instances of valid formulas about linear inequalities.
4. **P1.** $P(\phi) \geq 0$
5. **P2.** $P(\top) = 1$
6. **P3.** $P(\phi \wedge \neg\psi) + P(\phi \wedge \psi) = P(\phi)$
7. $P(\phi) = P(\psi)$ if $\phi \Leftrightarrow \psi$ is a propositional tautology.

The axioms of linear inequality are given in the papers cited above [FHM90, HT93]. The theory of linear inequalities is finitely axiomatized and is a complete theory for the axiomatization. Hence, for our purposes it suffices to take all valid formulas of the theory of linear inequality as axioms. It was shown there that this logic is sound, decidable, and complete. The complexity of the decision procedure for satisfiability is NP-complete. The

authors then extend the logic to deal with conditional probabilities. In this case, linear formulas are no longer adequate. The authors therefore consider a logic which is interpreted in a first order theory of real closed fields(RCF) [Sho67]. We describe their system briefly. The atomic formulas of the language $\mathcal{L}_C$ are obtained as follows. Let $f(x_1, \cdots, x_n)$ be a multivariate polynomial with integer coefficients. Substitute probability formulas $P(\phi_i)$ uniformly for some occurrences of x_i. Denote the resulting expression by Φ. Then $\Phi \geq 0$ is an atomic formula of $\mathcal{L}_C$. Compound formulas may be obtained as in any first order theory. The semantics is similar to the linear case. This theory may be recursively axiomatized. Since there are now variables ranging over some **RCF** we have to include the axioms of RCF. A decision procedure for the logic of $\mathcal{L}_C$ is obtained by systematic reduction to corresponding formulas in RCF. The latter is known to be a complete and decidable theory hence the axiomatization of $\mathcal{L}_C$ is complete and decidable. The present work follows the approach of [FHM90, HT93]. In the later chapters we refer to these works on several occasions.

2.4 Other Approaches to Reasoning about Quantum Probabilities

There have been several approaches to develop a "logical structure" of quantum mechanics starting with Birkhoff and von Neumann's pioneering work [BvN36]. They coined the term "quantum logic". In the decades following that work there was some significant development in this area. More recently, intense activity in the new and rapidly developing area of quantum computation and information has motivated logicians and computer scientists to take fresh interest in developing a logic for complex quantum protocols in the spirit of classical computer science. We will review most of these approaches and its relation to our approach.

2.4.1 Quantum Logic

Quantum logic was proposed by Birkhoff and von Neumann [BvN36]. Since then this field has grown with some significant contribution from many researchers. See Piron's book [Pir76] for development up to 1970's. More recently, Rawling and Selesnick [RS00] reviewed *orthologics*, a generalization of *minimal* quantum logics and gave some applications to quantum computation. We have seen that the algebraic semantics of propositional logics correspond to Boolean algebras, a distributive complemented lattice. Stone's representation theorem [Sto36] states that any Boolean algebra is isomorphic to a field of subsets of a set. A field of subsets of X is a set of subsets that contains X and is closed under complementations and intersections. The set X is actually the set of ultrafilters. An ultrafilter is a set of true propositions that is closed under conjunction and implication and is maximal in the sense that for any proposition α, either α or $\neg\alpha$ belongs to it. We have already seen an informal construction of this.

Let us analyze a realization of Boolean algebra in the context of classical physics. Here we follow the discussion given in Chapter 1 of Bub's book [Bub97]. The classical description of a physical system is given by points $(\mathbf{q}, \mathbf{p})$ in the phase space where $\mathbf{q}$ and $\mathbf{p}$ are the positions and momenta of the particles in the system. Thus each point $(\mathbf{q}, \mathbf{p})$ is a *state* of the system. We will not go into the details but it is sufficient to consider $\mathbf{q}$ and $\mathbf{p}$ as real vectors in n dimensions. An observable like energy is a real function $f(\mathbf{q}, \mathbf{p})$. A typical *classical proposition* is $f(\mathbf{q}, \mathbf{p}) > 0$. We take only a finite number of such propositions. We may also rewrite this as $f(\mathbf{q}, \mathbf{p}) \in (0, \infty)$. Thus, given an observable and a set of subsets on the real line the propositions are assertions about membership at some point in time. The actual truth of these propositions is of course determined by the dynamics of the system. But what we are interested in is the various logical relationships among the propositions. It is clear that ordinary propositional logic is adequate to deal with such relationships. Even when we have incomplete information we

may assign probabilities in the standard way because the set of propositions forms a Boolean algebras.

In the quantum case, we do not have a phase space description and the indeterminacy is not due to incomplete information but it is inherent. We may still talk of a particular state lying in some region(more precisely a *subspace*) but there is "incompatibility" between different propositions. Heisenberg's celebrated uncertainty principle is a statement of such incompatibility. Therefore, in building the quantum analogue of propositional logic we consider as our basic model the subspaces of a Hilbert space rather than subsets. The Boolean operation meet $\wedge$ corresponds to intersection of subspaces and complementation to orthogonal complement,$\perp$, of a subspace. The join may be derived as $A \vee B \equiv (A^{\perp} \wedge B^{\perp})^{\perp}$. The resulting lattice, called an ortholattice, is *not* distributive. The quantum probabilities are given by a very different prescription(see chapter 3). Ortholattices have many properties quite different from Boolean lattices. Classical probabilities are defined on a Boolean lattice(more generally, on a σ-algebra). Quantum probabilities are defined on an ortholattice. We refer to [Pir76] for a good account. But we mention three important theorems.

1. **Piron.** Every ortholattice can be realized as an ortholattice of subspaces of a Hilbert space. [Pir76]

2. **Gleason.** The only probability measures definable on the ortholattice of quantum propositions in dimension$\geq$ 3 are the ones given by quantum theory. [Per95]

3. **Kochen-Specker** If the dimension is $\geq$ 3 it is not possible to make consistent truth assignments in all the Boolean sublattices of an ortholattice. [Per95]

I have stated these theorems somewhat imprecisely avoiding some technical restrictions. The point is that, when we want to reason about quantum probabilities then Gleason's theorem restricts us to the measures generated

by $\mathtt{Tr}_\rho$ where ρ is a state(see chapter 3). In this work, we do not study the origins of quantum probability. Rather, we take the structure of quantum probabilities for granted and study the problem of representing this in a formal framework. Therefore, unlike classical probabilistic logics where we have considerable freedom with the choice of measures the above trace measure must be implicit in logics for quantum probability. We may not assign simultaneous truth values all to quantum propositions but we may do so to any particular Boolean sublattice *and* assign simultaneous probabilities to incompatible propositions. This does not mean that we consider simultaneous measurement of incompatible observables possible, but that *choice* of the alternatives is always possible. Statements like,"if I am to measure A I get result x with probability $1/2$ *and* if I measure B instead I get result y with probability $1/3$" is perfectly legitimate. This is the starting point of our logic: to avoid the complicated reasoning with quantum logic and provide an operational framework formally capturing the kind of reasoning used by practicing experimental physicists.

2.4.2 Exogenous Quantum Propositional Logic(EQPL)

Mateus and Sernadas proposed a propositional type of logic called *exogenous quantum propositional logic*(EQPL) [MS04b, MS04a]. They introduce a "classical" propositional basis which is the substructure for the Hilbert space quantum superstructure. More precisely, a set of propositional constants $\mathtt{q}B = \{\mathtt{q}_k\}$ forms the basis of the classical logic. Let V be the set of valuations on $\mathtt{q}B$. Form the free complex vector H space with V as the basis. The space H is simply the set of mappings of $V \to \mathbb{C}$ with point wise addition and multiplication by scalars and introduce the standard scalar product. We will assume $\mathtt{q}B$ to be finite for simplicity. A Nilsson type probability structure is introduced on the Boolean algebra of valuations but the probabilities are restricted to be quantum probabilities. New connectives are introduced whose semantics depends on global satisfaction(e.g. **A** false

unless true for all valuations from a given set of valuations). The other constructs are a kind of implication and tensor product. There are also real and complex terms. The authors also provide a weakly complete axiomatization of the logic.

Let us look at the propositional substructure. Each valuation is simply a sequence of 0's and 1's. That is, it is an *indexing* of some basis. In the logics developed in the current work this indexing is implicit. We do not directly refer to the states, rather probabilities in different "propositional bases" and the unitary operations connecting them. Although, EQPL is without quantifiers new connectives and the extra structures have to be introduced to make it expressive enough. Moreover, the language is restricted to qubits. That is, the irreducible systems have dimension 2. Thus the Kochen-Specker no-go theorem is circumvented. The logics in the present work describe quantum systems in arbitrary dimension n (qunits instead of qubits!!). Finally, the operational nature of our work makes it easier to implement as a simulation.

2.4.3 Logic of Quantum Programmes

The ideas of a quantum programming language originated in the work of Knill [Kni96] who proposed a set of conventions for writing quantum algorithms in pseudocode. These conventions essentially followed the *imperative* programming paradigm. Informally, in the imperative paradigm a programme is a set of instructions defining the sequential transformations of a *global state*, where a state is a tuple of values taken by a set of variables. Most common programming languages like C, FORTRAN and Java are examples of imperative language. Several quantum programming languages within this framework were made [Ö98, SP00, BCS01]. Of these, only the language **qGCL**(quantum guarded command language) had a semantics. The language **qGCL** can also be regarded as a specification language. The languages developed in the present work can be considered higher level

languages for formal reasoning about quantum systems in general. As such, with a few additions we may use the latter for reasoning about *quantum programmes*. In this regard we mention that Baltag and Smets developed a logic for quantum programmes [BS04] in analogy with classical *PDL*. They introduce two formal constructs corresponding to "quantum tests" and unitary operations. Both these operations induce binary relations on set of program states defined as sets of quantum states. The authors develop Kripke-type semantics with these binary relations as accessibility relations. We mention that such Kripke frames can be introduced in the present framework if we modify the definition of states.

A totally different approach to programming is provided by the functional paradigm. In this approach a programme is a series of computation of functions. Since a function, in the mathematical sense, is a rule of assignment between an input set and an output set, the functional approach may be viewed as a"black box" transformations of the input to some output encoded in the functions. Examples of functional languages are Lisp and Haskell. Functional languages have clean and elegant semantics. The most complete and elegant work in this approach is that of Selinger [Sel04]. In this work Selinger proposes a typed language whose constructs are represented by flow charts. The basic programming construct- functional application, selection, looping and procedures are represented by certain atomic diagrams for the corresponding flow charts. Composite flow charts are built out of these by well defined rules. This high level language is interpreted in certain monoidal categories $\mathbf{Q}$ (see below). First one starts with a category $\mathbf{V}$ whose objects are tuples of matrices considered as a vector space in an obvious way and the morphisms are complex linear maps on the latter. The category $\mathbf{V}$ admits a tensor product structure. The category $\mathbf{Q}$ is a subcategory of $\mathbf{V}$ with the same objects but whose morphisms are restricted to those morphisms (in $\mathbf{V}$) which are completely positive. All known quantum algorithms can be represented by a corresponding flow chart. Selinger's lan-

guage incorporates classical and quantum operations. This is done through static typing.

2.4.4 Categorical Semantics

The categorical semantics developed independently by Abramsky and Coecke [AC04a, AC04b] and Selinger [Sel05] is an elegant and abstract formulation of quantum theory in a categorical framework. The type of category that seems to capture most of the features of quantum mechanics are the *strongly compact closed categories*. It will take us too far afield to give all the necessary definitions of such categories. We only give an informal discussion of its properties. First, we start with a symmetric monoidal category which is essentially a category with a unit object I and a product like the tensor product which is associative and symmetric in a natural sense [Mac71]. We denote the product by $\otimes$. Compact closed categories come equipped with a contravariant functor $\star$, that corresponds to the notion of dual space in the category of vector spaces. The $\star$-functor is also required to induce some natural bijections. For compact closure it is required that

$$(A \otimes B)^\star \to A^\star \otimes B^\star$$

Further it is required there be a unit and a counit which are morphisms between the unit object and $A^\star \otimes A$. In vector spaces the unit roughly corresponds to a resolution of the identity operator. Strong compact closure requires that the $\star$-morphism extends to a covariant functor among the duals. The main example is the category of finite dimensional Hilbert spaces in which the scalar product induces a conjugate linear isomorphism between the space and its dual. The authors also require the notion of coproducts, corresponding to direct sums in vector spaces, to give a categorical formulation of quantum mechanics. The notion of unitary morphism is derived from $\star$ using strong compact closure. Scalars are identified as morphisms of the unit object and the notion of coproduct is used to introduce bases. The probability rule(Born rule) can also be derived in this setting. The fact

that teleportation is actually some commutative diagrams in these abstract categories is very interesting but not surprising, if we carefully look at the teleportation protocol.

Selinger [Sel05] constructs a functor from strong compact closed category to a category which abstracts the properties of the category **Q** described above. The arrows representing the morphisms in the various categories can be represented as labeled diagrams. One may identify some of the diagrams with certain flow charts, thus completing the circle.

The categorical formulation is elegant and interesting, but it is also quite abstract. Does it mean that there are other concrete settings for quantum mechanics? Also there is the contentious issue of projective formulation of quantum mechanics, the state is given only up to a multiple of complex number of modulus 1, that is the states live in a *projective* space. I understand that work on modifications in the categorical framework to deal with the projective case is in progress. I also note that a generalization of the logic presented in the present work is naturally interpreted in the categorical framework.

The aim of the current work is to formalize the operational basis of quantum theory. We accept the fundamental concepts of the theory: probability, unitary operations, projective measurement and tensor product at their face value as understood by practicing physicists. We take the conventional Hilbertian axiomatic approach to mathematics, most of which can be formulated in a first order framework. It is true that many of the operations in vector spaces with tensor products is best formulated in categorical language. However, our aim was to give concrete algorithms for a large class of decision problem in finite dimensional quantum systems. Since the probability expressions are nonlinear(even more so in the projective formulation) and also since, in general, we have to deal with not just unitary matrices but their *individual entries*, a reduction to decision problems in real and complex numbers seemed natural. The later theories are very well investigated

and many algorithms exist(though not very efficient due to the very nature of the problems). As a dividend we get several complexity bounds. I have also experimented with some simple implementations of the algorithms in MatLab. In this work, we give an axiomatization of the logic and prove some general properties like quantifier elimination, completeness and decidability. The proof theory needs further investigation.

2.5 A Short Introduction to Complexity Theory

In this section we review the notions of computability and efficient computability. The first is connected with the *decision problem*: Given a subset A of E find a method which decides in finite number of steps whether an individual belongs to A *or* prove that no such method exists. This is the decision method for A in E. In the first case when we have a decision method we say that A is decidable. Similarly, we have the decision problem for functions: Given a function $f : A \to B$ find a method which produces $f(a)$ for any input $a \in A$ or prove that no such methods exists. In the former case we say that f is *computable* and in the latter f is *uncomputable*. The decision problem for functions is a generalization of the decision problem for sets. We observe that the elements of the relevant sets for which we pose the decision problem must be concrete objects like strings of letters from an alphabet. We must also state precisely what are the "steps" in the above definitions. For example, we can not include "consult a fortune-teller" as a step![1] These steps will define what constitutes an effective procedure or computation. We will use the Turing machine model. However, other models are possible but they are all equivalent as far as computability is concerned [Pap94].

A Turing machine $\mathcal{T}$ is informally defined as follows [HU79]. It consists of a tape on which input and output are printed. It has finite number of states. It can "read" symbols from an an "alphabet". These symbols are divided into two categories: the tape symbols and the input symbols. The

[1] We may, however, consult *oracles* for decidable problems for theoretical comparisons.

tape is divided into squares and the "read-head" of the machine reads one symbol at a time. A Turing machine can change its state, move the head one square left or right, or write a symbol on the square it is scanning. These transitions depend both on its current state and the symbol just scanned. The tape is of infinite extent to the right but there is a leftmost square. There is a unique *initial* state and a set of states designated as final state. The input string is left justified on the tape and the machine starts operating with its head in the leftmost square and in the initial state. If it reaches one of the final states we say that $\mathcal{T}$ *halts* by accepting. We also allow that for some combinations of states and symbols there is transition to a special "rejecting" state . In that case we say that $\mathcal{T}$ halts without accepting.

More formally a Turing machine is a tuple:

$$\mathcal{T} = \{Q, \Sigma, \Gamma, \Delta, q_0, \#, Z, F\}$$

where

Q is a finite set of states including a special state Z *rejecting*,

Γ is a finite set of *tape symbols*,

$\# \in \Gamma$ is the special symbol for blank,

$\Sigma \subset \Gamma - \{\#\}$ is the set of *input symbols*,

$\Delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$

is the *transition or next move function*

q_0 is the *initial state*,

$F \subset Q - \{Z\}$ is the set of final states,

The symbols L and R are the instructions to move to left or right respectively. The "rejecting" state is used so that Δ is a *function.* Alternatively, we could use *partial functions* and require that the domain of Δ is only a subset of $Q \times \Gamma$. We also suppose that the rejecting state is a halting state. The set Γ is also called the *tape alphabet* and the set Σ the *input alphabet.* The set of all strings or finite sequences of symbols from any alphabet

$\mathcal{A}$ is denoted by $\mathcal{A}^*$. The language accepted by the Turing machine $\mathcal{T}$ is the set $L_{\mathcal{T}} \subset \Sigma^*$ such that for any input string $x \in L_{\mathcal{T}}$ the machine halts in an an accepting state. We can now formulate the decision problems as computations of some Turing machine. A Turing machine may be used to compute functions of natural numbers. We say that a Turing machine $\mathcal{T}$ computes a function f of k arguments if for any input in the domain of f, $\mathcal{T}$ halts on an accepting state after printing the value of the function for the given arguments on its tape. Similarly, we can use a Turing machine $\mathcal{T}$ for decision problems on Σ^*. We say that it decides $L_{\mathcal{T}}$ if it halts on all inputs. That is, given $x \in \Sigma^*$ we can decide if $x \in L_{\mathcal{T}}$ if the machine halts with acceptance, otherwise $x \notin \Sigma^*$.

The number of moves of the Turing machine(when it halts) for an input x is denoted by $t(x)$. It gives an estimate of the time taken by the machine for the input x. Similarly, we define $s(x)$ as the number of tape square used by the machine during the computation. The computation time and space of $\mathcal{T}$ are defined by

$$T(n) = \max\{t(x) \mid \text{for all input strings of length } n\} \tag{2.8a}$$

$$S(n) = \max\{s(x) \mid \text{for all input strings of length } n\} \tag{2.8b}$$

The Turing machine(TM) described above is a *deterministic* machine because the transition relation Δ is a function. Suppose, we now allow nondeterministic transitions to *sets* of states. That is, $\Delta : Q \times \Gamma \to 2^Q \times \Gamma \times \{L, R\}$ where 2^Q is power set-the set of subsets-of Q. We now define $t(x)$ and $s(x)$ as the minimum over all possible paths of the computation determined by the transition relation. Then the computation time and space are defined as above- maximum over all inputs of a given length. If a function can be computed by a nondeterministic Turing machine (NDTM) whose computation time $T(n)$ is bounded by a polynomial in n then we say that the function belongs to the class **NP**. Alternatively, we characterize the corresponding language as **NP**. For example, we could consider the language L

as formulas of propositional logic in conjunctive normal form and require that such string(a CNF formula!) is accepted by an NDTM if and only if the formula is satisfiable. It can be easily shown that this language is in **NP**. Similarly, we define **P** as the set of languages whose computation time for a TM is bounded by some polynomial. No TM is known whose computational time is polynomially bounded for the language L above. There is a class of languages called **NP**-complete languages (**NPC**) defined as follows: $L_1 \in \mathbf{NPC}$ if and only if $L_1 \in \mathbf{NP}$ and any other language $L' \in \mathbf{NP}$ is polynomial time reducible to L_1. That is, there is map $g : L' \rightarrow L$ which is in **P** such that any $x \in L_1$ is decided by the machine T iff $g(x) \in L'$ is decided by T'. By the famous theorem of Cook[HU79] states that the language L for the satisfiability problem is **NP**-complete.

We often speak of problems rather than languages. Thus we say that the satisfiability problem is in **NP** meaning that it can be encoded in a language recognized by *some* NDTM whose computation time bounded by a polynomial in the length of the input. Moreover, we often do not construct the detailed machine and its transition and argue at a higher level with familiar operations like multiplication and addition of numbers. Implicit in this argument is the understanding that we can always go to the low level operations if needed. There are some problems for which no polynomial algorithm for a TM is known but which are believed *not* to be in **NP** but not **NP**-complete. The most well-known example is the problem of factoring a large integer.

Chapter 3

Quantum Theory

The most complete description of a physical system, according to quantum theory, is given by its state-a unit vector in a Hilbert space. However, this is strictly true for a 'pure' system. Consider a physical system S, say an electron. Often we may be interested only some attributes. Thus, for the electron we may wish to consider only its spin in a particular direction and ignore other attributes like energy or position. Then S refers to the electron as an object with spin only. This is especially true in quantum computing since attributes like spin, polarisation etc. take a finite number of values. Associated with S is a Hilbert space, that is a complex vector space with an inner product. We shall restrict ourselves to finite-dimensional spaces (corresponding to attributes like spin) which are adequate for quantum computing and information. The state is often associated not with a single system but a large collection of (possibly imaginary) identical systems called an ensemble. This is because we have only probabilistic information from our observations of S. But how do we describe an ensemble with electrons in two different spin states? Even for a single electron we may only have incomplete information about its state. This situation is described by generalising the notion of state. Henceforth, a state will mean a general state (often called a 'mixed state' or 'density matrix'). The special case is called a pure state.

Postulate 1 *Associated to S is a finite dimensional complex Hilbert space*

$\{H, \langle\rangle\}$ *called the state space. The dimension n is determined by the system.*

We identify H with C^n with standard inner product; if $|\alpha\rangle = (x_1 \dots x_n)$, $|\beta\rangle = (y_1 \dots y_n)$ then $\langle\alpha|\beta\rangle = \sum \overline{x_i} y_i$ where $\langle\alpha|$ is the conjugate vector $(\overline{x_1}, \cdots, \overline{x_n})$. The inner product satisfies $\langle\alpha|\beta_1 + \beta_2\rangle = \langle\alpha|\beta_1\rangle + \langle\alpha|\beta_2\rangle$ and $\langle\alpha|\beta\rangle = \overline{\langle\beta|\alpha\rangle}$ and $\langle\alpha|\alpha\rangle \geq 0$. The quantity $\| |\alpha\rangle \| = \sqrt{\langle\alpha|\alpha\rangle}$ is called the length of $|\alpha\rangle$. If it is 1 then α is called a unit vector. Note that the length or more generally $\langle\alpha|\beta\rangle$ is invariant w.r.t. multiplication of $|\alpha\rangle$ and $|\beta\rangle$ by arbitrary complex numbers of modulus 1.

With this notation we extend Postulate 1.

Postulate 2 *The pure states are represented by a unit vector, determined up to a scalar multiple of modulus 1. Moreover, each such vector is realizable as a state.*

Thus $|\alpha\rangle$ and $|\beta\rangle$ represent the same state iff $|\alpha\rangle = e^{ic}|\beta\rangle$ c real.

A basis $\mathbf{b} = \{\alpha_1, \cdots, \alpha_n\}$ of H is a linearly independent set of vectors such that every vector in H is a (unique) linear combination of the $|\alpha_i\rangle$'s. It is orthonormal iff $\langle\alpha_i|\alpha_j\rangle = \delta_{ij}$ where $\delta_{ij} = 1$ if $i = j$ and 0 otherwise. From any set of n linearly independent vectors we can construct an orthonormal basis. It is the set of orthonormal vectors which correspond to the classical notion of states. Their occurrence in any test can be considered as mutually exclusive events. Henceforth basis will mean an orthonormal one.

Postulate 3 *Any orthonormal basis represents a realizable maximal test.*

Let n be the maximum number of different outcomes possible in a given system for any test. For example we may test for value of the z-component of spin of an electron or polarisation of a photon. We imagine we have a large number of similarly prepared systems called an ensemble and we test for the values of different measurable quantities like spin etc.

For a spin-1/2 system we always get a maximum of 2 outcomes ('up' and 'down') for any test. So $n = 2$. This number is a property of the system

and according to the postulate equals the dimension of the state space. In general, we postulate that for an ensemble in an arbitrary state, it is always possible to devise a test that yields the n outcomes corresponding to an orthonormal basis with definite probability.

We note that if H is a Hilbert space with inner product $\langle\ ,\ \rangle$, then H is isomorphic to the dual space of linear functionals (i.e. complex valued functions) on H. Thus for each $|\alpha\rangle$ let $\langle\alpha|$ denote its image under this isomorphism, called the dual. Then $(\langle\alpha|)(|\beta\rangle) = \langle\alpha|\beta\rangle$ by definition. Further$(|\alpha\rangle\langle\beta|)(|\gamma\rangle) \stackrel{\text{def}}{=} (\langle\beta|\gamma\rangle)|\alpha\rangle$ is a linear operator on H. Let $\mathcal{L}(\mathcal{H})$ denote the space of linear operators on H. An operator $A \in \mathcal{L}(\mathcal{H})$ is hermitian if $\langle\alpha|A\beta\rangle = \langle A\alpha|\beta\rangle$ for all $|\alpha\rangle$ and $|\beta\rangle$. An operator U is called unitary if $\langle U\alpha|U\beta\rangle = \langle\alpha|\beta\rangle$ for all $|\alpha\rangle$ and $|\beta\rangle$. In matrix notation let $B^\dagger$ denote the transposed conjugate of a square matrix B. Then B is hermitian if $B = B^\dagger$ and U is unitary if $U^{-1} = U^\dagger$. In particular a unitary operator is invertible. The set of hermitian and unitary operators on H are denoted by L(H) and U(H) respectively. The former is a real vector space and the latter a group. They play a crucial role in quantum theory. For any unit vector $|\psi\rangle$ the operator $|\psi\rangle\langle\psi|$ is hermitian and satisfies $P^2 = P$ i. e. it is a projection operator (projecting onto 1-dim subspace generated by $|\psi\rangle$)s. Also $\mathtt{Tr}(P) = 1$ where Tr where the *trace* Tr is the sum of the diagonal elements of a square matrix, which is independent of the representation. For a pure state represented by $|\psi\rangle$ then the projection $|\psi\rangle\langle\psi|$ is called the state of the system. A general state ρ is defined to be a convex combination of pure states:

$$\rho = \sum p_i |\psi_i\rangle\langle\psi_i|\,,\ p_i \geq 0 \text{ and } \sum p_i = 1.$$

We consider ρ representing a system for which knowledge of the state is uncertain and the probability of it being in state $|\psi_i\rangle\langle\psi_i|$ is p_i. In general a state $\rho \in L(H)$ is a positive definite ($\langle\psi|\rho|\psi\rangle \geq 0$ for all $|\psi\rangle$) matrix with $\mathtt{Tr}(\rho) = 1$. The state is pure iff it is of the form $|\psi\rangle\langle\psi|$ for some unit vector $|\psi\rangle$. We continue to call $|\psi\rangle$ the state in this case.

Postulate 4 *If the system is prepared in state ρ and a maximal test corre-*

sponding to a basis $\mathbf{b} = \{|\beta_i\rangle \mid i = 1, \cdots, n\}$ *is performed, the probability that the outcome i will occur is given by* $p_i = \mathtt{Tr}(|\beta_i\rangle\langle\beta_i| \rho)$ *and the corresponding state will be* $|\beta_i\rangle$.

Since one of the outcomes must occur, $\sum p_i = 1$. In the relative frequency interpretation of probability this means that if we have an ensemble of N systems and perform a maximal test corresponding to $\{|\beta_j\rangle\}$ then if the frequency of outcome corresponding to $|\beta_i\rangle$ is n_i, we have $p_i = \lim_{N\to\infty} \frac{n_i}{N}$. If it is only known that a test is performed but not its outcome then the post-test state $\rho = \sum_i p_i \psi_i$ is "mixed". We may think the probabilities of outcome of a test depends on two kinds of uncertainties: the first (classical) arising out uncertainty about state and the second (quantum) from the inherent uncertainty in quantum systems. If the original state is pure, α then $p_i = |\langle\alpha|\beta_i\rangle|^2$.

If the system is known to be in one of the states in a basis $\{|\beta_i\rangle\}$, say $|\beta_1\rangle$, then $p_1 = 1$ and $p_i = 0$ for $i \neq 1$. That is, we can predict the outcome with certainty for this maximal test. This is the case that corresponds to the classical theory. However, if we choose a different maximal test corresponding to a different basis then the outcomes become random.

Let $\{|\alpha_j\rangle \mid j = 1, \cdots, n\}$ be an orthonormal basis. Suppose the state of the system is $|\alpha_i\rangle$. Let $\{|\beta_i\rangle \mid i = 1 \cdots, n\}$ be another orthonormal basis. Then if we do a maximal test with respect to $\{|\beta_i\rangle\}$ then the probability of obtaining result $|\beta_j\rangle$ is $p_{ij} = |\langle\alpha_i|\beta_j\rangle|^2$. This can also be written as $\mathtt{Tr}((|\alpha_i\rangle\langle\alpha_i|)(|\beta_i\rangle\langle\beta_i|))$ where the *trace* Tr is the sum of the diagonal elements of a square matrix, which is independent of the representation.

The p_{ij} are called the transition probabilities. Let $U = (u_{ij} = \langle\alpha_i|\beta_j\rangle)$ be a matrix. Then U is unitary. It is the matrix which expresses the change of basis and $p_{ij} = |u_{ij}|^2$. We thus see that the transition probability matrix is doubly stochastic, i.e., $\sum_i p_{ij} = \sum_j p_{ij} = 1$. But an arbitrary doubly stochastic matrix (for example appearing in classical Markhov processes) may not correspond to transition probability matrix in quantum theory be-

cause it may not satisfy $p_{ij} = |u_{ij}|^2$ for some unitary $U = (u_{ij})$. Such matrices (p_{ij}) are called orthostochastic. Thus the p_{ij} must satisfy some relations. We thus see an important difference with classical probability theory. We cannot make arbitrary probability assignments (satisfying of course the usual probability constraints) but the probabilities must satisfy certain nonlinear inequalities. This is also true of the probabilities p_i introduced earlier.

The state of a system can change by two kinds of operations. The first is measurement which is a generalization of the concept of maximal tests. A (projective) measurement is a set of projection operators $\{P_{m_i}\}$ such that $\sum_i P_{m_i} = I_n$, the identity operator in dimension n. The $\{m_i\}$ are the possible outcomes of the measurement. The probability for i^{th} outcome is $p_i = \mathtt{Tr}(P_{m_i}\rho)$. If the outcome is known to be m_i then the postmeasurement state is $\rho_i = \frac{P_{m_i}\rho}{p_i}$. Sometimes we know only that a measurement has been done not its outcome. Then the state after the measurement is $\sum_i p_i\rho_i = \sum_i P_{m_i}\rho$. We observe that for a state ρ the map $\mathtt{Tr}_\rho(P_i) \equiv \mathtt{Tr}(\rho P_i)$ defines a probability on the set of orthonormal projections $\{P_i\}$. The second operation is unitary evolution. Simply put it means that if ρ is the state then after an operation by a unitary matrix U the state is $U\rho U^{-1}$. The physical basis of this operation is more subtle and I refer to standard texts on quantum theory. I only mention that the "quantum gates" are unitary operations.

Given two systems S_1 and S_2 with state spaces H_1 and H_2 resp. the combined system has the state space $H_1 \otimes H_2$ which is the vector space of linear combinations $\sum_i c_i(|\alpha_i\rangle \otimes |\beta_i\rangle), |\alpha_i\rangle \in H_1$ and $|\beta_i\rangle \in H_2, c_i$ complex such that for any number c $(c|\alpha\rangle) \otimes |\beta\rangle = |\alpha\rangle \otimes (c|\beta\rangle) = c(|\alpha\rangle \otimes |\beta\rangle)$. If $\{|\alpha_i\rangle\}_{i=1}^m$ and $\{|\beta_j\rangle\}_{j=1}^n$ are bases for H_1 and H_2 resp. then $\{|\alpha_i\rangle \otimes |\beta_j\rangle\}$ form a basis for $H_1 \otimes H_2$. The inner product is defined by $\langle\alpha| \otimes \langle\beta| \, |\gamma\rangle \otimes |\delta\rangle = \langle\alpha|\gamma\rangle\langle\beta|\delta\rangle$. Thus $H_1 \otimes H_2$ is a Hilbert space of dimension mn. We can extend to tensor product of operators on H_1 and H_2. Let T_1(resp. T_2) be an operator on H_1(resp. H_2). Since the product vectors $\{|\alpha\rangle \otimes |\beta\rangle\}$ form a

basis of the space $H_1 \otimes H_2$. We define $T_1 \otimes T_2(|\alpha\rangle \otimes |\beta\rangle) = T_1 |\alpha\rangle \otimes T_2 |\beta\rangle$ and extend by linearity. In terms of matrices, the tensor product of a an $m \times n$ matrix and $p \times q$ matrix is a matrix of order $mp \times nq$, defined as follows which, of course, corresponds to the matrix representation of the operators. Let

$$A = \begin{pmatrix} a_{11} & \cdots & \cdots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & \cdots & \cdots & a_{mn} \end{pmatrix} \quad B = \begin{pmatrix} b_{11} & \cdots & \cdots & b_{1q} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ b_{p1} & \cdots & \cdots & b_{pq} \end{pmatrix} \text{ then}$$

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & \cdots & a_{1n}B \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1}B & \cdots & \cdots & a_{mn}B \end{pmatrix}$$

If $\{P_{m_i}\}$ and $\{P_{n_j}\}$ are two measurements on S_1 and S_2 resp. with outcomes $\{m_i\}$ and $\{n_j\}$ then $\{P_{m_i} \otimes P_{n_j}\}$ constitutes a measurement on the combined system $S_1 + S_2$ with outcomes $\{m_i n_j\}$. Further if the composite system is in a product state $\rho_1 \otimes \rho_2$ then the probability of outcome $m_i n_j = p_i^{(1)} p_j^{(2)}$ where $p_i^{(i)}$ (resp. $p_j^{(2)}$) is the probability of outcome m_i (resp. n_j) if we choose to measure only S_1 (resp. S_2). This is a consequence of $\mathtt{Tr}(A \otimes B) = \mathtt{Tr}(A)\mathtt{Tr}(B)$. If the state is not a product state or a convex combination of product states then it is said to be *entangled.* For details the reader may consult books on quantum theory mentioned earlier.

Chapter 4

Logics for Quantum Probability

This chapter is central to the whole work. In this chapter I present the logics for reasoning about quantum systems specifically designed for quantum computation and information **QCI**.

4.1 Introduction

In any probabilistic theory one has to deal with real numbers. Hence for the general formalization of such theories we have to include a formal presentation of real closed field **RCF**. In the case of quantum theory however, the situation is somewhat more complex. Quantum probabilities are given by the squared modulus of complex numbers depending on the quantum state. That is, the probability measure is state dependent in a definitive way and the state is a *complex* vector. Thus any formalization of quantum probabilities would require a formal theory of the pair: real closed field and its algebraic closure (complex field). I present one such formalization and prove some model theoretic properties crucial to the main logic presented in this chapter.

4.2 The theory $\mathbb{RC}$

In this section we develop a theory which incorporates the theory of real closed fields and their algebraic closures, an algebraically closed field. The theory of real and algebraically closed fields have been well studied, both from model theoretic [Hod97] and algorithmic viewpoints [BPR03]. In fact, a large part of model theory *is* the formal study of such fields. In the context of quantum theory we need complex numbers because the states are defined as unit vectors in a *complex* Hilbert space *and* real numbers because probabilities must be real non-negative numbers. Further, since probabilities arise out of complex state vectors, we have to use the fact that the field of real numbers is a subfield of complex numbers. The aim of this section is to formalize this structure. The formal treatment of the corresponding structures may be found, for example, in van der Waerden's classic text on algebra [Wae53]. But we would also like to develop the proof theory and study some of its properties. The basic idea is the formalization of standard construction of the complex numbers as pairs of real numbers. The resulting theory is called $\mathbb{RC}$. The theory has models other than the field of complex numbers. We note that the complex field as an extension of real fields is the standard approach to introduction of complex numbers. Our aim is to formalize this approach and study some general and algorithmic properties of the models.

The nonlogical symbols of the first order language $\mathcal{L}_{RC}$ are given below. I assume the standard *logical* symbols including equality [Sho67].

1. **Function Symbols:** The function symbols include the usual binary symbols '$+$' and '$\cdot$' in the infix notation. We also have the defined symbol $\overline{z}$ for complex conjugate of z.

2. **Predicate Symbols:** It is understood that $\mathbb{RC}$ is a theory with equality with the usual axioms and rules for equality. The only nonlogical symbols are '$<$' and 'R'. The former is written in infix notation and

R is a unary predicate, the intended interpretation of R that the corresponding term is real. We also use x^n as a shorthand for $\overbrace{x \cdots x}$ and $s \neq t$ for $\neg(s = t)$.

3. **Constant Symbols:** The four principal constants are $0, 1, -1$ and i. For convenience, I include, for each positive integer k, symbols k^{-1} for the inverses, and for each pair of given positive integers (n, k) the k^{th} roots $n^{1/k}$, $k \neq 0$. The constant $n^{1/2}$ will be written $\sqrt{n}$.

Note that one may define all but the principal constants within the theory. For example 2 is a symbol for 1+1. Similarly, we define all positive integers as constants. It is important that we have an efficient representation of the integers since the size of the input depends on this representation. First the positive integers are defined as follows. The standard procedure is to represent them as $1 + 1 + \cdots + 1$(unary representation). However, it is more efficient to represent them as a k-ary expansion, where $k > 1$ is a positive integers such that all integers up to and including k are assumed to be defined. For example, for $k = 2$ we first define $2 = 1 + 1$. Then consider the binary expressions

$$a_0 \cdot 1 + a_1 \cdot 2 + a_2 \cdot (2 \cdot 2) + a_3 \cdot (2 \cdot 2 \cdot 2) + \cdots$$

This is a term of $\mathbb{RC}$. In the complexity theory of real or algebraically closed field we often have to deal with multivariate polynomials with integer(or rational) coefficients. The size of the polynomial may then defined to be proportional to the sum of the bit lengths of the coefficients [BKR86].

The inverses and roots of the positive integer constants may be eliminated as defined constants. I explain briefly the notion of extension of a theory by definition [Sho67]. Suppose we have two theories $\mathbf{T}$ and $\mathbf{T}'$ whose languages are identical except that $\mathbf{T}'$ has an extra k-ary function symbol f and an extra axiom

$$(f(x_1, \cdots, x_k) = y \Leftrightarrow \exists y D(x_1, \cdots, x_k, y)) \wedge$$
$$D(x_1, \cdots, x_k, y) = D(x_1, \cdots, x_k, y') \Rightarrow y = y'$$

where D is a formula of $\mathbf{T}$ in which no variables other than $x_1, \cdots, x_k$ and y are free. The formula on the right simply asserts the existence of a unique y for given $x_1, \cdots, x_k$ such that D is true. Then, $\mathbf{T}'$ is a conservative extension of $\mathbf{T}$ [Sho67]. This means that any formula in the language of $\mathbf{T}$ that is provable in $\mathbf{T}'$ is also provable in $\mathbf{T}$. Moreover, due to the above formula and results on equivalent formulas [Sho67] any formula containing f can be replaced by a formula which does not contain it, that is, a formula $\mathbf{T}$. As constants are treated as 0-ary function symbols the defining formula for a constant is of the form $(\exists y D(y)) \wedge D(y') \Rightarrow y = y'$.

The axioms of the theory $\mathbb{RC}$ are as follows. First the field axioms.

FL1 $(x + y) + z = x + (y + z)$

FL2 $x + 0 = x$

FL3 $x + (-1 \cdot x) = 0$

FL4 $x + y = y + x$

FL5 $(x \cdot y) \cdot z = x \cdot (y \cdot z)$

FL6 $x \cdot 1 = x$

FL7 $x \neq 0 \Rightarrow \exists y(x \cdot y = 1)$

FL8 $x \cdot y = y \cdot x$

FL9 $x \cdot (y + z) = (x \cdot y) + (x \cdot z)$

FL10 $0 \neq 1$

Next the axioms of a real field and for the defined symbols.

R1 $R0 \wedge R1$

R2 $Rxy \Rightarrow R(x + y) \wedge R(x \cdot y)$

R3 $x < y \Leftrightarrow Rx \wedge Ry \wedge \exists z(Rz \wedge z \neq 0 \wedge y = x + z^2)$

For each positive integer n and k,

R4 $(Rx_1x_2 \ldots x_n \wedge x_1^2 + x_2^2 + \ldots + x_n^2 = 0) \Rightarrow \bigwedge_i x_i = 0$

De1 $n/k \cdot k = n$

De2 $n^{1/k} \geq 0 \wedge (n^{1/k})^k = n$

De3 $i^2 + 1 = 0$

I use shorthand $Rx_1x_2\ldots x_n$ for $Rx_1 \wedge Rx_2 \wedge \ldots \wedge Rx_n$, $-x$ for $(-1)\cdot x$. The standard notation and the binding rules for addition and multiplication are used throughout. **De1** and **De2** simply express the obvious property of the notation for fractions and roots respectively. **R1** and **R2** capture the fact that the reals form a subring of the complex numbers. It will soon be shown that it is actually a subfield. The formula **R3** is the defining axiom for $<$ and **R4** expresses the fact that 0 can not be expressed as a sum of squares unless all summands are zero. I have omitted the transitive property of "¡" since it can be derived from the other axioms (see below). The following 2 axioms express the notion of complex numbers as pairs of reals and the fact that the complex field is algebraically closed. Note that term of **RC** with exactly one variable is a polynomial in that variable and a general term may be considered as polynomial with "variable" coefficients.

CR $\exists xy(Rx \wedge Ry \wedge z = x + i\cdot y)$

For each positive integer n we have

AC $y_n \neq 0 \Rightarrow \exists x(y_n\cdot x^n + y_{n-1}\cdot x^{n-1} + \ldots + y_1\cdot x + y_0 = 0)$.

Some simple consequences of the axioms of $\mathbb{RC}$ are given by

Lemma 1 *The following are theorems of* $\mathbb{RC}$ *For each positive integer* n

$$Rx_1x_2\ldots x_n \Rightarrow x_1^2 + \ldots + x_n^2 \neq -1 \tag{4.1}$$

$$(x + i\cdot y = x' + i\cdot y' \wedge Rxyx'y') \Leftrightarrow x = x' \wedge y = y' \tag{4.2}$$

$$Rx \wedge x \neq 0 \Rightarrow \exists y(Ry \wedge x\cdot y = 1) \tag{4.3}$$

$$Rx \Rightarrow \exists y(Ry \wedge (x = y^2 \vee -x = y^2)) \tag{4.4}$$

$$\forall xyz(x < y \wedge y < z \Rightarrow x < z) \tag{4.5}$$

Moreover, $\neg Ri$ *and if* **a** *is variable free term which does not contain* i *then* R**a**.

Proof: I sketch informal proofs. The first formula follows from the fact that $1 = 1^2$ (**FL6**) and $Rx_1x_2 \ldots x_n \wedge x_1^2 + \ldots + x_n^2 = -1 \Leftrightarrow Rx_1x_2 \ldots x_n \wedge x_1^2 + \ldots + x_n^2 + 1 = 0$. Hence, from **R4** for all i, $x_i = 0$ and 1=0. The last formula contradicts **FL10**. The definition of real fields is usually in given in the form of formula 4.1.

The second formula formalizes the notion that the 'real' and 'imaginary' parts of a complex number are unique. It will be sufficient to show that $0 = x + i \cdot y \wedge Rxy \Rightarrow x = 0 \wedge y = 0$. The latter follows immediately from the fact that $x + i \cdot y = 0 \Rightarrow x = -i \cdot y$ which in turn implies that $x^2 + y^2 = 0$ (using **De3**). Hence from **R4** it follows that $x = y = 0$.

Now for the third formula it follows from **FL7** and **CR** that $z \neq 0 \Rightarrow \exists xy(Rxy \wedge z \cdot (x + i \cdot y) = 1)$. From this we deduce that $(z \cdot x - 1)^2 + (z \cdot y)^2 = 0$ and thus from **R4**, $Rz \wedge z \neq 0$ implies that $Rx \wedge z \cdot x = 1$. The advantage of **R4** over the usual 4.1 [Sho67] is that with the former we have to only postulate that the reals form a subring and the fact that they also constitute a subfield can be deduced from the rest.

The formula 4.4 can be inferred as follows. From **CR** and **AC** it follows that $\exists xy(Rxy \wedge z = (x + i \cdot y)^2)$. Expanding the right side $z = x^2 - y^2 + 2 \cdot i \cdot x \cdot y$. If Rz then $2 \cdot i \cdot x \cdot y = 0$. This follows from arguments similar to those used above. Hence $x = 0 \vee y = 0$ and the assertion follows. The transitivity relation 4.5 is easily derived from the identity $a^2 + b^2 = [(a - b)^2 + (a + b)^2]/2$.

Using **De3** we get $\neg Ri$. Next, note that the formula $x > 0 \wedge y > 0 \Rightarrow x \cdot y > 0$. This follows easily from **R3** and the field axioms(we simply note that $(xy)^2 = x^2y^2$. The fact that for any non-zero term the inverse is unique is a simple consequence of the field axioms. Hence, it follows from **De1** and Eq.(4.3) that $R(n/k)$ and $R(n^{1/k})$ follows from **De2**. The general case for a variable term without i will follow by induction from **R2**. □

The fact that the real elements form a real closed field is expressed by the following lemma. First, we introduce the defined function symbol $z \rightarrow \overline{z}$

for complex conjugation and the axiom

De5 $Rxy \Rightarrow (z = x + i \cdot y \Leftrightarrow \overline{z} = z = x - i \cdot y)$

Note that, $\overline{(z_1 + z_2)} = \overline{z_1} + \overline{z_2}$ and $\overline{(z_1 \cdot z_2)} = \overline{z_1} \cdot \overline{z_2}$.

Lemma 2 *For each odd positive integer n, the formula*

RF $Rx_0x_1 \ldots x_n \wedge x_n \neq 0 \Rightarrow \exists y(Ry \wedge x_n \cdot y^n + \ldots + x_1 \cdot y + x_0 = 0.)$

is a theorem of $\mathbb{RC}$. In other words, every polynomial of odd degree with real coefficients has a real root.

Proof: To prove this first observe that a polynomial of degree n over any algebraically closed field has exactly n roots (counting repetitions). The standard elementary proof [Wae53] can be formalized using only the field axioms **FL1** through **FL10** and **AC**. We need **AC** to show that for any polynomial $p(x) = y_n x^n + \ldots + y_0$, $\forall x(p(x) = 0) \Leftrightarrow \wedge_i y_i = 0$. This can be proved by induction on n. Thus, $\forall x(p(x)) = 0) \Rightarrow \forall x(p(x) + 1 \neq 0$. Hence **AC** implies that $y_n = 0$ and the assertion follows by induction. As a consequence, we have: two polynomials are equal iff all the coefficients are equal. In particular, two polynomials of different degree can not be equal. It follows from the division algorithm that, if a is a root of a polynomial $p(x)$ then $x - a$ divides $p(x)$. Then one proves that a polynomial of degree n has at most n roots. For example, the formula for degree=2 is $\forall y_2 y_1 y_0 \exists x_1 x_2 \forall x(y_2 x^2 + y_1 x + y_0 \Leftrightarrow x = x_1 \vee x = x_2)$. Now using complex conjugation defined above it follows that if $z = x + i \cdot y$ then $Rz \Leftrightarrow z = \overline{z}$. This may be seen as follows. In one direction, $z = \overline{z}$ implies $y = 0$ and $z = x$ and hence Rz. Conversely, from the identity $((z^2 - x^2 + y^2)^2 + 4x^2y^2 = 0)$ and Rz we have $4x^2y^2 = 0$. Hence, $x = 0$ or $y - 0$. But, if $x = 0$ then $z = iy$ and $z^2 + y^2 = 0$ which implies $z = x = 0$. If $p(x)$ has real coefficients then $p(z) = 0 \Leftrightarrow p(\overline{z}) = 0$. That is nonreal roots come in distinct pairs. Hence if n is odd, then a real polynomial of degree n has a real root. We note that the informal arguments above can be formalized. □

The next lemma is useful.

Lemma 3 *The relation $<$ is a total linear order on values satisfying R. That is, it is irreflexive, asymmetric, and transitive and the following formula is a theorem of* **RC**.

$$Rxy \Rightarrow x < y \vee x = y \vee y < x \tag{4.6}$$

Proof: The formula $\neg(x < x)$ (irreflexivity) is a theorem. It is deduced from $x = x + z \Rightarrow z = 0$ and **R3**. Using the formula 4.4 and **R4** we deduce

$$Rxy \Rightarrow \exists z(Rz \wedge z^2 = x^2 + y^2)$$

Transitivity ($x < y \wedge y < z \Rightarrow x < z$) is an easy consequence of this and **R3**. The formula $x < y \Rightarrow \neg(y < x)$ (assymetry) is a consequence of transitivity and irreflexivity.

Observe that $Rxy \Rightarrow x < y \Leftrightarrow x - y < 0$ is a theorem. Hence, to prove 4.6, it is sufficient to prove $Rx \Rightarrow x < 0 \vee x = 0 \vee 0 < x$ But this follows from 4.4 and the axiom **R3**. □

A theory **T** is said to admit elimination of quantifiers if for any formula A in **T** there is an open (i e. quantifier free) formula B such that $A \Leftrightarrow B$ is provable in **T**. Recall that a formula F in **T** is consistent iff $\neg F$ is not a theorem and the theory is consistent if there is a consistent formula. A theory **T** is a *complete* theory if every closed formula **A** is decidable. That is, **A** or $\neg$**A** is a theorem. Complete theories have many pleasant properties [Sho67]. Some of them are listed below for latter use.

1. For any two models of a complete theory **T** the same formulas are valid.

2. To prove that a formula F is a theorem it suffices to show its validity in any model.

Actually, each of the above properties characterizes complete theories. As a consequence of the second property we may use any method to prove the

validity of some a formula in some model. For example, since the theory of real closed fields **RCF** is complete we may use analytical tools (e.g. differentiation and integration) in the field of real numbers to prove that some formula of **RCF** is valid for real numbers. Then we are guaranteed that it is a theorem of **RCF**.

The first principal result of this section is the following.

Theorem 4 *The theory $\mathbb{RC}$ admits elimination of quantifiers. It is a complete theory.*

Proof: To prove the theorem one may follow the standard techniques used in [Sho67]. However, a shorter proof is possible. First observe that since the field of complex numbers is a model of $\mathbb{RC}$ it is consistent. Further, the atomic formulas of the theory are $p(z_1,\ldots,z_n) = 0$ and $q(z_1,\ldots,z_m) < 0$ where p and q are polynomials whose coefficients are variable free terms in $\mathbb{RC}$. Let **RCF** be the theory of real closed fields whose axioms are **FL1-FL10**, **R1-R3**, and **RF**. Using **CR** the first set of formulas are equivalent to a pair of formulas in **RCF**. That is, there are real terms $\mathbf{t}_1$ and $\mathbf{t}_2$ in the latter such that $p(x_1,\ldots,x_n) = 0 \Leftrightarrow \exists x_1\cdots x_n y_1\cdots y_n \wedge_i (z_i = x_i + iy_i \wedge Rx_iy_i) \wedge \mathbf{t}_1(x_1\cdots x_n y_1\cdots y_n) = 0 \wedge \mathbf{t}_2(x_1\cdots x_n y_1\cdots y_n = 0)$. We may construct t_1 and t_2 as follows. Each coefficient in p is a term built out of the constants of $\mathbb{RC}$. The only constant which is not real is i. We substitute $z_i = x_i + iy_i$ and write $p = t_1(x_1\cdots x_n y_1\cdots y_n) + it_2(x_1\cdots x_n y_1\cdots y_n)$ using $i^2 = -1$ with t_1 and t_2 real. Hence from the fact that **RCF** admits quantifier elimination we conclude that so does **RC**. The second type of atomic formulas $q < 0$ implies Rq by **R3**. By extracting the (unique) real and imaginary parts q_1 and q_2 of q we see that it is equivalent to a pair of formulas $q_1 < 0$ and $q_2 = 0$. The only point we have to consider here is that $\mathbb{RC}$ has some extra constants. For example, consider the real constants n/k, $k \neq 0$. First, the constants k may be replaced by terms which represent the integer k. As $k > 0$ and it is easily shown that the multiplicative inverse is unique we can prove $n/k = n \cdot k^{-1}$. Similarly, the constants defining the positive

real roots $n^{1/k}$ may be eliminated. Thus, if k is odd by the real closure (see **RF**) $\exists x(Rx \wedge x > 0 \wedge x^k = n)$, since n is a positive. The identity,

$$x^k - y^k = (x-y)(x^{k-1}y + x^{k-2}y + \ldots + y^{k-1})$$

can be used to show that the root x is unique. If k is even, then we write $k = k'2^r$, k' odd , and first show the existence of a unique positive real number $x^{1/k'}$ as above. Then by successive application of **R3** we get a unique x such that $x > 0 \wedge x^k = n$. Hence, the constants $n^{1/k}$ may be eliminated by the theorem on extension by definition (see Ch.3 of [Sho67]). We conclude that any formula of $\mathbb{RC}$ is equivalent to a finite set of formulas in the theory **RCF**. The latter, is known to be complete [Sho67]. Hence, $\mathbb{RC}$ admits elimination of quantifiers and is complete. □

Finally, we introduce a defined function symbol for square root. The problem is, for every complex number other than 0 there are two square roots and we must make a consistent choice. Let

$$\mathbf{D}(x,z) \stackrel{\text{def}}{=} x^2 = z \wedge (x+\overline{x}) > 0 \vee ((x+\overline{x}) = 0 \wedge (-i)\cdot(x-\overline{x}) \geq 0). \quad (4.7)$$

The formula **D** simply expresses that x is a square root of z and if its real part is not zero then x is the root with positive real part, otherwise in case roots are pure imaginary, choose the one which is positive real multiple of i.

Lemma 4 *The following are theorems of* $\mathbb{RC}$.

$$\exists x \mathbf{D}(x,z)$$

and

$$\mathbf{D}(x,z) \wedge \mathbf{D}(x',z) \Rightarrow x = x'$$

.

Proof: The proof is outlined below. The standard notation $\vdash_T \mathbf{A}$ meaning that **A** is a theorem of the theory T. Since we are dealing with theory $\mathbb{RC}$ throughout this section, it is not mentioned below. That is, all the formulas are theorems of $\mathbb{RC}$.

1. $\vdash \exists x(x^2 = z)$. Follows easily from **AC** .

2.
$$\vdash x^2 = (-x)^2$$
This is an easy consequence of $(-x)(-x)+(-x)(x) = (-x)(-x+x) = (-x)0 = 0$ and the formulas $x^2 + (-x)(x) = x(x + (-x)) = 0$, using the axioms **FL2-FL9**.

3.
$$\vdash (x+\overline{x}) > 0 \ \vee \ ((-x)+\overline{(-x)}) > 0 \vee$$
$$((x+\overline{x}) = 0 \ \wedge \ (-i)(x-\overline{x}) \geq 0 \vee ((-i)((-x)-\overline{(-x)})) \geq 0$$
This formula follows from the fact that $x+\overline{x}$ and $(-i)(x-\overline{x})$ are real and the lemma 3. From the above formulas it follows that:

4. $\vdash \exists x\mathbf{D}(x,z)$

The second formula of the lemma expresses the uniqueness of the chosen square root. The uniqueness of $\sqrt{x}$ is a consequence of the fact except for $z = 0$ only 2 square roots x and $-x$ are possible for

$$x^2 = y^2 \Leftrightarrow (x+y)(x-y) = 0 \Leftrightarrow x = y \vee x = -y. \tag{4.8}$$

□

It now follows that if one adds a function symbol $\sqrt{}$ and the defining axiom $\sqrt{z} = x \Leftrightarrow \mathbf{D}(x,z)$ then the resulting theory -call it (temporarily) $\mathbb{RC}_{\sqrt{}}$ is essentially equivalent to $\mathbb{RC}$. In model theoretic language, for any model $\mathcal{M}$ of $\mathbb{RC}$ there is (unique) model $\mathcal{M}'$ of $\mathbb{RC}_{\sqrt{}}$ which is obtained from $\mathcal{M}$ by the addition of the square root function. We continue to call the extended theory $\mathbb{RC}$ instead of $\mathbb{RC}_{\sqrt{}}$. In the proof of the Theorem 4, I used a standard technique in the algebraic theory of complex numbers: splitting a complex term into real and imaginary parts. One can apply this to the

satisfiability problem in $\mathbb{RC}$. For any formula $\mathbf{F}$, the length of the formula, $|\mathbf{F}|$ is the number of symbols in $\mathbf{F}$. An algorithm for the reduction of $\mathbf{F}$ into $\mathbf{F}'$ is outlined in the proof of the next lemma which is the second principal result of this section.

Lemma 5 *For any formula* $\mathbf{F}$ *of* $\mathbb{RC}$ *there is formula* $\mathbf{F}'$ *of* **RCF** *such that* $\mathbf{F}$ *satisfiable if and only if* $\mathbf{F}'$ *is satisfiable. Moreover,* $|\mathbf{F}'|$ *is of polynomial order in* $|\mathbf{F}|$.

Proof: We define a map $T : \mathbf{Fm}(\mathbb{RC}) \to \mathbf{Fm}(\mathbf{RCF})$ from the set of formulas of the theory $\mathbb{RC}$ to the theory **RCF**. We further introduce a map $\mathtt{t}(\mathbb{RC}) \to \mathtt{t}(\mathbf{RCF}) \times \mathtt{t}(\mathbf{RCF})$ from the set of *terms* of $\mathbb{RC}$ to a *pair* of terms of **RCF** denoted by $t \to (T_r(t), T_i(t))$. Intuitively, $T_r(t)$ and $T_i(t)$ are respectively the real and imaginary parts of the term $\mathtt{t}$. The terms of $\mathtt{t}$ are constructed by recursive application of the function symbols $'+'$ and $'\cdot'$. We assume that all constant terms have been reduced to the form $a + \imath b$ where a and b are rational integers. It would be convenient to first reduce the given $\mathbb{RC}$ formula to a standard form. Thus, define the function $N : \mathbf{Fm}(\mathbb{RC}) \to \mathbf{Fm}(\mathbb{RC})$ as follows. First, an auxiliary function $\mathtt{t}(\mathbb{RC}) \to \mathtt{t}(\mathbb{RC})$. We denote this map by $t \to h_t$ and define

$$h_t = \begin{cases} t \text{ if } t \text{ is a variable or constant} \\ z_t \text{ is a new variable not in } \mathbf{F}, \text{ otherwise} \end{cases}$$

Let $\mathbf{F}(t)$ be atomic, that is, of the form $\mathbf{F}(t) = 0$ or $\mathbf{F}(t) < 0$ or Rt. If $t = t_1 \circ t_2$, where $\circ$ is either $+$ or $\cdot$ or $-$ and neither t_1 nor t_2 is a variable or constant then let

$$N(\mathbf{F}(\mathtt{t})) \equiv \exists z_{t_1} z_{t_2} N(\mathbf{F}(z_{t_1} \circ z_{t_2})) \wedge N(t_1 - z_{t_1} = 0) \wedge N(t_2 - z_{t_2} = 0)$$

If one of them, say t_1, is a variable (or constant) and t_2 is not

$$N(\mathbf{F}(\mathtt{t})) \equiv \begin{cases} \exists z_{t_2} N(\mathbf{F}(t_1 \cdot z_{t_2})) \wedge N(t_2 - z_{t_2} = 0) \text{ if } \circ \text{ is } \cdot \\ F(z_{t_2} \circ t_1) \wedge (t_2 - z_{t_2} = 0) \text{ if } \circ \text{ is } + \text{ or } - \end{cases}$$

Similarly for the case of t_2. It follows that if both are variables, say $t = z \circ z'$ then $N(F(z \circ z')) = F(z \circ z')$. It is clear that the recursive application of N yields an $\mathbb{RC}$ formula such that the terms of each atomic subformula of the image $N(\mathbf{F})$ is of the form $\mathbf{F}'(t)$ where $t = c \circ z$ or $t = z \circ z'$, c a constant and z, z' variables. We call such terms elementary. Moreover, the length of $N(\mathbf{F})$ is linear in $|\mathbf{F}|$. It is also clear from equality axioms that $\mathbf{F}$ is satisfiable(valid) if and only if $N(\mathbf{F})$ is satisfiable(valid). Moreover, in the case of satisfiability, if an elemantary formula $\mathtt{A}(z)$, with a free variable z is true in some interpretation then $\mathbf{N}(\mathtt{A})(z)$ is true in the *same* interpretation. Note that all the new variables introduced are existentially quantified. We extend N to general formulas as follows.

1. If $\mathbf{F}$ is of the form $\mathbf{F}_1 \wedge \mathbf{F}_2$ then we let $N(\mathbf{F}) = N(\mathbf{F}_1) \wedge N(\mathbf{F}_2)$. Now $\mathbf{F}_1 \wedge \mathbf{F}_2$ is true in some interpretation iff $\mathbf{F}_1$ and $\mathbf{F}_2$ are true in the same interpretation. If $\mathbf{F}_1$ and $\mathbf{F}_2$ are true for some interpretaion then the corresponding $N(\mathbf{F}_1)$ and $N(\mathbf{F}_2)$ hold in the same interpretation. The converse also holds. Hence, by induction $N(\mathbf{F}_1) \wedge N(\mathbf{F}_2)$ is satisfiable if and only if $\mathbf{F}_1 \wedge \mathbf{F}_2$ is satisfiable.

2. If $\mathbf{F} \equiv \neg\mathbf{F}'$ then from the definition of N for atomic formulas and using induction we can show that $N(\mathbf{F}') = \exists z_{t_1} \ldots z_{t_k} \mathbf{F}'(z_{t_1}, \ldots, z_{t_k}) \wedge (t_1 = z_{t_1} \wedge \cdots \wedge t_k = z_{t_k})$. Where $z_{t_1}, \ldots, z_{t_r}$ are the *new* variables introduced for the terms appearing in $\mathbf{F}$. We let

$$N(\mathbf{F}) \equiv \exists z_{t_1} \ldots z_{t_k} (t_1 = z_{t_1} \wedge \cdots \wedge t_k = z_{t_k}) \neg\mathbf{F}'(z_{t_1}, \ldots, z_{t_k})$$

   That is, the new formulas introduced are not negated. If $\mathbf{F}$ is satisfiable then $\mathbf{F}'$ is not valid. Hence, by induction $N(\mathbf{F}')$ is not valid and the right hand side of the above equation is satisfiable.

3. If $\mathbf{F} \equiv \exists z \mathbf{F}'$ then $N(\mathbf{F}) \equiv \exists z N(\mathbf{F}')$. Similarly for the universal quantifier. Again it is obvious that $\mathbf{F}$ is satisfiable if and only if $N(\mathbf{F})$ is satisfiable.

Next, we define T. First define T for elementary terms

$$\begin{aligned} T(k) &= (k,0), \text{ and } T(i) = (0,1), \ k \text{ an integer} \\ T(z) &= (z_r, z_i), \quad T(t_1 + t_2) = (T_r(t_1) + T_r(t_2), T_i(t_1) + T_i(t_2)), \\ T(t_1 \cdot t_2) &= (T_r(t_1)T_r(t_2) - T_i(t_1)T_i(t_2), T_r(t_1)T_i(t_2) + T_i(t_1)T_r(t_2)) \end{aligned} \tag{4.9}$$

where t_1 and t_2 are variables or constants.

Let $\mathbf{F}$ be an atomic formula whose terms are elementary. Then $\mathbf{F}$ must be of the form $t = 0, t > 0$ or Rt. We define $T(\mathbf{F})$ in each case.

$$\begin{gathered} T(t=0) \equiv T_r(t) = 0 \wedge T_i(t) = 0 \quad T(t>0) \equiv T_r(t) > 0 \wedge T_i(t) = 0 \\ T(Rt) \equiv T_i(t) = 0 \end{gathered} \tag{4.10}$$

The general case is easily handled by induction. Assume first that $\mathbf{F}$ contains only elementary terms.

1. If $\mathbf{F} \equiv \mathbf{F}_1 \wedge \mathbf{F}_2$ then $T(\mathbf{F}) \equiv T(\mathbf{F}_1) \wedge T(\mathbf{F}_2)$.

2. If $\mathbf{F} \equiv \neg \mathbf{F}_1$ then $T(\mathbf{F}) \equiv \neg T(\mathbf{F}_1)$. The definition of T has not involved any quantification over new variables(so far!).

3. If $\mathbf{F} \equiv \exists z \mathbf{F}'$ then let $T(\mathbf{F})$ be obtained as follows. If the variable z occurs in $\mathbf{F}$ then let $T(\mathbf{F}) \equiv \exists z_r z_i T(\mathbf{F}')$ where $T(z) = (z_r, z_i)$. Similarly we define $T(\mathbf{F})$ for universally quantified formulas.

Finally, for a general formula $\mathbf{F}$ we define

$$T(\mathbf{F}) \equiv T(N(\mathbf{F}))$$

Since $N(\mathbf{F})$ consists of elementary terms only $T(N(\mathbf{F}))$ is already defined. Moreover, as $N(\mathbf{F}) = \mathbf{F}$ if the latter consists of elementary terms only the two definitions of T in this case are consistent.

We may easily show by induction that $T(\mathbf{F})$ is satisfiable if and only if $\mathbf{F}$ is satisfiable. Next, to deal with the square root symbol we use the defining formula 4.7 for the square root function to reduce a formula $\mathbf{F}$ in $\mathbb{RC}$ to a

an equivalent formula $\mathbf{F}_1$ free of square roots. Then we follow the steps in the above algorithm to obtain $\mathbf{F}_1$.

The proof is complete □

Note that we could have used the usual ansatz for converting a complex equation to a pair of real equation to define the function T. That is, replace each complex variable z by $x + iy$ for a pair of real variables x, y. But then the size of the resulting formula may be exponential. Take the simple example of the equation $z^n = 1$. But the current reduction algorithm has a drawback that we introduce several auxiliary variables increasing the time complexity of the decision problem. It can be modified so that the number of auxiliary variables is minimized.

Theorem 5 *The satisfiability of a formula in* $\mathbf{A}$ *in* $\mathbb{RC}$ *may be decided in exponential space. If* $\mathbf{A}$ *is quantifier free then satisfiability may be decided in polynomial space.*

The theorem may be proved using the complexity bounds derived by Ben-Or, Kozen and Reif [BKR86] for a general formula, and Canny [Can88] for the quantifier free case. I conclude this section with some remarks. Since the theory $\mathbb{RC}$ is a complete theory if a formula of $\mathbb{RC}$ is valid in any model then it is valid in all models and hence a theorem of $\mathbb{RC}$. Conversely, if a formula is not satisfiable in some model then it is not satisfiable in any model. The field of complex numbers considered as an algebraic extension of the field of real numbers is a model of $\mathbb{RC}$. In the subsequent sections the elements of any model of $\mathbb{RC}$ is referred to as complex numbers. From the above remarks it is seen that any statement using such reference is actually a statement about any model of $\mathbb{RC}$.

4.3 A Logic for Quantum Probability

This chapter is devoted to a logic for reasoning about quantum probabilities in a fixed dimension. This means that we treat the system as a whole and ignore the internal structure of its components. A simple physical example would be the hydrogen atom. One may consider, for example, the spin states of the whole atom ignoring the fact that it is composed of an electron and a proton. Another simplistic example is the 2-qubit system. One simply treats it as a single 4-dimensional system. We need the basic language for a fixed dimension to build more complex languages for composite systems in the next chapter. The section is divided into several subsections. The first two deal with syntax and semantics followed by a subsection on examples. The last two subsections deal with axiomatics and decidability and complexity properties of the logic.

4.3.1 The Language $\mathcal{L}_n(P)$ and $\mathcal{L}_n(P, \mathrm{m})$

Quantum theory is modeled in separable Hilbert space. As emphasized at the outset this work will be restricted to finite dimensional Hilbert spaces which model quantum systems with finite number of possible outcomes. Possible extensions to infinite dimensional systems will be discussed in concluding sections of the next chapter. In this section I present two languages which describe quantum systems in a fixed dimension n. These languages serve two purposes. First they form the basis for the full logic and secondly provide the necessary tools and techniques for studying formal properties of the extended language to be introduced in the next chapter.

Let n be a fixed natural number called the dimension. The symbols $\mathtt{b}$,$\mathtt{c}$,$\mathtt{d}$, etc. will stand for basis variables. Sometimes I use $\mathtt{b}'$, $\mathtt{b}''$, $\mathtt{b}^i$ for basis variables. However, for reasons made clear below subscripts are *never* used to refer to basis variables. Occasionally, I use the ungainly notation like $\mathtt{b}^{(n)\prime}$ to emphasize the dimension. Associated to each basis variable $\mathtt{b}$ is a set $\{\mathtt{b}_0, \mathtt{b}_1, \ldots, \mathtt{b}_{n-1}\}$ of symbols called the basis components. This association

is fixed. Semantically the basis variables correspond to an orthonormal basis in a Hilbert space of dimension n and basis components correspond to the vectors in the basis in some ordering. It will be assumed that a basis specification comes with an ordering prescription. The intuition behind this notation reflects the (idealised) measurement of a physical system. Recall that a measurement of some attribute of a quantum system is always with reference to a basis of the ambient Hilbert space of the system or part thereof. The basis could be explicit or implicit. The measurement statistics correspond to the probability distributions predicted by the theory. Thus let $\{\alpha_0, \alpha_2, \ldots, \alpha_{n-1}\}$ be a basis in the Hilbert space H. The outcome of a measurement in this basis would yield a result corresponding to one of the α_i's. These outcomes are mutually exclusive and repeatable in the sense that if the result of a measurement in the basis yields α_0, say, then a subsequent measurement in the same basis is certain to yield α_0. This is a typical classical situation. Thus as long we confine to a single basis the classical situation prevails. Thus for a generic basis variable $\mathfrak{b}$ define basis formulas or $\mathfrak{b}$-formulas as follows:

1. Each $\mathfrak{b}_i$ is a $\mathfrak{b}$-formula.
2. If α and α' are $\mathfrak{b}$-formulas then so is $\alpha \vee \alpha'$.
3. If α is a $\mathfrak{b}$-formula then $\neg\alpha$ is also one.
4. Any $\mathfrak{b}$-formula is constructed as above.

Note that by definition basis formulas always refer to one basis. Thus, for example, $\mathfrak{b}_0 \vee \neg\mathfrak{c}_2$ is *not* a basis formula. Therefore, the interpretational problems of quantum logic (see e. g. [Bub97]) do not arise. The semantics of these connectives will be given in the next section. For example, $\neg\mathfrak{b}_0$ corresponds to the subspace orthogonal to that of $\mathfrak{b}_0$. Define as usual $\alpha \wedge \beta \stackrel{\text{def}}{=} \neg(\neg\alpha \vee \neg\beta)$. Intuitively, $\mathfrak{b}_0 \vee \mathfrak{b}_1$ would correspond subspaces spanned by vectors corresponding to $\mathfrak{b}_0$ and $\mathfrak{b}_1$. The use of the boolean connectives is

justified by the fact that the set of subspaces of a Hilbert space generated by direct sum from a set of mutually orthogonal 'atomic' subspaces is closed under complementation, sums, and intersections.

A *probability term* is an expression of the form $P(\alpha)$ where α is a $\mathtt{b}$-formula for some basis variable $\mathtt{b}$. A *linear probability atom* is an expression of the form

$$a_0 \cdot P(\alpha_0) + \ldots + a_k \cdot P(\alpha_k) < c \text{ or}$$
$$a_0 \cdot P(\alpha_0) + \ldots + a_k \cdot P(\alpha_k) = c$$

where each a_i is an integer, c is an integer, and each α_i is a $\mathtt{b}^i$-formula for some basis variable $\mathtt{b}^i$. Sometimes, I use the predicate $\leq$, but it must be remembered that $\mathtt{t}_1 \leq \mathtt{t}_2$ stands for $(\mathtt{t}_1 < \mathtt{t}_2) \vee \mathtt{t}_1 = \mathtt{t}_2$. However, we note that it suffices to start with the predicate $\leq$ only. The formulas of the language $\mathcal{L}_n(P)$ are all the boolean combinations of linear probability atoms i.e., each linear probability atom is a formula of $\mathcal{L}_n(P)$, and if ϕ_1 and ϕ_2 are formulas of $\mathcal{L}_n(P)$ then so are $\neg\phi_1$, and $\phi_1 \wedge \phi_2$. The constructions $\phi_1 \vee \phi_2$, $\phi_1 \Rightarrow \phi_2$, $\phi_1 \Leftrightarrow \phi_2$ may be defined in this language as usual. Expressions such as $X < Y$, $X = Y$, where X, Y are linear combinations of probability terms are also definable in this language. Thus, $X = Y$ is defined as $X \leq Y \wedge Y \leq X$ and $X < Y$ as $X \leq Y \wedge \neg(Y \leq X)$.

For the language $\mathcal{L}_n(P, \mathbf{m})$, we add *transition matrix or amplitude terms*, which are expressions of the form $m_{ij}(\mathtt{b}, \mathtt{c})$, where $i, j \in \{0, \ldots, n-1\}$. The reason for this nomenclature is that they give rise to transition probability terms $T(\mathtt{b}_i, \mathtt{c}_j)$ which are introduced as defined expressions by the equation

$$T(\mathtt{b}_i, \mathtt{c}_j) = |m_{ij}(\mathtt{b}, \mathtt{c})|^2$$

Intuitively, transition probabilities are a kind of conditional probability, expressing the probability that a measurement in basis $\mathtt{c}$ will have outcome satisfying $\mathtt{c}_j$, given that the current state is described by $\mathtt{b}_i$. However, the transition probabilities cannot be assigned arbitrarily, they arise out of

certain *linear* relationships that exists between the bases. More precisely, these relationships are but the *unitary* transformations from one orthonormal basis to the other which we call the transition matrix. The transition probabilities are the squared modulus of the appropriate entries of the corresponding unitary matrix(see Chapter 2). Thus the unitary matrices are more fundamental than the transition probabilities although only the later can be directly 'measured'. As stated earlier the present approach to formal reasoning about quantum system is motivated by the view that the goal of any theory is to give a reasonable explanation of observed phenomena. In a quantum system the observed quantities are the probability distributions-both absolute and transitional. Thus for each pair of basis symbols $\mathtt{b}$ and $\mathtt{c}$, $m_{ij}(\mathtt{b},\mathtt{c})$ is a symbol. If we view $\mathcal{L}_n(P,\mathbf{m})$ as a multi-sorted logic then $m_{ij}(\mathtt{b},\mathtt{c})$ are function symbols from the sort representing basis variable to those of $\mathbb{RC}$. The intuition behind the these notation is clear: $m_{ij}(\mathtt{b},\mathtt{c})$ represent the elements of the unitary matrix connecting the bases corresponding to $\mathtt{b}$ and $\mathtt{c}$. Further, as explained in Chapter 2, a unitary transformation may also be viewed as an evolution operator characterizing the dynamics of the system. There are many subtleties and only some of these are touched upon in the aforementioned chapter. The reader is urged to a perusal of literature on quantum theory (e.g [Per95], [dE76], [Got03] to name a few).

The language $\mathcal{L}_n(P,\mathbf{m})$ is defined as extension of the language $\mathcal{L}_{\mathbb{RC}}$ of the previous section. Consequently, $\mathcal{L}_n(P,\mathbf{m})$ inherits the symbols (both logical and nonlogical) of $\mathbb{RC}$, in particular, the predicate R standing for real terms and the quantifiers. Usually, x, y, z etc. (possibly with subscripts) will stand for variables in $\mathbb{RC}$. Note that we now have two types of variables: basis variables for $\mathcal{L}_n(P)$ and the variables of $\mathbb{RC}$. Sometimes, I use abbreviation "type" for real variables especially in conjunction with quantifiers. Thus, if $\Phi(x_1,\ldots,x_n)$ is a formula with free variables $(x_1,\ldots,x_n)$ then

$$\exists x_1 \ldots x_n : \mathbb{R}(\Phi(x_1,\ldots,x_n)) \text{ stands for}$$
$$\exists x_1 \ldots x_n (Rx_1 \ldots x_n \wedge \Phi(x_1,\ldots,x_n))$$

Similar syntactic constructs apply to the universal quantifiers. Note that this notation reflects the multi-sorted approach we mentioned earlier. I use the standard notation for polynomials in several variables. These are the terms of $\mathbb{RC}$. If p, q, r etc. are such polynomials then the atomic formulas of $\mathbb{RC}$ are of the form

$$R(p),\ q = 0,\ \text{and}\ r > 0.$$

Given an atomic formula of **RC** the *atomic formulas* of $\mathcal{L}_n(P, \mathbf{m})$ are defined by replacing (uniformly) some of the variables in the $\mathbb{RC}$-formula by probability terms, transition probability terms and unitary terms. The unitart terms are constructed from $m_{ij}(\mathtt{b}, \mathtt{c})$. For example, let $x_1^2 x_2^3 + 5x_1 + x_3^2 - 1$ be a polynomial expression then

$$(P(\mathtt{b}_i))^2 (T(\mathtt{b}_i, \mathtt{c}_j))^3 + 5(P(\mathtt{b}_i)) + (T(\mathtt{b}_i, \mathtt{c}_j))^2 - 1 > 0$$

is an atomic formula.

4.3.2 Semantics of $\mathcal{L}_n(P, \mathbf{m})$

We now present the semantics for the language $\mathcal{L}_n(P, \mathbf{m})$(and consequently for the sublanguage $\mathcal{L}_n(P)$). It is interpreted in the theory $\mathbb{RC}$. However, although there are no explicit modal operators there are some similarities to Kripke semantics [Gol92] for modal logics. The probability formulas are interpreted with respect to a specified collection of states.

A *structure* for $\mathcal{L}_n(P, \mathbf{m})$ is an n-dimensional Hilbert space H over any model of $\mathbb{RC}$. Recall that $\mathbb{RC}$ is the theory of a real closed field *and* its unique algebraic extension by $i = \sqrt{-1}$. A *state* within this structure is a unit vector ψ in H. An *interpretation* of $\mathcal{L}_n(P, \mathbf{m})$ in a structure H is function π, such that

1. for each basis variable $\mathtt{b}$, $\pi(\mathtt{b})$ is an orthonormal basis $\psi_0, \ldots, \psi_{n-1}$ of H; (we write $\pi(\mathtt{b})_i$ for ψ_i)

2. for each complex variable X, $\pi(X)$ is a complex number.

3. for each real variable x, $\pi(x)$ is a real number; in the context of $\mathbb{RC}$ this means that Rx holds. Equivalently, if one treats $\mathbb{RC}$ as a multi-sorted logic then one could assign different sets of variables for complex and real sorts. Sometimes the latter approach is more convenient. Thus, saying that a term $\mathbf{t}$ is of sort real is equivalent to the formula $R\mathbf{t}$.

If $M = (m_{ij})$ is an $n \times n$ unitary matrix and $B = \psi_0, \ldots, \psi_{n-1}$ is a sequence of vectors of H, we write MB for the sequence of vectors $\psi'_0, \ldots, \psi'_{n-1}$, where $\psi'_k = \Sigma_{i=0}^{n-1} m_{ik}\psi_i$.[1] If B is an orthonormal basis of H then so is MB. Next extend the interpretation π to terms $\mathbf{t}$ of various sorts as follows. Given the term $\mathbf{t}$, a state ψ and an interpretation π, we define the interpretation $[\![\mathbf{t}]\!]_{\pi,\psi}$ of $\mathbf{t}$ with respect to π and ψ as follows. Basis variables are interpreted as bases:

1.
$$[\![\mathsf{b}]\!]_{\pi,\psi} = \pi(\mathsf{b}),$$
when b is a basis variable.

For a basis variable b we interpret b-formulas as projection operators on H (these may also be understood as representing the subspaces of H onto which they project):

2.
$$[\![\mathsf{b}_i]\!]_{\pi,\psi} = |\psi'\rangle \langle\psi'|,$$
where $\psi' = \pi(\mathsf{b})_i$. Note that the fact that an orthonormal basis is determined up to multiples of complex numbers of unit modulus is already taken into account in the above interpretation since projection operators are invariant with respect to such multiples. Explicitly, if a is a complex number of modulus 1, and $|\alpha\rangle$ is a unit vector then
$$(a\,|\alpha\rangle)(\langle\alpha|\,\overline{a}) = |\alpha\rangle\langle\alpha|$$

[1] This is the standard convention for transformation of basis [NC01]

3.
$$[\![\alpha_1 \vee \alpha_2]\!]_{\pi,\psi} = [\![\alpha_1]\!]_{\pi,\psi} \oplus [\![\alpha_2]\!]_{\pi,\psi}$$
this is the projection operator projecting onto sum of the subspaces of H that are the images of the projectors $[\![\alpha_1]\!]_{\pi,\psi}$ and $[\![\alpha_1]\!]_{\pi,\psi}$ which could be written as the product of these projectors.

4.
$$[\![\neg\alpha]\!]_{\pi,\psi} = [\![\alpha]\!]^{\perp}_{\pi,\psi}$$

is the projection operator projecting onto the orthogonal complement of the image of H under $[\![\alpha]\!]_{\pi,\psi}$. Terms of **RC** including the unitary matrix entry terms $m_{ij}(\mathsf{b},\mathsf{c})$, are interpreted as complex numbers:

5. $[\![x]\!]_{\pi,\psi} = \pi(x)$, when x is real or complex variable;

6. $[\![k]\!]_{\pi,\psi} = k$, when k is an integer;

7.
$$[\![P(\alpha)]\!]_{\pi,\psi} = \mathtt{Tr}([\![\alpha]\!]_{\pi,\psi}(|\psi\rangle\langle\psi|))$$
Recall that $[\![\alpha]\!]_{\pi,\psi}$ is a projection operator onto some subspace S defined by the interpretation of the constituent basis variables of α and hence $[\![\alpha]\!]_{\pi,\psi}(\psi)$ is the projection of ψ on S.

8. $[\![m_{ij}(\mathsf{b},\mathsf{c})]\!]_{\pi,\psi} = c_{ij}$, where $M = (c_{ij})$ is the $n \times n$ (unitary) complex array such that $M\pi(\mathsf{b}) = \pi(\mathsf{c})$; Thus if $\pi(\mathsf{b}) = \{|\alpha_0\rangle, \ldots, |\alpha_{n-1}\rangle\}$ and $\pi(\mathsf{c}) = \{|\beta_0\rangle, \ldots, |\beta_{n-1}\rangle\}$ then $c_{ij} = \langle\alpha_i|\beta_j\rangle$.

9. $[\![T(\mathsf{b}_i,\mathsf{c}_j)]\!]_{\pi,\psi} = \mathtt{Tr}([\![\mathsf{b}_i]\!]_{\pi,\psi}[\![\mathsf{c}_j]\!]_{\pi,\psi}) = |m_{ij}(\mathsf{b},\mathsf{c})|^2$.

10. $[\![X \cdot Y]\!]_{\pi,\psi} = [\![X]\!]_{\pi,\psi} \cdot [\![X]\!]_{\pi,\psi}$

11. $[\![X + Y]\!]_{\pi,\psi} = [\![X]\!]_{\pi,\psi} + [\![Y]\!]_{\pi,\psi}$

To give semantics to formulas of $\mathcal{L}_n(P,\mathbf{m})$, we define a relation of satisfaction of a formula ϕ at a state ψ in a structure H, with respect to an interpretation π, denoted by $H,\pi,\psi \models \phi$. The definition is by the following:

1. $H, \pi, \psi \models X \leq Y$ if $[\![X]\!]_{\pi,\psi} \leq [\![Y]\!]_{\pi,\psi}$ and $H, \pi, \psi \models R(X)$ if $R([\![X]\!]_{\pi,\psi})$ in $\mathbb{RC}$.

2. $H, \pi, \psi \models \neg\phi$ if not $H, \pi, \psi \models \phi$;

3. $H, \pi, \psi \models \phi_1 \wedge \phi_2$ if $H, \pi, \psi \models \phi_1$ and $H, \pi, \psi \models \phi_2$;

4. $H, \pi, \psi \models \exists x(\phi)$ if there exists a complex number c such that $H, \pi[c/x], \psi \models \phi$;

A formula Φ of $\mathcal{L}_n(P, \mathbf{m})$ is *satisfiable* (in the n-dimensional Hilbert space H) if there exists an interpretation π and a state ψ such that $H, \pi, \psi \models \Phi$. A formula ϕ is *valid* (in H) if $H, \pi, \psi \models \Phi$ for all interpretations π and states ψ. I now prove a lemma which will be useful for bringing formulas to a standard form. Let $p(x_1, \ldots, x_k)$ be a real polynomial. We define a probability atom as a formula in $\mathcal{L}_n(P, \mathbf{m})$ obtained by replacing the x_i by probability terms $P(\phi_i[\mathfrak{b}_j])$. Here $\phi_i[\mathfrak{b}]$ is a basis formula in the basis variable $\mathfrak{b}$. Now for a basis variable $\mathfrak{b}$ let $\pi(\mathfrak{b}) = \{\alpha_0, \ldots, \alpha_{n-1}\}$. Then $\pi(\mathfrak{b}_i)$ is the projection operator $|\alpha_i\rangle\langle\alpha_i|$. It is clear from the definitions that $\pi(\neg\mathfrak{b}_i) = |\alpha_0\rangle\langle\alpha_0| + \ldots + |\alpha_{i-1}\rangle\langle\alpha_{i-1}| + |\alpha_{i+1}\rangle\langle\alpha_{i+1}| + \ldots + |\alpha_{n-1}\rangle\langle\alpha_{n-1}|$. The interpretation of $\neg\mathfrak{b}_i$ is the projection operator onto the orthogonal complement of the subspace spanned by α_i. Similarly the interpretation of $\mathfrak{b}_i \vee \mathfrak{b}_j$ is the projection operator $P_{\alpha_i} + P_{\alpha_j}$. The upshot is that any $\mathfrak{b}$-formula can be written as a sum of distinct projection operators: $\phi \Leftrightarrow \vee_j \mathfrak{b}_{i_j}$. Conjunction of two $\mathfrak{b}$-formulas is interpreted as the projection onto the intersection of corresponding subspace. Again it is clear from definitions that if $P(\mathfrak{b}_i) = x_i$ and $P(\mathfrak{b}_j) = x_j$ $(i \neq j)$ then $P(\mathfrak{b}_i \vee \mathfrak{b}_j) = x_i + x_j$. In other words in a maximal measurement with respect to some basis the events are mutually exclusive. Since we can reduce any basis formula to a disjunction over distinct basis variables the above additive property implies that each $P(\phi)$ may be replaced by terms like $\sum_k P(\mathfrak{b}_{i_k})$. Hence replacing it in the probability atom $p(x_1/P(\phi_1), \ldots, x_k/P(\phi_k))$ we get

Lemma 6 *Given polynomial* $p(x_1,\ldots,x_k)$ *and the probability atom* $\Phi \equiv p(x_1/P(\phi_1),\ldots,x_k/P(\phi_k))$ *in* $\mathcal{L}_n(P,\mathbf{m})$ *there is a polynomial* $q(y_1,\ldots,y_m)$ *such that* $p(x_1/P(\phi_1),\ldots,x_k/P(\phi_k)) > 0$ *iff* $q(y_1/P(\mathtt{b}_0^1),\ldots,y_m/P(\mathtt{b}_{n-1}^r)) > 0$ *where* $\{\mathtt{b}^1,\ldots,\mathtt{b}^r\}$ *are the distinct basis variables used in the formula.*

Proof: Let $\mathtt{b}^1,\ldots,\mathtt{b}^r$ be the distinct basis symbols appearing in Φ. Let $m = n\cdot r$. Without loss of generality one may assume that the variables for which *distinct* substitutions (of basis variables) are made are $x_1,\ldots,x_r$ respectively. For each variable x_i choose a set of new variables $\{y_{(i-1)n+1}, y_{(i-1)n+2},\cdots,y_{in}\}$. From the discussion above each basis formula ϕ_i is of the form $\mathtt{b}_{j_1}^i \vee \mathtt{b}_{j_2}^i \vee \ldots \vee \mathtt{b}_{j_s}^i$. Replace x_i by $\{y_{(i-1)n+j_1} + y_{(i-1)n+j_2} + \ldots + y_{i(n-1)+j_s}\}$ in the polynomial $p(x_1,\ldots,x_k)$ and let $q(y_1,\ldots,y_m)$ be the resulting polynomial. Hence, $\pi(p(x_1/P(\phi_1),\ldots,x_k/P(\phi_k)))$ is equal to $q(y_1/P(\mathtt{b}_0^1),\ldots,y_m/P(\mathtt{b}_{n-1}^r)$ and the lemma follows. □

4.3.3 Examples

In this subsection I give some examples of formulas $\mathcal{L}_n(P,\mathbf{m})$ which express important physical concepts. The restrictions on the language to a fixed dimension put limitations on the expressiveness of the logic. Nevertheless, some important and interesting notions of quantum theory and in particular, quantum computation can be expressed.

Superposition: A vector $|\alpha\rangle$ is a *superposition* of two vectors $|\beta_0\rangle$ and $|\beta_1\rangle$ if it is a linear combination of the two, i.e., $|\alpha\rangle = c_1|\beta_0\rangle + c_2|\beta_1\rangle$ for some complex numbers c_1, c_2. Consider the formula $T(\mathtt{b}_0, \mathtt{b}_0' \vee \mathtt{b}_1') \stackrel{\text{def}}{=} T(\mathtt{b}_0,\mathtt{b}_0') + T(\mathtt{b}_0,\mathtt{b}_1') = 1$. If $\pi(\mathtt{b}_0) = |\alpha\rangle$, $\pi(\mathtt{b}_0') = |\beta_0\rangle$ and $\pi(\mathtt{b}_1') = |\beta_1\rangle$ then $H,\pi,\psi \models T(\mathtt{b}_0,\mathtt{b}_0'\vee\mathtt{b}_1') = 1$ iff $\mathbf{Tr}(|\alpha\rangle\langle\alpha|(|\beta_0\rangle\langle\beta_0| + |\beta_1\rangle\langle\beta_1|)) = 1$. This is equivalent to $|\langle\beta_0|\alpha\rangle|^2 + |\langle\beta_1|\alpha\rangle|^2 = 1$, which is true iff the state $|\alpha\rangle$ is a superposition of the states $|\beta_1\rangle$ and $|\beta_2\rangle$. That is, the formula expresses that "$[\mathtt{b}_0]_\pi$ is a vector in the subspace spanned by $[\mathtt{b}_0']_\pi$ and $[\mathtt{b}_1']_\pi$".

Phase Relations: Let $\mathfrak{b}^0, \ldots, \mathfrak{b}^k$ be $k+1$ bases. Then the following formula states a relation between $\mathfrak{b}^0$ and the $\mathfrak{b}^j$, for $j = 1 \ldots k$.

$$\begin{aligned}\mathbf{MP}_k \quad & \forall x_1 \ldots x_n : \mathbb{R}.(\textstyle\bigwedge_{i=0}^{n-1} P(\mathfrak{b}_i^0) = x_i^2 \Rightarrow \\ & \exists z_0 \ldots z_{n-1} : \mathbb{C}(\textstyle\bigwedge_{i=0}^{n-1} |z_i| = 1 \wedge \\ & \textstyle\bigwedge_{j=1}^{k} \bigwedge_{i=0}^{n-1} P(\mathfrak{b}_i^j) = |\Sigma_{r=1}^{n} m_{ir}(\mathfrak{b}^j, \mathfrak{b}^0) x_r z_r|^2))\end{aligned}$$

Recall that if $\{\alpha_0, \alpha_1, \ldots, \alpha_{n-1}\}$ is a basis and a quantum system is in a state $\beta = \sum_{i=0}^{n-1} c_i \alpha_i$ then a measurement in the α-basis yields α_i with probability $|c_i|^2$. The probability distributions thus give the modulus of the coefficients but if we measure in some other basis then the *phases* come into play, hence the name. The above formulas simply state the following. Given a state of the system and a set of bases along with their (unitary) transformation relations-in fact a single basis $[\mathfrak{b}^0]_\pi$ and the unitary matrices connecting it to all other bases in the set will suffice-the relation amongst the probabilities are given by the formulas $\mathbf{MP}_k$. Conversely, if given probability distributions with respect to a given set of bases then the above set of formulas express that there is a state, i. e. a vector in Hilbert space which generates the distribution according to the laws of quantum mechanics. A formula that is closely related to $\mathbf{MP_k}$ is the one that follows the proposition below.

Proposition 1 *The formula* $\mathbf{MP}_k$ *is valid for all* $k \geq 1$.

Proof: Let π be any interpretation and $|\psi\rangle$ any vector in H_n. If $[\![P(\mathfrak{b}_i^0)]\!]_{\pi,|\psi\rangle} = \pi(x_i)^2$, then we may write $|\psi\rangle = \sum_{i=1}^{n} c_i \pi(x_i) \pi(\mathfrak{b}^0)_i$, where the c_i are complex numbers with $|c_i| = 1$. Define $\pi(z_i) = c_i$. We can then calculate the probabilities with respect to other bases as follows:

$$\begin{aligned}[\![P(\mathfrak{b}_i^j)]\!]_{\pi,|\psi\rangle} &= |\langle \pi(\mathfrak{b}^j)_i \mid \psi\rangle|^2 \\ &= |\langle \textstyle\sum_{r=1}^{n} [\![m_{ri}(\mathfrak{b}^0, \mathfrak{b}^j)]\!]_{\pi,|\psi\rangle} \cdot \pi(\mathfrak{b}^0)_r \mid \psi\rangle|^2 \\ &= |\textstyle\sum_{r=1}^{n} [\![m_{ri}(\mathfrak{b}^j, \mathfrak{b}^0)]\!]_{\pi,|\psi\rangle} \cdot \pi(x_r) \cdot \pi(z)_r|^2\end{aligned}$$

from which it can be seen that $\mathbf{MP}_k$ holds. □

Quantum State Tomography: Suppose we are given a collection of identically prepared systems (an ensemble), which corresponds to an unknown

quantum state. Quantum state tomography (QST) addresses the problem of determining this unknown state. By measuring the ensemble in a single basis, we may determine a probability distribution over the outcomes associated to the basis elements. This distribution does not suffice to determine the state of the system. However, we may also divide the original collection of systems into subcollections and subject each subcollection to maximal measurement corresponding to an appropriately chosen orthonormal bases. We get sets of probability distributions. For an appropriately chosen set of measurements, this set of distributions suffices to determine the state uniquely. The following formula expresses this fact.

Let $\mathbf{u}$ be the sequence of variables u_{ij}^k where $1 \leq i, j, k \leq n$.

$$\begin{aligned}\exists \mathbf{u}[\ & \textstyle\bigwedge_{1\leq i,j,k\leq n} m_{ij}(\mathsf{b}, \mathsf{c}^k) = u_{ij}^k \ \Rightarrow \forall z_1 \ldots z_n(\\ & \textstyle\bigwedge_{1\leq i,k\leq n} P(\mathsf{c}_i^k) = |\sum_j u_{ij}^k z_j \sqrt{P(\mathsf{b}_j)}|^2 \\ & \Rightarrow \textstyle\bigwedge_i P(\mathsf{b}_i') = |\sum_j m_{ij}(\mathsf{b}, \mathsf{b}') z_j \sqrt{P(\mathsf{b}_j)}|^2 \ \ ) \ ].\end{aligned}$$

This formula is valid. It expresses the fact that there is a "pattern of inter-relation"(the unitary transformation relating them) between a set of bases b and $\mathsf{c}^1, \mathsf{c}^2, \ldots \mathsf{c}^n$, captured by the values $\mathbf{u}$, such that for any vector ψ, the probabilities of measurements associated to a set of bases related in this pattern provide sufficient information to calculate the probabilities of measurements with respect to any other basis b'. Note that by the discussion of the formula MP_k above, it is always possible to find complex numbers $z_1, \ldots z_k$ such that

$$\bigwedge_{1\leq i,k\leq n} P(\mathsf{c}_i^k) = |\sum_j u_{ij}^k z_j \sqrt{P(\mathsf{b}_j)}|^2 \tag{4.11}$$

is satisfied. Thus the universally quantified formula in the conclusion is never true vacuously. The formula therefore expresses that given an appropriately related set of bases b and $\mathsf{c}^1, \mathsf{c}^2, \ldots \mathsf{c}^n$, it is possible, given any basis b', to calculate the values $P(\mathsf{b}_i')$ from the values u_{ij}^k, $P(\mathsf{b}_i)$, $P(\mathsf{c}_i^k)$ and $m_{ij}(\mathsf{b}, \mathsf{b}')$. We do this by first solving the equation (4.11) for the phase values $z_1, \ldots, z_n$, and then computing $P(\mathsf{b}_i')$ as $|\sum_j m_{ij}(\mathsf{b}, \mathsf{b}') z_j \sqrt{P(\mathsf{b}_j)}|^2$.

Uncertainty Relations: We say that bases b and b' are complementary if $nT(\mathsf{b}_i, \mathsf{b}_j{}') = 1$ holds for all $i, j = 1 \ldots n$. Let $n = 2$. Then the formula

$$\bigwedge_i 2T(\mathsf{b}_i, \mathsf{b}_j{}') = 1 \ \wedge \ P(\mathsf{b}_1) = x_1^2 \ \wedge \ P(\mathsf{b}_2) = x_2^2 \ \Rightarrow$$
$$(x_1 - x_2)^2 \leq 2P(\mathsf{b}_j)' \leq (x_1 + x_2)^2$$

expresses an uncertainty relation. For example if $x_1 = 1$ then there is no uncertainty in the result for a maximal test with respect to the b-basis. But then we get the probability for both results in the b'-basis equal to one half. Thus, there is maximum uncertainty in an information theoretic sense.

Quantum Gates:

The familiar 'not' gate is also called X (or Pauli-X in honour of W. Pauli who used them first in his analysis of electron spin). It can be expressed in terms of transition probabilities because its entries are real. To express other quantum gates (unitary matrices) uniformly we have to use the notation for (unitary) transformation of basis. Below, I give the formulas corresponding to some of the important unitary gates. Later I write this in an equivalent form that is more akin to the circuit model favoured by quantum computing community. The notation for the gates below are standard [NC01] and will be used later. The gates below are all 2 dimensional, that is, all the formulas are in $\mathcal{L}_2(P, \mathbf{m})$. I also suppress the basis variables. Thus m_{ij} stands for $m_{ij}(\mathsf{b}, \mathsf{c})$ for some basis variables.

1. **Pauli-X Gate**

$$X \stackrel{\text{def}}{=} m_{00} = m_{11} = 0 \wedge m_{01} = m_{10} = 1$$

2. **Pauli-Y Gate**

$$Y \stackrel{\text{def}}{=} m_{00} = m_{11} = 0 \wedge m_{01} = -m_{10} = i$$

3. **Pauli-Z Gate**

$$Z \stackrel{\text{def}}{=} m_{00} = -m_{11} = 1 \wedge m_{01} = m_{10} = 0$$

4. **Hadamard Gate**

$$H \stackrel{\text{def}}{=} m_{00} = -m_{11} = 1/sqrt2 \wedge m_{01} = m_{10} = 1/sqrt2$$

5. **Phase Gate**

$$Ph \stackrel{\text{def}}{=} m_{00} = 1 \wedge m_{11} = i \wedge m_{01} = m_{10} = 0$$

6. $\pi/8$ **Gate**

$$\pi/8 \stackrel{\text{def}}{=} m_{00} = 1 \wedge m_{11} = (1+i)\sqrt{2} \wedge m_{01} = m_{10} = 0$$

These are some of the most frequently used gates. Another useful gate is the controlled-NOT gate C. It is a 4 dimensional gate and thus a formula of $\mathcal{L}_4(P, \mathbf{m})$. One can write it in the present formalism. But it would be more meaningful and intuitive in the tensor formalism to be introduced later. The Hadamard gate, phase gate, $pi/8$ gate and the controlled-NOT gate constitute an *universal* set of gates in the sense that any unitary operation on any number of qubits may be approximated by these gates. This is significant because the unitary gates play the role of logic operations(e. g. AND, OR, NOT, XOR) of classical circuits. Where as in the former 2 gates like OR and NOT constitute a universal basis the uncountable infinity of unitary gates can only be approximated by a given finite set of gates. Another point to be noted is that the constants used in the definition of these gates are expressible in $\mathbb{RC}$.

4.4 Axiomatization

I now present axiomatizations and prove completeness results for the languages $\mathcal{L}_n(P)$ and $\mathcal{L}_n(P, \mathbf{m})$. Although the former has very limited expressive power I deal with it separately for three reasons. First, because of its close correspondence with logics for classical probabilistic reasoning. Secondly, it provides us with some techniques for proving results about the properties of the logic. Finally, it helps us understand the 'quantum' characteristics of $\mathcal{L}_n(P, \mathbf{m})$ and further extensions in the next chapter.

4.4.1 Axiomatizing $\mathcal{L}_n(P)$

The axiomatization of $\mathcal{L}_n(P)$ consists of a number of parts, each dealing with one of the syntactic constructs of the language.

The first fragment of the axiomatization deals with the boolean logic of the $\mathtt{b}$-formulas. As this is slightly richer than propositional logic, we identify for each basis variable $\mathtt{b}$ a fragment of the proof theory that deals only with $\mathtt{b}$-formulas. The axioms of this fragment can be taken to be any complete axiomatization of propositional logic over the atomic formulas $\mathtt{b}_1, \ldots, \mathtt{b}_n$, with, e.g., Modus Ponens as the proof rule, plus the following axioms that capture the fact that we are dealing with an n-dimensional Hilbert space. Let $J_n \stackrel{\text{def}}{=} \{0, \ldots, n-1\}$.

B1 $\mathtt{b}_0 \vee \ldots \vee \mathtt{b}_{n-1}$

B2 $\neg(\mathtt{b}_i \wedge \mathtt{b}_j)$ for $i \neq j$ and $i, j \in J_n$

We say that a $\mathtt{b}$-formula ϕ is a $\mathtt{b}$-tautology, and write $\vdash_{\mathtt{b}} \phi$ if it is a tautology of ordinary propositional logic or it can be derived from the above axioms alone by propositional reasoning. Note that these definitions isolate reasoning about $\mathtt{b}$-formulas from reasoning about $\mathtt{c}$-formulas when $\mathtt{b}$ and $\mathtt{c}$ are distinct. It is convenient to introduce propositional constants $\top$ and $\bot \equiv \neg\top$ along with axiom $\top$. Then the two axioms above are equivalent to

$$\mathtt{b}_0 \vee \ldots \vee \mathtt{b}_{n-1} \Leftrightarrow \top$$

and

$$(\mathtt{b}_i \wedge \mathtt{b}_j) \Leftrightarrow \bot$$

Next, we have some axioms capturing the properties of the probability operator. The following axioms correspond very closely to the axioms W1-W4 of Fagin et al. [FHM90], but with the difference that we need to be careful to respect the syntactic constraints on probability terms. In the following, we require that there is some basis term $\mathtt{b}$ such that ϕ, ϕ_1 and ϕ_2 are $\mathtt{b}$-formulas:

P1 $0 \leq P(\phi) \leq 1$

P2 $P(\phi) = 1$ if ϕ is a b-tautology(i.e. $\vdash_{\mathfrak{b}} \phi$)

P3 $P(\phi_1 \wedge \phi_2) + P(\phi_1 \wedge \neg\phi_2) = P(\phi_1)$

P4 $P(\phi_1) = P(\phi_2)$ if $\phi_1 \Leftrightarrow \phi_2$ is a b-tautology

Note that in P2 and P4, we deal with b-tautologies where Fagin et al have tautologies of propositional logic. Fagin et al note that in their axiomatization, there is no axiom corresponding to the fact that probability measures are countably additive. This does not make their axiomatization incomplete, because the countable additivity is not expressible in the logic. In our logic, we also do not have a countable additivity axiom, but the reason is somewhat simpler: the only measurable properties with respect to a basis denoted by b are those corresponding to the 2^n inequivalent b-formulas. Semantically, these formulas correspond to the linear subspaces generated by subsets of the n basis vectors. But, for infinite-dimensional quantum systems one must somehow incorporate countable additivity. Two possible courses are: 1) infinitary logic, i. e. admit (countably) infinite conjunctions and disjunctions or 2) replace axiom **P3** by an infinite set of axioms. We leave the infinitary case for future study.

In addition to the above axioms, we also need a set of axioms that capture reasoning about linear inequalities. That is, we need to be able to derive formulas such as $(2P(\phi_1) \leq 3 \wedge 4P(\phi_2) \leq 1) \Rightarrow P(\phi_1) + 2P(\phi_2) \leq 2$, the validity of which follows just from the meaning of these operations on real numbers, rather than the meaning of the probability terms. Hence we have the following set of axioms

LinInEq All instances of valid formulas about linear inequalities

for integer constants a_i and c. Here, the x_i are to be instanttiated by probability atoms. [2] There is a however, a small difference between our

[2]The theory of linear inequality can be recursively axiomatized. I refer the reader to Fagin et al. [FHM90]. There are seven axioms on linear inequality. Note that we define $t \geq c$ for $-t \leq -c$, $x = y \Leftrightarrow x \leq y \wedge y \leq x$ and $t < c \equiv (t \leq c) \wedge \neg(t = c)$. Then the axioms are:

I1. $x \leq x$

I2. $a_1x_1 + \cdots + a_kx_k \leq a_1x_1 + \cdots + a_kx_k + 0x_{k+1}$

notation and that of FHM. In the latter the basic predicate is " $\geq$ " and the relations of equality and " $>$" are defined using the basic relation. We have opted to work in language with equality and the basic relation " $<$". The advantage of FHM is that in the simple case of linear probability formulas there is a simple axiomatization of linear equality formulas which does not require the full axiom system of real closed fields. It follows from FHM that such an axiomatization is complete.

A basic formula or a linear probability formula, like the one above, in $\mathcal{L}_n(P)$ is obtained by replacing the variables in a formula of $\mathcal{L}_{INEQ}$ by probability atoms $P(\phi)$. In the case of FHM-logic the construction is identical but ϕ is required to be propositional formula.

As an illustration of axiomatization I make several simple deductions in $\mathtt{Ax}_n(P)$. These results are useful for later purposes. The deductions are informal but could be easily formalized. First, fix some notation. Uppercase roman letters A, B, C (possibly with subscripts)etc. will be used as metavariables over basis formulas (b-formulas). Sometimes I write $A[\mathtt{b}]$ to indicate the particular basis symbol (rather the components $\mathtt{b}_i$) used in A. Bold uppercase letters $\mathbf{A}, \mathbf{B}, \mathbf{C}$ etc. will represent metavariables denoting linear probability atoms i.e. formulas of the form $\sum_i k_i P(A_i) \geq a$ with integer constants a, k_i. Uppercase greek letters will stand for general formulas of the language under consideration, in the present case $\mathcal{L}_n(P)$. Write $\Phi[\mathtt{b}^1, \ldots, \mathtt{b}^j]$ to indicate that $\mathtt{b}^1, \ldots \mathtt{b}^j$ are among the basis variables that occur in Φ. Let $\mathtt{x}$ denote metavariables which denote n variables $x_0, \ldots, x_{n-1}$ in $\mathbf{RC}$. Call

I3. $a_1x_1 + \cdots + a_kx_k \leq c \Rightarrow a_{j_1}x_1 + \cdots + a_{j_k}x_k \leq c$ if $j_1, \ldots, j_k$ is a permutation of $\{1, \ldots, k\}$

I4. $a_1x_1 + \cdots + a_kx_k \leq c \wedge a'_1x_1 + \cdots + a'_kx_k \leq c' \Rightarrow (a_1 + a'_1)x_1 + \cdots + (a_k + a'_k)x_k \leq c + c'$

I5. $a_1x_1 + \cdots + a_kx_k \leq c \Leftrightarrow da_1x_1 + \cdots + da_kx_k \leq c$ for integer $d > 0$

I6. $(t \leq c) \vee (t \geq c)$ where t is a term.

I7. $(t \leq c) \Rightarrow t < d$ for $c < d$.

Call this theory $\mathtt{L}_{\mathtt{INEQ}}$. In this theory equality is a defined predicate. Since the axiom system will be soon replaced the richer theory $\mathbf{RC}$ in which the axioms of linear inequality are theorems we omit futher discussion on this theory.

these vector variables. Bold lowercase roman letters will denote vector variables such that the above correspondence between the 'vector' and 'components' always holds. For example, $\mathbf{x^k}$ will stand for the n-tuple of variables $(x_0^k, \ldots, x_{n-1}^k)$. Let $\Phi[\mathbf{x^1}/\mathbf{b^1}, \ldots .\mathbf{x^k}/\mathbf{b^k}]$ denote the formula in **RC** obtained by substituting x_i^j for every occurrence $P(\mathbf{b}_i^j)$, $0 \leq i \leq n-1$ and $1 \leq j \leq k$. The motivation is a systematic reduction of formulas in $\mathcal{L}_n(P)$ and $\mathcal{L}_n(P, \mathbf{m})$ to formulas of **RC**.

D1. Let ϕ be the formula $\mathbf{b}_{i_1} \vee \mathbf{b}_{i_2} \vee \ldots \vee \mathbf{b}_{i_k}$. Then

$$\vdash_{\mathbf{b}} \neg\phi \Leftrightarrow \mathbf{b}_{j_1} \vee \mathbf{b}_{j_2} \vee \ldots \vee \mathbf{b}_{j_{n-k}} \tag{4.12}$$

where $\{j_1, j_2, \cdots j_{n-k}\}$ is the complement of $\{i_1, i_2, \cdots i_k\}$ in $\{0, 1, \ldots n-1\}$.

It follows from **B1** and distributive laws that

$$\begin{aligned} \neg\phi &\Leftrightarrow \neg\phi \wedge (\mathbf{b}_1 \vee \ldots \vee \mathbf{b}_n) \\ &\Leftrightarrow (\neg\mathbf{b}_{i_1} \wedge \ldots \wedge \neg\mathbf{b}_{i_k}) \wedge (\mathbf{b}_1 \vee \ldots \vee \mathbf{b}_n) \\ &\Leftrightarrow \bigvee_{j=1}^{n} ((\neg\mathbf{b}_{i_1} \wedge \ldots \wedge \neg\mathbf{b}_{i_k}) \wedge \mathbf{b}_j) \end{aligned}$$

Now by **B2**, we have that $\vdash \mathbf{b}_j \Rightarrow \neg\mathbf{b}_i$ for $i \neq j$. It follows that each term $(\neg\mathbf{b}_{i_1} \wedge \ldots \wedge \mathbf{b}_{i_k}) \wedge \mathbf{b}_j$ is provably equivalent to $\mathbf{b}_j$ if $j \in \{i_1, \ldots, i_k\}$ and provably equivalent to $\perp$ otherwise. This yields the result.

D2. For two disjoint subsets I and J of $\{1, 2, \cdots, n\}$ we have

$$(\bigvee_{i \in I} \mathbf{b}_i) \bigwedge ((\bigvee_{i \in J} \mathbf{b}_i) \Leftrightarrow \perp$$

This is an easy consequence of **B2** and the distributive laws.

D3. If $\vdash_{\mathbf{b}} \neg\phi$ then $\vdash P(\phi) = 0$.

This is obtained by taking ϕ_1 to be any $\mathtt{b}$-tautology and ϕ_2 equal to ϕ in **P3**. Noting that by **P4** we have $\vdash P(\phi_1 \wedge \psi) = P(\psi)$ for any $\mathtt{b}$-formula ψ, we obtain $\vdash P(\phi) + P(\neg\phi) = P(\phi_1)$. By **P2**, we have $\vdash P(\phi_1) = 1$ Similarly, by **P2**, if $\neg\phi$ is a $\mathtt{b}$-tautology also, we have $\vdash P(\neg\phi) = 1$. It follows that $\vdash P(\phi) = 0$. Note that we use the axioms of the theory **LinInEq** which also yield the familiar laws equality.

D4. If $\phi = \mathtt{b}_{i_1} \vee \mathtt{b}_{i_2} \vee \ldots \vee \mathtt{b}_{i_k}$ and $\psi = \mathtt{b}_{j_1} \vee \mathtt{b}_{j_2} \vee \ldots \vee \mathtt{b}_{j_m}$ then $\vdash \phi \wedge \psi \Leftrightarrow \mathtt{b}_{r_1} \vee \mathtt{b}_{r_2} \vee \ldots \vee \mathtt{b}_{r_s}$ where $\{r_1, r_2, \cdots, r_s\} = \{i_1, i_2, \cdots, i_k\} \cap \{j_1, j_2, \cdots, j_m\}$.

The proof uses the distributive laws and **B2**.

D5. For every $\mathtt{b}$-formula ϕ there exists a $\mathtt{b}$-formula ψ of the form $\mathtt{b}_{i_1} \vee \mathtt{b}_{i_2} \vee \ldots \vee \mathtt{b}_{i_k}$ such that $\vdash_{\mathtt{b}} \phi \Leftrightarrow \psi$.

The proof is by induction on the construction of ϕ. We assume that ϕ is expressed using disjunction and negation only. The base case, of a formula of the form $\mathtt{b}_i$, is trivial. Then use of **D1** and the tautology $\phi \vee \phi \Leftrightarrow \phi$ proves **D5**.

D6. If $\vdash_{\mathtt{b}} \phi = \mathtt{b}_{i_1} \vee \mathtt{b}_{i_2} \vee \ldots \vee \mathtt{b}_{i_k}$ then $\vdash P(\phi) = P(\mathtt{b}_{i_1}) + P(\mathtt{b}_{i_2}) + \ldots + P(\mathtt{b}_{i_k})$.

For the proof,

$$P((\phi \vee \psi) \wedge \phi) + P((\phi \vee \psi) \wedge \neg\phi) = P(\phi \vee \psi)$$

Using **P4**, the first term in this equation is provably equal to $P(\phi)$. Similarly, if $\vdash_{\mathtt{b}} \neg(\phi \wedge \psi)$, then the second term is equal to $P(\psi)$. This shows $\vdash P(\phi) + P(\psi) = P(\phi \vee \psi)$ when $\vdash_{\mathtt{b}} \neg(\phi \wedge \psi)$. We apply this fact, together with **B2**, to obtain the result, using **D6** and induction.

The language $\mathcal{L}_n(P)$ captures reasoning about quantum systems in a fixed basis and is thus very restricted in expressing assertions about quantum probabilities. Our reason for such an axiomatization is firstly, to study the fragment of $\mathcal{L}_n(P, \mathbf{m})$ that closely corresponds to the classical case. Secondly, the simpler structure of this fragment simplifies the task of studying

its properties which prove useful later. Although the language $\mathcal{L}_n(P)$ is of very restricted expressive power it is instructive to go through the proofs-first, because of the close analogy with classical probabilistic logic. The axioms **P1-P4** are identical to the classical axioms. The only difference are the axioms **B1** and **B2** which capture a kind of 'internal structure' of the basis variable b. The second reason, when we extend the logic so as to include correlations amongst the probability distributions corresponding to different bases, is to bring forth the distinction between the classical and the quantum case. I first give a brief account of the classical case. I loosely follow [FHM90], and assume some familiarity with the basic definitions of measure theory. (I confine this discussion to the case when the basic propositions are measurable.)

Let Φ be the set of primitive propositions. For simplicity assume Φ to be the finite set $\{p_1, p_2, \ldots, p_m\}$. A *probability structure* is a pair consisting of a measure space (S, Ξ, μ) and a function $\mathbf{M} : \Phi \to 2^S$, where S is a set, Ξ is a Boolean ring of subsets of S closed under complementation and countable unions, and μ is a measure on Ξ. We also assume that each $\mathbf{M}(p_i)$ is measurable. The set $\mathbf{M}(p_i)$ is to be understood as the subset of S on which event p_i occurs (or p_i is true). The probability assigned to the basic proposition p_i in a probability structure is written $W(p_i)$, and defined to be the value $\mu(\mathbf{M}(p_i))$.

A formula ϕ (expressing a boolean combination of linear inequalities over probability terms) is defined to be satisfiable in the classical theory if there exists a probability structure with respect to which ϕ is true. The completeness proof proceeds by constructing a probability structure satisfying a given consistent formula ϕ. In particular, in this construction, one has a large degree of freedom in the choice of the measure space (S, Ξ, μ), as well as the interpretation function $\mathbf{M}$. This freedom is used to advantage in the completeness proof. It is first shown that the formula ϕ is equivalent to a formula ϕ' in which the basic probability terms are of the form

$W(l_1 \wedge \ldots \wedge l_m)$ where each l_i is either p_i or $\neg p_i$. A set of 2^m real variables $x_1, \ldots, x_{2^m}$ are introduced to correspond to these terms. Replacing these terms in ϕ' by their corresponding variables, and conjoining the constraints $x_i \geq 0$ and $\sum_{i=1}^{m} x_i = 1$, we obtain a consistent formula ϕ'' concerning just the real variables $x_1, \ldots, x_{2^m}$. *Any* values of the x_i satisfying these may be be used to construct a probability structure satisfying ϕ'' and the correspondence formulas $W(l_1 \wedge \ldots \wedge l_m) = x_i$, and hence ϕ. We refer to FHM for the details.

The arguments for the quantum case (for both $\mathcal{L}_n(P)$ and $\mathcal{L}_n(P, \mathbf{m})$) follow a similar structure, in that we first reduce the construction of a model for a given consistent formula to the problem of finding a set of real values x_i that correspond to a set of probabilities. However, once we obtain the values x_i we are somewhat more constrained in the way we construct a model. Our probabilities arise not from a completely undetermined measure space, but from vectors and bases in the (essentially unique) Hilbert space H_n of dimension n. Instead of constructing a measure space, we need to construct a vector and a set of bases that give rise to the values x_i through the inner product.

4.4.2 The Case of $\mathcal{L}_n(P)$

We first deal with completeness for the language $\mathcal{L}_n(P)$.

Lemma 7 *For every atomic formula Φ of $\mathcal{L}_n(P)$, with basis variables amongst $\mathtt{b}^1, \ldots, \mathtt{b}^m$, it is possible to construct in time $O(|\Phi| \cdot n)$ an atomic formula Φ^* of the form $\sum_{k=1}^{m} \sum_{j=0}^{n-1} c_{jk} P(\mathtt{b}_j^k) \leq d$, where the c_{jk} and d are integers, such that $\vdash \Phi \Leftrightarrow \Phi^*$.*

Proof: Since Φ is an atomic formula, it has the form $\sum_{j=1}^{n} a_j P(A_j) \leq d$, where the a_j and d are integers. Let $\mathtt{b}$ be the basis symbol such that A_j is a $\mathtt{b}$-formula. By **D5**, there exists a set $\{i_1, \ldots, i_r\}$ such that $\vdash_{\mathtt{b}} A_j \Leftrightarrow \mathtt{b}_{i_1} \vee \mathtt{b}_{i_2} \vee \ldots \vee \mathtt{b}_{i_r}$. Using **D1** and **D4**, the computation of this set can be

done in time $O(|A_j| \cdot n)$, following the inductive construction of A_j. By **D6** and the axiom **P4**, we have $P(A_j) = \sum_{r=1}^{n} P(\mathtt{b}_{i_r})$. Using reasoning about linear inequalities, we can now (in time $O(|A| \cdot n)$ substitute the right hand side of these equations equations in Φ, collect terms of the form $aP(\mathtt{b}_j^k)$ with the same basis term $\mathtt{b}_j^k$ into a single term of this form, and add coefficients $c_{jk} = 0$ for those j such that the corresponding $\mathtt{b}_j^k$ does not appear. This turns Φ into a basic formula of the required form. Call Φ^* the canonical form of Φ. □

First, a simple but useful fact about the group of unitary matrices.

Lemma 8 *Given any two unit vectors $|\alpha\rangle$ and $|\beta\rangle$ there is a unitary transformation U such that $U|\alpha\rangle = |\beta\rangle$. That is, the action of unitary group on the unit sphere (the set of points in $\mathbf{C}^n$ given by vectors of unit length) is* transitive.

Proof: This is easily seen as follows. Construct two bases $\mathcal{B}_\infty = \{|\alpha\rangle_0 = |\alpha\rangle, |\alpha\rangle_1, \ldots, |\alpha\rangle_{n-1}\}$ and $\mathcal{B}_\in = \{|\beta\rangle_0 = |\beta\rangle, |\beta\rangle_1, \ldots, |\beta\rangle_{n-1}\}$ say by the Gramm-Schmidt technique [Gre75]. Then,

$$|\beta\rangle_j = \sum_{i=0}^{n-1} \langle\alpha_i|\beta_j\rangle \alpha_i = \sum_i U_{ij}\alpha_i.$$

That is, $U_{ij} \stackrel{\text{def}}{=} \langle\alpha_i|\beta_j\rangle$ is the unitary matrix we seek. One can visualise this in real 3-dimensional space. A rotation which carries $|\alpha\rangle$ to $|\beta\rangle$ will suffice because rotation matrices are *real* unitary matrices. □

Theorem 6 *Let $\mathbf{x}^1 = \{x_0^1, x_2^1, \ldots, x_{n-1}^1\}$, ..., $\mathbf{x}^k = \{x_0^k, x_2^1, \ldots, x_{n-1}^k\}$ be k real vectors with non-negative components x_i^j such that $\sum_{i=0}^{n-1} x_i^j = 1$ for all $1 \leq j \leq k$ then there is a state $|\alpha\rangle$ and k bases $B_1 = \{\beta_0^1, \ldots, \beta_{n-1}^1\}$, ..., $B_k = \{\beta_0^k, \ldots, \beta_{n-1}^k\}$ with the following property:*

$$\mathtt{Tr}(|\alpha\rangle\langle\alpha| \left|\beta_i^j\right\rangle\left\langle\beta_i^j\right|) = |\langle\alpha|\beta_i^j\rangle|^2 = x_i^j.$$

Proof: Let $y_i^j = \sqrt{x_i^j}$ for all $0 \leq i \leq n-1$ and $1 \leq j \leq k$. Choose an orthonormal basis $B_1 = \{\beta_0^1, \ldots, \beta_{n-1}^1\}$ arbitrarily and let

$$|\alpha_j\rangle = \sum_i y_i^j \beta_i^1.$$

The vectors $|\alpha_j\rangle$ satisfies

$$\mathtt{Tr}(|\alpha_j\rangle\langle\alpha_j| \left|\beta_i^j\right\rangle\left\langle\beta_i^j\right|) = |\langle\alpha|\beta_i^j\rangle|^2 = x_i^j.$$

But we want a single vector α that satisfies the above equation. But we have now freedom to choose different bases. The idea is simple. Start with α_1. The basis B_1 has the required probabilities x_j^1 with respect to α_1 and the vector $|\alpha_i\rangle$ has the probabilities x_j^i with respect to the *same* basis B_1. Note that for any unitary operator U and any two vectors $|\alpha\rangle$ and $|\beta\rangle$, $(\langle\alpha|U^\dagger)(|\beta\rangle) = (\langle\alpha|)U|\beta\rangle)$ by definition of an unitary operator. That is, the inner product of the vectors $U^\dagger|\alpha\rangle = U^{-1}|\alpha\rangle$ with β equals the inner product of α with $U|\beta\rangle$. This is a rather trivial consequence of the unitarity of U and the properties of the inner product. Let U_j be a unitary operator which takes $|\alpha_1\rangle$ to $|\alpha_j\rangle$, i. e. $U_j|\alpha_1\rangle = |\alpha_j\rangle$. The lemma 8 shows that such an operator exists. Then,

$$\begin{aligned}(\langle\alpha_j|)(|\beta_i^1\rangle) =&(\langle\alpha_1|U_j^\dagger)(|\beta_i^1\rangle) = \\ &(\langle\alpha_1|)(U_j^{-1}|\beta_i^1\rangle) = y_i^j.\end{aligned}$$

Now, let B_j be the basis $\{U_j^{-1}\beta_0^1, U_j^{-1}\beta_1^1, \ldots, U_j^{-1}\beta_{n-1}^1$ for $j = 2, \ldots, k$. Then, $|\alpha\rangle = |\alpha_1\rangle$ and the bases $B_1, \ldots, B_k$ satisfy the conditions of the theorem. □

Lemma 9 *For every formula Φ in $\mathcal{L}_n(P)$ containing the basis variables $\mathtt{b}^1, \ldots, \mathtt{b}^k$ there is a formula $\Phi'[x_0^1, \ldots, x_{n-1}^1, \ldots, x_0^k, \ldots, x_{n-1}^k]$ in $\mathrm{L}_{\mathtt{INEQ}}$ such that Φ is satisfiable iff Φ' is satisfiable.*

Proof: First we introduce some notation which will prove useful later. Let $x_0, \ldots, x_{n-1}$ be n variables. We use the vector notation $\mathbf{x}$ which represents this collection of n variables. Define the formula

$$\mathbf{Prob(x)} \equiv \bigwedge_{0 \leq i \leq n} x_i \geq 0 \bigwedge \sum_i x_i = 1 \tag{4.13}$$

Now, let Φ be an atomic formula. Then from the previous lemma there is a formula $\Phi^* \equiv \sum_{k=1}^{m} \sum_{j=0}^{n-1} c_{jl} P(\mathtt{b}_j^l) \sim d$ where $\sim$ stands for either the relation $" ="$ or $" <"$ such that $\Phi \Leftrightarrow \Phi^*$ is provable in the theory of $\mathcal{L}_n(P)$. Let be the formula obtained by replacing the the probability term $P(\mathtt{b}_j^i)$ by the $\mathbb{RC}$-variables x_j^i. Let

$$\Phi' \equiv (\sum_{j=1}^{m} \sum_{j=0}^{n-1} c_{jl} x_j^l \sim d) \wedge (\wedge_i \mathbf{Prob(x}^i)) \equiv \Phi''(\mathbf{x}) \mathbf{Prob(x)}$$

where the left conjunct in the first equation is obtained by substituting $P(\mathtt{b}_j^l) = x_j^l$ in Φ^*. Now, if Φ is satisfiable then there is a quantum state $|\psi\rangle$ and bases $B^i \equiv \{\alpha_j^i | j = 0, \cdots, n-1\}$ such that if $|\langle \alpha_j^i | \psi \rangle|^2 = a_j^i$ then $\sum_{i=1}^{m} \sum_{j=0}^{n-1} c_{ij} a_j^i \sim d$. Thus, the assignment $x_j^i = a_j^i$ satisfies Φ'. Conversely, if Φ' is satisfiable then there exist numbers a_j^i such that for each i, $a_j^i \geq 0$ and $\sum_{i=1}^{m} \sum_{j=0}^{n-1} c_{ij} a_j^i \sim d$. By the theorem 6 it is then possible to construct bases $B^i \equiv \{\alpha_j^i | j = 0, \cdots, n-1\}$ such that $|\langle \alpha_j^i | \psi \rangle|^2 = a_j^i$. But this implies that Φ is satisfiable. For non-atomic formulas we construct Φ' by the following rules. If $\Phi \equiv \Phi_1 \vee \Phi_2$ then first let $Phi'' \equiv \Phi_1'' \vee \Phi_2''$, and $\Phi' \equiv \Phi''(\mathbf{x}) \wedge \mathbf{Prob(x)}$ where $\mathbf{x}$ is the succinct notation denoting the collection of all relevant vector variables. Similarly, if $\Phi \equiv \neg \Phi_1$ then $\Phi' \equiv \neg \Phi_1''(\mathbf{x}) \wedge \mathbf{Prob(x)}$. The general case follows by an easy structural induction. □

Let $\mathtt{AX_n(P)}$ be the axioms and rules of inference given above. Then we have the following:

Theorem 7 $\mathtt{AX_n(P)}$ *is a sound and complete axiomatization of* $\mathcal{L}_n(P)$.

Proof: The fact that the axiomatization is sound follows from the fact that any n-dimensional complex Hilbert space, is a model for $\mathcal{L}_n(P)$. The

verification of validity of the axioms and the inference rules is easy.

Now, completeness is equivalent to the following: a formula Ψ in the theory is satisfiable if it is consistent. By definition Ψ is consistent if $\neg\Psi$ is not a theorem. Suppose that Ψ is not satisfiable. Write Ψ in disjunctive normal form. Let $\Phi[\mathtt{b}^1,\ldots,\mathtt{b}^k]$ be an atomic subformula of Ψ. Let Φ' be the corresponding formula in $\mathcal{L}_{INEQ}$ constructed in lemma 9 and let Ψ'' be the formula obtained from Ψ by replacing atomic formulas Φ by Φ''. Let $\Psi' \equiv \Psi'' \wedge \mathbf{Prob}$. We omit the variables from these formulas for convenience. It follows that Ψ is satisfiable if and only if Ψ' is satisfiable. Suppose Ψ' is not satisfiable then $\neg\Psi'$ must be valid. This implies that $\mathbf{Prob} \Rightarrow \neg\Psi''$ must be a valid valid formula of $\mathcal{L}_{INEQ}$ Now substitute probability atoms ($P(\mathtt{b}^i_j)$ for $\mathbf{x}^i_j$). Then from the probability axioms **P1-P4**, **Prob** is provable. Since all formulas obtained by substituting probability terms in any valid formula of $\mathcal{L}_{INEQ}$ are provable by definition we conclude that Ψ' and hence $\neg\Psi''(\mathtt{b}^i/\mathbf{x}^i)$ is provable. But the latter is equivalent to $\neg\Psi$(see the previous lemma) by construction. We infer that $\neg\Psi$ is a theorem of $\mathbf{Ax}_n(P)$ and Ψ is inconsistent. □

Remark. The proof of the completeness theorem 7 for $\mathtt{AX_n(P)}$ consists of two parts. The first part is identical to the reasoning in [FHM90] for classical probabilistic logic. The purpose of the second part- the 'quantum' part- is to show that given a consistent set of 'classical' probability distributions it is always possible to find a state and orthonormal bases which yield the former. As we shall see in the next section the true quantum behaviour emerges when one fixes the relations among the bases.

The next theorem further highlights the similarity and differences between the classical probabilistic logic and $\mathcal{L}_n(P)$. The estimates of complexity are given as functions of the length of the formulas.

Theorem 8 *Satisfiability of a formula of $\mathcal{L}_n(P)$ in n-dimensional Hilbert space (with $n \geq 2$) is NP-complete.*

Proof: The restriction on dimension is clear since the 1-dimensional case is trivial. In order to show that the decision problem in the theorem is NP-hard first we show that SAT can be reduced in polynomial time to an instance of satisfiability in $\mathcal{L}_n(P)$. For any propositional formula A we construct a formula A^* in $\mathcal{L}_n(P)$. Let $\mathbf{A}$ be formula with k propositional variables $\{p_1, \ldots, p_k\}$. Let $\mathtt{b}^1$, ..., $\mathtt{b}^k$ be k basis symbols. Define A^* recursively as follows:

$$p_i^* \equiv P(b_0^i) = 1$$
$$(\neg\alpha)^* \equiv \neg(\alpha^*)$$
$$(\alpha_1 \vee \alpha_2)^* \equiv \alpha_1^* \vee \alpha_2^*$$

We now show that $\alpha \in \mathtt{SAT}$ iff α^* is satisfiable. It is easily seen that if α^* is satisfiable then so is α. For, suppose that $H_n, \pi, \psi \models \alpha^*$. Then it is immediate that the assignment $V : \{p_1, \ldots, p_m\} \to \{0, 1\}$ defined by $V(p_i) = 1$ iff $H_n, \pi, \psi \models p_i^*$ is a satisfying assignment for α. Conversely, suppose that α is satisfiable, and let V be a satisfying assignment. We construct a vector ψ in H_n and an interpretation π of the basis symbols $\mathtt{b}^1, \ldots, \mathtt{b}^m$ such that $H_n, \pi, \psi \models \alpha^*$. For this, let $B_0 = \psi_0, \ldots, \psi_{n-1}$ be any orthonormal basis of H_n. Let B_i be the sequence of vectors obtained by swapping ψ_0 and ψ_i in B_0. Clearly, B_i is also an orthonormal basis. By definition, $\langle\psi_0|\psi_i\rangle = 0$, $i \neq 0$. We now take $\psi = \psi_0$, and define π as follows: for each $i = 1 \ldots m$, we let $\pi(\mathtt{b}^i) = B_i$ if $V(p_i) = 1$ and $\pi(\mathtt{b}^i) = B_0$ otherwise. It is now straightforward to check that $H_n, \pi, \psi \models p_i^*$ iff $V(p_i) = 1$, from which it follows that $H_n, \pi, \psi \models \alpha^*$.

To show that the decision problem is in NP we may assume without loss of generality that the given formula Φ is in conjunctive normal in $\mathcal{L}_n(P)$. By Lemma 7, the formulas Φ^* and hence $\Phi^*(\mathbf{y}^1, \ldots, \mathbf{y}^k)$ may be constructed in time $O(|\phi| \cdot n)$, as may the formula $\mathtt{Prob}(\mathbf{y}^1, \ldots, \mathbf{y}^k)$. Hence we also obtain the formula $\Psi \equiv \Phi^*(\mathbf{y}^1, \ldots, \mathbf{y}^k) \wedge \mathtt{Prob}(\mathbf{y}^1, \ldots, \mathbf{y}^k)$ in time $O(|\phi| \cdot n)$. Using Lemma 9 we show that ϕ is satisfiable iff $\Phi^*(\mathbf{y}^1, \ldots, \mathbf{y}^k) \wedge \mathtt{Prob}(\mathbf{y}^1, \ldots, \mathbf{y}^k)$ is satisfiable. The latter is a boolean combination of linear constraints. Thus

each atomic subformula β in $\Phi^*(\mathbf{y}^1, \ldots, \mathbf{y}^k)$ is a linear constraint of the form $\sum_{ij} y_j^i \sim d$ where c_{ij} and d are integers. Now use a non-det determine truth assignments to the β's which make Φ^* true. For each such truth assignment we have a set of linear constraints β which have to be satisfied along with $\mathtt{Prob}(\mathbf{y}^1, \ldots, \mathbf{y}^k)$. It follows from the fact that linear programming is in PTIME [Kha79] [Kar02] that satisfiability of such formulas is in NP. Thus, satisfiability of ϕ in $\mathcal{L}_n(P)$ is also in NP. □

The same complexity was obtained in [FHM90] for their logic of classical probabilities. The part of the proof for classical probability distribution, however, is different from the one in [FHM90]. I summarise the present approach. First, identify the different basis variables $\mathtt{b}^j$ in the formula and treat each one as representing a different *classical* probability distribution. Test for satisfiability of these distributions separately. If one can find the (classical) probability distributions then the quantum counterpart can be constructed. We could do this because the distributions are independent. What if there are correlations? This is where the unitary matrices relating the bases and the corresponding transition or conditional probabilities come in.

4.4.3 Axiomatizing $\mathcal{L}_n(P, \mathbf{m})$

To capture the properties of the probability operator P, the axiomatization contains the axiomatization of b-formulas used above, and the probability axioms P1-P4. As seen above the situation is not very different from the classical picture. True quantum behaviour emerges when one incorporates the relation between two bases. This is expressed by the operators $m_{ij}(\mathtt{b}, \mathtt{c})$. Call it the transition matrix or amplitude for the bases $\mathtt{b}$ and $\mathtt{c}$. The axioms for the transition matrix follow. It is assumed that each axiom actually represents n^2 formulas- one for every pair $\{ij | 0 \leq i, j \leq n-1\}$.

M1 $\quad m_{ij}(\mathtt{b}, \mathtt{c}) = \overline{m_{ji}(\mathtt{c}, \mathtt{b})}$

M2a $m_{ij}(\mathtt{b},\mathtt{b}) = 1$ if $i = j$

M2b $m_{ij}(\mathtt{b},\mathtt{b}) = 0$ if $i \neq j$

M3 $m_{ij}(\mathtt{b},\mathtt{d}) = \Sigma_{k=1}^{n} m_{ik}(\mathtt{b},\mathtt{c}) m_{kj}(\mathtt{c},\mathtt{d})$

Axiom set **M1** expresses the fact that the the transformation matrix is *unitary*. This can be deduced in conjunction with the other axioms. **M2a** and **M2b** capture the fact that the identity transformation is given by the unit matrix of order n and **M3** expresses that given three bases b, c and d the transition matrix $m(\mathtt{b},\mathtt{d})$ is the matrix product of transition matrices $m(\mathtt{b},\mathtt{c})$ and $m(\mathtt{c},\mathtt{d})$. It thus follows from the axioms **M1-M3** that the $n \times n$ matrix whose elements are given by $m_{ij}(\mathtt{b},\mathtt{d})$ is unitary. Recall that the transition probability terms,

$$T(\mathtt{b}_i, \mathtt{c}_j) \equiv |m_{ij}(\mathtt{b},\mathtt{c})|^2$$

are defined over atomic components.

This relation between the transition probability and transition amplitude is a distinguishing characteristic of quantum systems. It is also the source of interference effects. See [Fey63] for an excellent discussion. We interpret $T(\mathtt{b}_i, \mathtt{c}_j)$ as the conditional probability of a maximal test in the c basis yielding the state $\mathtt{c}_j$ if the initial state of the system is $\mathtt{b}_i$;hence, the following axiom.

T1 $P(\mathtt{b}_i) = 1 \Rightarrow P(\mathtt{c}_j) = T(\mathtt{b}_i, \mathtt{c}_j)$

But there is a problem with the interpretation of $T(\mathtt{b}_i, \mathtt{c}_j)$ as conditional probability. Consider the following situation in *classical* probability theory. Suppose there are two set of events $A = \{a_1, \ldots, a_k\}$ and $B = \{b_1, \ldots, b_k\}$ such that the probability of the (atomic) event a_i is $p(a_i)$ and the conditional probability of the event b_j after the occurrence of a_i is $p(b_j|a_i)$. Then the probability of the event b_j is given by, $p(b_j) = \sum_i p(b_j|a_i)p(a_i)$. However, in the quantum case $P(\mathtt{c}_j) \neq \sum_i T(\mathtt{b}_i, \mathtt{c}_j) P(\mathtt{b}_i)$. Rather there are certain quadratic relations which depend upon the state. One can no longer treat the probabilities with respect to different bases as we did in the last section.

To ensure that probability assignments to different bases can be ascribed to some state, i. e. the probabilities are actually quantum probabilities a final axiom is required. This is the formula $\mathbf{MP_k}$ of Section 4.3.3. I repeat it here for the sake of convenience.

$$\begin{aligned}\mathbf{MP}_k \quad & \forall x_1 \dots x_n (Rx_1 \cdots x_n \wedge (\textstyle\bigwedge_{i=0}^{n-1} P(\mathfrak{b}_i^0) = x_i^2 \Rightarrow \\ & \exists z_0 \dots z_{n-1} (\textstyle\bigwedge_{i=0}^{n-1} |z_i| = 1 \wedge \\ & \textstyle\bigwedge_{j=1}^{k} \bigwedge_{i=0}^{n-1} P(\mathfrak{b}_i^j) = |\Sigma_{r=1}^{n} m_{ri}(\mathfrak{b}^j, \mathfrak{b}^0) x_r z_r|^2)))\end{aligned}$$

This is the most complicated axiom: one for each positive integer k. Actually, it is sufficient to require that MP_k hold up to a finite k depending on the dimension n. Then it holds for all k. This fact will be proved later.

Let $\mathbf{Ax}_n(P, \mathbf{m})$ denote the theory resulting from the above axiomatization of $\mathcal{L}_n(P, \mathbf{m})$. Recall that it is an extension of $\mathbb{RC}$ and hence one may use the properties of the latter for pure $\mathbb{RC}$ formulas. In a probabilistic logic real numbers are necessarily an integral part. Hence, any axiomatization must incorporate axioms dealing with real numbers. However, quantum probabilities arise out of *complex* amplitudes. Recall the recipe for constructing a formula of $\mathcal{L}_n(P, \mathbf{m})$. A general term in $\mathbb{RC}$ is obtained by substituting terms of the form $\sqrt{\mathfrak{t}'}$, $\overline{\mathfrak{t}'}$, or simply $\mathfrak{t}$ in the multivariate polynomial in $\{u_i | \, i = 1, \dots, r\}$,

$$\sum_{k_1 \dots k_r} a_{k_1 \dots k_r} u_1^{k_1} \dots u_r^{k_r},$$

where $a_{k_1 \dots k_r}$ are constants of $\mathbf{RC}$ and $\mathfrak{t}'$ is free of the defined function symbols$(, \sqrt{})$. For any term $\mathbf{t}$, if $\mathbf{t} \geq 0$ or $\mathbf{t} < 0$ is a formula then substitute probability terms $P(\phi)$, $\mathbf{m}$-terms, and defined terms like the transition probability terms $T(\mathfrak{b}_i, \mathfrak{c}_j)$ *uniformly* for some of the variables x_i. One obtains an atomic formula of $\mathcal{L}_n(P, \mathbf{m})$. It is clear that a formula of $\mathbf{RC}$ is also a formula of $\mathcal{L}_n(P, \mathbf{m})$. Several properties of $\mathbb{RC}$ were proved in Section 4.2. I remind the reader that probabilistic statements about quantum systems are usually about relations among real and complex numbers. That is, some formulas in $\mathbb{RC}$. It is natural that we look for a systematic reduction of formulas in $\mathcal{L}_n(P, \mathbf{m})$ to those of $\mathbb{RC}$. As the first step we prove the following.

Let the length of a formula ϕ, denoted by $|\phi|$, be the number of symbols appearing in the formula.

Lemma 10 *Let Φ be a formula of $\mathcal{L}_n(P, \mathbf{m})$ containing the basis variables $\mathtt{b}^1, \ldots, \mathtt{b}^m$. Then it is possible to construct in polynomial time a formula Φ^* such that $\vdash \Phi \Leftrightarrow \Phi^*$ and all atomic subformulas of Φ^* are of the form $p = 0$ where p is a polynomial with constant coefficients over terms of the form $P(\mathtt{b}_i^r)$, or $m_{ij}(\mathtt{b}^1, \mathtt{b}^r)$ or their conjugates with $1 \leq r \leq k$ and $1 \leq i, j \leq n$ or a formula of $\mathbb{RC}$.*

Proof: From the remarks preceding the lemma Φ is constructed by appropriate substitution of probability terms, matrix terms, etc. for variables in a term of $\mathbb{RC}$ which may contain the square root function. First, eliminate the square roots by the defining formula **D** (see Equation 4.7). This elimination process can be done in polynomial time. Thus, one may assume that that Φ is square root free. It suffices to show that each atomic subformula Ψ of Φ is equivalent to a formula Ψ^* of the required form. First, we eliminate $<$ and $>$ by adding x^2 for some new real variable x. For example, $\mathtt{t} > 0$ is provably equivalent to $\exists x(Rx \wedge \mathtt{t} = x^2)$. Next, note that by the arguments of Lemma 7, all probability terms $P(\gamma)$, with γ a $\mathtt{b}^r$ formula, are provably equivalent to a sum of terms of the form $P(\mathtt{b}_i^r)$. Also all transition probability terms $T(\mathtt{b}_i^r, \mathtt{b}_j^s)$ can be reduced to transition matrix terms and hence, it suffices to consider only matrix terms: $m_{ij}(\mathtt{b}^r, \mathtt{b}^s)$. Using **M3**, we may express the terms $m_{ij}(\mathtt{b}^s, \mathtt{b}^r)$ as a sum of terms of the form $\overline{m_{ik}(\mathtt{b}^s, \mathtt{b}^1)} \cdot m_{jk}(\mathtt{b}^1, \mathtt{b}^r)$. By **M1**, the terms $m_{ik}(\mathtt{b}^s, \mathtt{b}^1)$ are equal to $\overline{m_{ki}(\mathtt{b}^1, \mathtt{b}^s)}$.

The result of these transformations is to show that Ψ is equivalent to a formula composed from real and complex variables and terms of the form $P(\mathtt{b}_i^r)$ and $m_{ij}(\mathtt{b}^1, \mathtt{b}^r)$ using addition and multiplication. □

The intuition of the preceding lemma is the following. If there are r basis variables then each pair gives rise to corresponding matrix terms. There are

thus, $r(r-1)/2$ sets of matrix terms. We choose some basis arbitrarily, say, $\mathtt{b}^1$ and consider only transformation matrices $m_{ij}(\mathtt{b}^1, \mathtt{b}^k)$. All other transformation matrices can be generated from these. Of course, in some actual implementation of the algorithms for the reduction some heuristics for this choice would be useful.

Before giving the proof of completeness of the axiomatization $\mathbf{Ax_n}(P, \mathbf{m})$ let us discuss informally the ideas behind the proof. First, we assume that a consistent formula Φ of $\mathcal{L}_n(P, \mathbf{m})$ containing the basis symbols $\mathtt{b}^1, \ldots, \mathtt{b}^k$ is given. Our goal is to find an interpretation of the basis variables as basis components and a unit vector(the state) in H_n with respect to which Φ is satisfied. Using Lemma 10, it suffices to show that Φ^* is satisfiable. For $1 \leq j \leq k$ and $0 \leq i \leq n-1$, let x_i^j be variables, such that Rx_i^j is true(intuitively x_i^j are real variables). For each $1 \leq j \leq k$ and $0 \leq i, r \leq n-1$, let y_{ir}^j be a variable. Write $\mathbf{x}_i^j$ for the s $x_0^j \ldots x_{n-1}^j$ and write $\mathbf{y}$ for the sequence of variables y_{ir}^j. Note that $\mathbf{x}$ and $\mathbf{y}$ are not variables of the object language or $\mathbb{RC}$ but only a shorthand notation for a sequence of variables. Let θ be the substitution that puts $P(\mathtt{b}_i^j)$ for each x_i^j, and substitutes $m_{ir}(\mathtt{b}^1, \mathtt{b}^j)$ for each y_{ir}^j.

Lemma 11 *Given a set of basis variables $\mathtt{b}^1, \ldots, \mathtt{b}^k$, there is a formula $\tilde{\Phi}$ of $\mathbb{RC}$ with free variables amongst $\mathbf{x}, \mathbf{y}$ such that*

1. *$|\tilde{\Phi}|$ is of polynomial order in $|(\mathbf{x}, \mathbf{y})|$,*

2. *if π is an interpretation of the real and complex variables $\mathbf{x}$ and $\mathbf{y}$ that satisfies $\tilde{\Phi}$, then there exists a vector $|\psi\rangle$ of H_n and an extension of π to an interpretation for the basis symbols $\mathtt{b}^1, \ldots, \mathtt{b}^k$ in H_n, such that*

   (a) $[\![P(\mathtt{b}_i^j)]\!]_{\pi,|\psi\rangle} = \pi(x_i^j)$ for $1 \leq j \leq k$ and $1 \leq i \leq n$, and

   (b) $[\![m_{ir}(\mathtt{b}^1, \mathtt{b}^j)]\!]_{\pi,|\psi\rangle} = \pi(y_{ir}^j)$ for $1 \leq j \leq k$ and $1 \leq i, r \leq n$.

3. *$\vdash \tilde{\Phi}\theta$,*

Proof: Write $\mathbf{x}^j$ for the sequence of variables $x_1^j, \ldots, x_n^j$. Recall that $\mathtt{Prob}(\mathbf{x}^j)$ is the formula

$$\bigwedge_i x_i^j \geq 0 \bigwedge \sum_{i=0}^{n-1} x_i^j = 1$$

Let $\mathtt{Prob}(\mathbf{x}^1, \ldots, \mathbf{x}^k)$ stands for $\bigwedge_{j=1}^k \mathtt{Prob}(\mathbf{x}^j)$. Further, for each $j = 1 \ldots, k$, define $\mathtt{Unitary}(\mathbf{y}^j)$ to be the formula

$$\bigwedge_{i=0}^{n-1} \bigwedge_{r=0}^{n-1} \sum_{s=0}^{n-1} y_{is}^j \cdot \overline{y_{rs}^j} = \delta_{ir}$$

. Define $\mathtt{Phase}(z_0, \ldots, z_{n-1}, \mathbf{x}, \mathbf{y})$, where the z_i are variables, to be the formula

$$\bigwedge_{i=0}^{n-1} (|z_i| = 1) \ \wedge \ \bigwedge_{j=2}^{k} \bigwedge_{i=0}^{n-1} |\sum_{r=0}^{n-1} y_{ir}^j z_r \sqrt{x_r^1}|^2 = x_i^j. \tag{4.14}$$

We now take $\tilde{\Phi}(\mathbf{x}, \mathbf{y})$ to be the conjunction of the formulas $\mathtt{Prob}(\mathbf{x}^j) \wedge \mathtt{Unitary}(\mathbf{y}^j)$, for $j = 1 \ldots k$, with the formula $\exists z_0, \ldots, z_{n-1}(\mathtt{Phase}(z_0, \ldots, z_{n-1}, \mathbf{x}, \mathbf{y}))$. Clearly $|\tilde{\Phi}(\mathbf{x}, \mathbf{y})|$ is of polynomial order in $|(\mathbf{x}, \mathbf{y})|$. In fact, $|\tilde{\Phi}(\mathbf{x}, \mathbf{y})| = O(k \cdot n^2)$ and for fixed n $|\tilde{\Phi}(\mathbf{x}, \mathbf{y})| = O(|(\mathbf{x}, \mathbf{y})|)$. So the first condition of the lemma is satisfied.

We next show that the second condition is satisfied. Let π be an assignment of real and complex numbers to the variables $\mathbf{x}$ and $\mathbf{y}$ such that $\tilde{\Phi}(\mathbf{x}, \mathbf{y})$ is satisfied. Moreover, we let π assign complex numbers to the variables $z_0, \ldots, z_{n-1}$ such that the formula $\mathtt{Phase}(z_0, \ldots, z_{n-1}, \mathbf{x}, \mathbf{y})$ is satisfied. The fact that such an assignment is always possible follows from the transitive action of the unitary group (see Theorem 6). In other words the formula $\exists z_0 \ldots z_{n-1} \mathtt{Phase}(z_0, \ldots, z_{n-1}, \mathbf{x}, \mathbf{y})$ is satisfiable. Now extend the interpretation (continue to call it π) for $\mathbf{b}^1, \ldots, \mathbf{b}^k$ in H_n and the vector $|\psi\rangle$ in H_n as follows. For $\pi(\mathbf{b}^1)$ take any orthonormal basis $|\epsilon_1\rangle, \ldots, |\epsilon_n\rangle$. For the remaining bases $\mathbf{b}^j$, with $2 \leq j \leq k$, define

$$\pi(\mathbf{b}^j)_i = \sum_{r=0}^{n-1} \pi(y_{ir}^j) \cdot |\epsilon_r\rangle .$$

We take

$$|\psi\rangle = \sum_{r=0}^{n-1} \pi(z_r) \cdot \sqrt{\pi(x_r^1)} \cdot |\epsilon_r\rangle \,.$$

This is a unit vector because, by assumption, we have that $\|\pi(z_r)\| = 1$ and $\sum_{r=0}^{n-1} \pi(x_r^j) = 1$. The intuition behind the above formula is that we express the state as a unit vector represented in the basis $\mathtt{b}^1$, the squared amplitude of the coefficients are the probabilities and the phases are the complex numbers z_i of modulus 1.

We show that the two parts of condition (2) of Lemma 11 are satisfied. The second part is immediate from the definition of the $\pi(\mathtt{b}^j)_i$. For the first part, note that

$$\begin{aligned}
&[\![P(\mathtt{b}_i^j)]\!]_{\pi,|\psi\rangle} \\
&\quad = \|\langle \pi(\mathtt{b}_i^j)|\psi\rangle\|^2 \\
&\quad = |\ \textstyle\sum_{r=0}^{n-1} \pi(y_{ir}^j) \cdot \pi(z_r) \cdot \sqrt{\pi(x_r^j)}\ |^2 \\
&\quad = \pi(x_i^j)
\end{aligned}$$

where the last step follows from the fact that π satisfies $\mathtt{Phase}(z_1, \ldots, z_n, \mathbf{x}, \mathbf{y})$.

The third condition $\vdash \tilde{\Phi}\theta$ is a consequence of the axioms for the probability operator and transformation matrix. By **D6** and **P1, P2**, we have that $\vdash \mathtt{Prob}(\mathbf{x}^j)\theta$ for each $j = 1 \ldots k$. It follows from **M1** and **M4** that $\vdash \mathtt{Unitary}(\mathbf{y}^j)\theta$ for each $j = 1 \ldots k$. By the axiom $\mathbf{MP_k}$ we have $\vdash \exists z_1 \ldots z_n \mathtt{Phase}(z_1, \ldots, z_n, \mathbf{x}, \mathbf{y})\theta$. Thus, each of the conjuncts of $\tilde{\Phi}\theta$ is derivable, so this formula itself is derivable. □

The lemma is the first step toward reducing the problem of satisfiability in $\mathcal{L}_n(P, \mathbf{m})$ to that in $\mathbb{RC}$. There are two primary reasons for this reduction: first, it will be used to prove important properties of the former, second it will allow us to use decision methods for real closed field (see [BKR86], [Can88], [BPR03]).

Lemma 12 *With the above notation let Φ be formula of $\mathcal{L}_n(P, \mathbf{m})$ with basis variables $\mathtt{b}_1, \ldots, \mathtt{b}_m$. Let the formula $\tilde{\Phi}(\mathbf{x}, \mathbf{y})$ in $\mathbb{RC}$ be constructed with $\mathbf{x}$*

and $\mathbf{y}$ *as above. Then there is a* $\mathbb{RC}$*-formula* Φ'' *such that* $\Phi' \stackrel{\text{def}}{=} \Phi'' \wedge \tilde{\Phi}(\mathbf{x}, \mathbf{y})$ *is satisfiable iff* Φ *is satisfiable. The algorithm for translating* Φ *into* Φ' *is in* **PTIME**$(|\Phi)$.

Proof: First, note that the formula $\tilde{\Phi}(\mathbf{x}, \mathbf{y})$ can be always constructed, it depends only on the basis variables appearing in Φ. Hence, we have to deal with Φ'' only. We will define Φ'' recursively.

1. Φ atomic. Since Φ is atomic it must be obtaibed from $p(z_1, z_2, \ldots, z_k) = 0$ or $p(z_1, z_2, \ldots, z_k) < 0$ or $R(p(z_1, z_2, \ldots, z_k))$, by substituting $P(\phi_i[\mathfrak{b}^{k_i}])$ for some z_i. We also allow the substitution of matrix symbols $m_{ij}(\mathfrak{b}^i, \mathfrak{b}^j)$ for some of the variables z_i. Now, we assume that the basis formulas are in the canonical form: $\phi_i = \mathfrak{b}^{k_i}_{j_1} \vee \ldots \vee \mathfrak{b}^{k_i}_{j_i}$. Moreover, from Lemma 10 we may confine to matrix symbols of the form $m_{ij}(\mathfrak{b}^1, \mathfrak{b}^j)$, that is transformations from some arbitrary but fixed basis. Recall that the "vector" variable $\mathbf{x}^i$ is used to denote a set of n variables corresponding to the basis components $\mathfrak{b}^i_j$. Thus, we obtain Φ'' by making the following substitutions in the polynomial p:

   (a) For $m_{kl}(\mathfrak{b}^i, \mathfrak{b}^j)$ substitute $\sum_r y^i_{kr} \overline{y_{lr}}^j$. The intuition is that if $U_i(U_j)$ represent the transformation matrices from the basis $\mathfrak{b}^1$ to the basis $\mathfrak{b}^i(\mathfrak{b}^j)$ then $U_i U_j^\dagger$ is the corresponding transformation from $\mathfrak{b}^i$ to $\mathfrak{b}^j$.

   (b) Convert the basis formula into canonical form as disjunction over the basis variables.

   (c) If a basis formula (in canonical form) is $\phi[\mathfrak{b}^i] = \vee_k \mathfrak{b}^i_{j_k}$ then substitute $\sum_k x^i_{j_k}$ for $P(\phi)$. The intuition for this substitution is that: $P(\phi) = \sum_k P(\mathfrak{b}^i_{j_k})$ (formally, $\vdash P(\phi) = \sum_k P(b^i_{j_k})$; see **D6**).
   In this algorithm for constructing Φ'' we take the obvious precaution that variables used in the substitution are different from those in the original formula Φ.

(d) If Φ is a formula of **RC** then $\Phi'' = \Phi$.

2. If Φ is $\neg\Phi_1$ for some formula Φ_1 then $\Phi'' = \neg\Phi_1''$.

3. If Φ is $\Phi_1 \wedge \Phi_2$ for then $\Phi'' = \Phi_1'' \wedge \Phi_2''$.

4. If Φ is $\exists x\Phi$ then $\Phi'' = \exists x\Phi''$.

Next we show that Φ is satisfiable if and only if $\Phi' \equiv \tilde{\Phi} \wedge \Phi''$ is satisfiable. We prove this using the recursive construction above. A formal proof may be given by induction on the structure of formulas. It is left to the reader. We only remark that the satisfiability of $\Phi = \exists x\Phi_1$ is equivalent to that of Φ_1.

We show now that Φ is satisfiable if and only if Φ' is satisfiable. The proof is essentially an application of Lemma 11. The formula Φ' is $\tilde{\Phi} \wedge \Phi'$. The subformula $\tilde{\Phi}$ assigns variables to the probability terms of the $P(\mathsf{b}_i^j)$ and matrix terms $m_{ij}(\mathsf{b}^1, \mathsf{b}^k)$ and by the Lemma mentioned above it is satisfiable. Any assignment of real and complex numbers to the variable in $\tilde{\Phi}$ that satisfies the probability and unitary constraints yields an interpretation π of the probability and matrix terms. To be precise this interpretation is given as follows. The basis b^1 may be taken to be any set of orthonormal vectors. The other bases b^j are then given by $\pi(\mathsf{b}_j^i) = \sum_l \pi(y_{lj}^i \pi(\mathsf{b}_l^1))$. The state is given by $\psi = \sum_j \pi(\sqrt{x_j})\pi(\mathsf{b}_j^1)$. Now substituting these values in Φ gives an interpretation of the $\mathbb{RC}$ formula Φ''. Hence, Φ evaluates to true in the state ψ if Φ'' does. The converse is straightforward. We simply take the give interpretation of the probability terms $P(\mathsf{b}_j^i)$ and the matrix terms $m_{ij}(\mathsf{b}^1, \mathsf{b}^k)$ as interpretation of x_j and y_{ij}^k respectively. The proof is complete. □

From the two preceding lemmas the following theorem can be inferred.

Theorem 9 $\mathtt{AX_n}(\mathtt{P}, \mathbf{m})$ *is a sound and complete axiomatization for the language* $\mathcal{L}_n(P, \mathbf{m})$.

Proof: Soundness of $\mathtt{AX}_n(P, \mathbf{m})$ is a straightforward verification that the axioms are valid in any structure. Given a formula of Φ of $\mathcal{L}_n(P, \mathbf{m})$ let $\Phi' \equiv \tilde{\Phi} \wedge \Phi''$ be the formula constructed in the previous lemma. Suppose Φ is unsatisfiable. Then from the last lemma Φ' is unsatisfiable. Let

$$\Psi \equiv \bigwedge_{ijk}(P(\mathsf{b}_j^i) = x_j^i \bigwedge(\bigwedge_{lr}(m_{lr}(\mathsf{b}^1, \mathsf{b}^k) = y_{lr}^k))$$

be the formula which simply assigns appropriate variables to probability and matrix terms. Recall that the new variables x_j^i and y_{lm}^k in Ψ and Φ' are different from those in Φ. Then, as a simple consequence of equality axioms and $\mathbf{MP_k}$ we have $\Psi \Rightarrow \tilde{\Phi}$ is derivable. Moreover, from **D6**, the definition of Φ'' and equality axioms it follows that $\Psi \wedge \Phi \Rightarrow \Phi'$ is derivable. This can be proved formally by structural induction. As Φ' is unsatisfiable $\neg\Phi'$ is a valid formula of $\mathbb{RC}$. Hence, it is derivable because $\mathbb{RC}$ is a complete theory. It follows that $\Psi \Rightarrow \neg\Phi$ is a theorem. Since the variables appearing in Ψ are distinct from those in Φ it follows from the repeated application of $\exists$-introduction rule of first order logic [Sho67] that

$$\exists \mathbf{x}^1 \ldots \mathbf{x}^k \mathbf{y}^1 \ldots \mathbf{y}^k \Psi \Rightarrow \neg\Phi$$

is a derivable. Here the expression $\exists \mathbf{x}^1 \ldots \mathbf{x}^k \mathbf{y}^1 \ldots \mathbf{y}^k$ is a compact notion for existential quantification over all relevant variables x_j^i and y_{rs}^l appearing in Ψ. But $\exists \mathbf{x}^1 \ldots \mathbf{x}^k \mathbf{y}^1 \ldots \mathbf{y}^k \Psi$ is an instance of an axiom of a theory with equality. Hence, $\neg\Phi$ is derivable. That is Φ is not consistent. The theorem is proved. □

In the above proofs axiom $\mathbf{MP}_k$ plays crucial role. As stated, this is an infinite schema, one for each positive integer k. It will be shown below that it is sufficient to take k up to a finite number(depending on the dimension). But I first demonstrate that the axiom is necessary by an example.

Example

Let the dimension $n = 2$ and define unitary matrices

$$U_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & i \\ i & 1 \end{pmatrix} \text{ and } U_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}.$$

Then the following probability and unitary assignment is unsatisfiable.

$$P(\mathfrak{b}_0^0) = 1/2 \wedge P(\mathfrak{b}_0^1) = 1/3 \wedge P(\mathfrak{b}_0^2) = 3/4 \bigwedge$$
$$(\bigwedge_{ij}(m_{ij}(\mathfrak{b}^0, \mathfrak{b}^1) = U_1(ij) \wedge m_{ij}(\mathfrak{b}^0, \mathfrak{b}^2) = U_2(ij)))$$

This can be verified by an easy calculation. Although it is impossible to satisfy the above formula for three bases with the given unitary transformations it is possible to satisfy any two of them. However, for any finite collection of basis symbols with any given unitary transformations if the probability assignments are satisfiable for any *three* bases then the whole collection is jointly satisfiable. This will follow as a special case (in dimension 2) of Theorem 10 below.

Theorem 10 *Let the formula*

$$\begin{aligned} \mathbf{MP}_k \equiv\ & \forall x_1 \dots x_n (Rx_1 \cdots x_n \wedge (\textstyle\bigwedge_{i=0}^{n-1} P(\mathfrak{b}_i^0) = x_i^2) \Rightarrow \\ & \exists z_0 \dots z_{n-1} (\textstyle\bigwedge_{i=0}^{n-1} |z_i| = 1 \wedge \\ & \textstyle\bigwedge_{j=1}^{k} \bigwedge_{i=0}^{n-1} P(\mathfrak{b}_i^j) = |\Sigma_{r=1}^{n} m_{ir}(\mathfrak{b}^j, \mathfrak{b}^0) x_r z_r|^2)) \end{aligned}$$

be an axiom for all positive integer $k \leq n^2 - n + 1$, n the dimension. Then for all k it is a theorem of $\mathtt{AX_n}(P, \mathbf{m})$.

Proof: First, if $k \leq n^2 - n + 1$ then the formula $\mathbf{MP}_k$ is an axiom. We show that for larger k, the formula also follows. Let $k \geq n^2 - n + 1$. Then, to prove the theorem it suffices to show that the

$$\vdash \mathbf{MP}_k \text{ implies } \vdash \mathbf{MP}_{k+1}$$

We first show the validity of a formula $\mathbf{F}_k$ of $\mathbb{RC}$. Recall that we use the vector notation $\mathbf{y}^r$ as a shorthand for a group of variables y_{ij}^r, $i, j = 0, \dots, n-1$

(see Lemma 11). For $k+1$ such variables $\{\mathbf{y}^1, \ldots, \mathbf{y}^k\}$ define first the formula

$$\mathbf{E}_{k,r} \equiv \exists z_0 \ldots z_{n-1}(\bigwedge_{i=0}^{n-1} |z_i| = 1 \bigwedge \ (\bigwedge_{j=1, j\neq r}^{k+1} \bigwedge_{i=0}^{n-1} x_j^i = |\Sigma_{r=0}^{n-1} y_{ir}^j x_r z_r|^2)) \text{ and}$$

$$\mathbf{E}_k \equiv \exists z_0 \ldots z_{n-1}(\bigwedge_{i=0}^{n-1} |z_i| = 1 \bigwedge \ (\bigwedge_{j=1}^{k+1} \bigwedge_{i=0}^{n-1} x_j^i = |\Sigma_{r=0}^{n-1} y_{ir}^j x_r z_r|^2))$$

Now define $\mathbf{F}_k$ as

$$\mathbf{F}_k \equiv (\bigwedge_j R x_0^j \cdots x_{n-1}^j \wedge (\bigwedge_{j=1}^{k+1} \sum_i (x_i^j)^2 = 1 \wedge \mathbf{Unitary}(\mathbf{y}^j)) \wedge \bigwedge_{r=1}^{k} \mathbf{E}_{k,r}) \Rightarrow \mathbf{E}_k \tag{4.15}$$

Suppose we have shown that $\mathbf{F}_k$ is a theorem of $\mathbb{RC}$. Substitute $\mathbf{M}^j$ for $\mathbf{y}^j$ where $\mathbf{M}^j$ is shorthand for the collection of matrix terms $m_{ir}(\mathfrak{b}^j, \mathfrak{b}^0)$, similar to the definition of $\mathbf{y}^j$. Then

$$\bigwedge_{\substack{i=0 \\ j=1}}^{\substack{k \\ n-1}} P(\mathfrak{b}_i^j) = (x_i^j)^2 \Rightarrow (\bigwedge_j (\sum_i (x_i^j)^2 = 1 \wedge \mathbf{Unitary}(\mathbf{M}^j)) \bigwedge$$

$$\exists z_0 \ldots z_{n-1}(\bigwedge_{i=0}^{n-1} |z_i| = 1 \bigwedge \ \bigwedge_r \bigwedge_{j=1, j\neq r}^{k+1} (\bigwedge_{i=0}^{n-1} x_j^i = |\Sigma_{r=0}^{n-1} m_{ir}(\mathfrak{b}^0, \mathfrak{b}^j) x_r z_r|^2)) \Rightarrow$$

$$\exists z_0 \ldots z_{n-1}(\bigwedge_{i=0}^{n-1} |z_i| = 1 \bigwedge \bigwedge_{j=1}^{k+1} (\bigwedge_{i=0}^{n-1} x_j^i = |\Sigma_{r=0}^{n-1} m_{ir}(\mathfrak{b}^0, \mathfrak{b}^j) x_r z_r|^2)))$$

(4.16)

The formula is a theorem of $\mathbf{Ax}_n(P, \mathbf{m})$ if $\mathbf{F}_k$ is a theorem since we may infer $\vdash A \Rightarrow B$ from $\vdash B$. The formula simply asserts that if the probability assignments $P(\mathfrak{b}_i^j) = (x_i^j)^2$ satisfy the quantum probability condition(the subformula on the right of second implication) for every k-subset of $k+1$ unitary matrices then the condition is satisfiable for all the $k+1$ matrices. Now, from the probability, equality and unitary axioms it follows that

$$\bigwedge_{\substack{i=0 \\ j=1}}^{\substack{k \\ n-1}} P(\mathfrak{b}_i^j) = (x_i^j)^2 \Rightarrow \bigwedge_j (\sum_i (x_i^j)^2 = 1) \bigwedge (\bigwedge_j \mathbf{Unitary}(\mathbf{M}^j))$$

Suppose also that $\vdash \mathbf{MP}_k$. Then by simple substitutions in $\mathbf{MP}_k$ we get

$$\bigwedge_{i,j} P(\mathfrak{b}_i^j) = (x_i^j)^2 \Rightarrow$$

$$\exists z_0 \ldots z_{n-1} (\bigwedge_{i=0}^{n-1} |z_i| = 1 \bigwedge \bigwedge_r \bigwedge_{j=1, j\neq r}^{k+1} (\bigwedge_{i=0}^{n-1} x_j^i = |\Sigma_{r=0}^{n-1} m_{ir}(\mathfrak{b}^j, \mathfrak{b}^0) x_r^0 z_r|^2))$$

Combining the preceding two formula, and from the fact that formula 4.16 and $\mathbf{MP}_k$ are theorems it follows by the detachment rule(see [Sho67]) that $\vdash \mathbf{MP}_k$. Hence to prove the theorem it suffices to show that $\mathbf{F}_k$ is a theorem for $k \geq n^2 - n + 1$. Let us briefly discuss the formula $\mathbf{F}_k$ in the context of quantum probability. A formula like $\exists z_0 \ldots z_{n-1} \bigwedge_{i=0}^{n-1} |z_i| = 1 \bigwedge (\bigwedge_{j=1}^{k+1} \bigwedge_{i=0}^{n-1} x_j^i = |\Sigma_{r=0}^{n-1} y_{ir}^j x_r z_r|^2)$ asserts the existence of "phases" z_i of a state vector. Recall that a state is a unit vector in an n-dimensional complex vector space. Choose any basis $\{\alpha_0, \ldots, \alpha_{n-1}\}$. Then the state Ψ can be written as $\Psi = \sum_i c_i \alpha_i$. Write the complex coordinates as $c_i = x_i z_i$ where $|c_i| = x_i$. The probability of getting the i^{th} outcome in a measurement in the α-basis is x_i^2. Let $m_{ir}(\mathfrak{b}^0, \mathfrak{b}^j) = y_{ir}^j = p_{ir}^j \cdot e^{\imath \beta_{ir}^j}$, with $p_{ir}^j \geq 0$ be the polar form of the entries of the unitary matrix. Denote this unitary matrix by U_j and let U_0 denote the $n \times n$ unit matrix. For this proof it will be convenient to use the column matrices for vectors. Thus let,

$$|\psi\rangle \equiv \begin{pmatrix} c_0 \\ \vdots \\ c_{n-1} \end{pmatrix}$$

be the unknown vector whose existence we have to ascertain. The first set of equations with respect to the basis, say $\mathfrak{b}^0$, fixes the modulus $|c_i| = x_i^0$. It is convenient to write in polar form. Thus $c_i = x_i^0 e^{\imath\theta_i}$($z_i = e^{\imath\theta_i}$) for some real arguments θ_i to be determined. The formulas $\mathbf{F}_k$ yield equations in the variables z_i. Thus, for each k we have,

$$|\sum_j y_{ij}^k c_j|^2 = (x_i^k)^2, \quad i = 0, \ldots, n-1$$

For each k th basis corresponding to $\mathfrak{b}^k$ provides a set of $n-1$ equations. Call this set the block B_j. To show that the desired formula is valid, we need to show that a set of equations in the unknwons z_i has a solution. Let $y^j_{ir} = p^j_{ir}e^{\imath\beta^j_{ir}}$ in polar form. Then the set of equations to be satisfied by $z_i = e^{\imath\theta_i}$ is given by,

$$|\sum_s p^j_{si}e^{\imath(\beta^j_{ri}+\theta_r)}r^0_s|^2 = (x^j_i)^2, \tag{4.17}$$

where $1 \leq i, r \leq n$ and $1 \leq j \leq k$. This equation can be written as

$$\begin{array}{c}\sum_{l<s} p^j_{li}p^j_{si}r^0_l r^0_s \cos(\beta^j_{li} - \beta^j_{si} + \theta_l - \theta_s) \\ = \frac{1}{2}((x^j_i)^2 - \sum_l (p^j_{li})^2(x^0_l)^2)\end{array}$$

Writing the right side of the above equation as q^j_i we rewrite it as

$$\begin{array}{c}\sum_{l<s} p^j_{li}p^j_{si}r^0_r r^0_s(\cos(\beta^j_{li} - \beta^j_{si})\cos(\theta_l - \theta_s) \\ -\sin(\beta^j_{li} - \beta^j_{si})\sin(\theta_r - \theta_s)) = q^j_i.\end{array} \tag{4.18}$$

If one treats $\cos(\theta_l - \theta_s)$ and $\sin(\theta_l - \theta_s)$, for $1 \leq l < s \leq n$, as independent variables then (4.18) represents a set of linear equations. However, these variables are quadratically constrained by the relations $\cos^2(\theta_l - \theta_s) + \sin^2(\theta_l - \theta_s) = 1$ and the (quadratic) relations among $\cos(\theta_l - \theta_s)$ and $\sin(\theta_l - \theta_s)$) and $\cos(\theta_l)$, $\cos(\theta_s)$ $\sin(\theta_l)$, and $\sin(\theta_s)$. Thus despite the appearance of transcendental functions $\cos(\theta_l - \theta_s)$, the equations (4.18) is a set of algebraic (in fact quadratic) equations.

There are $n(n-1)/2$ variables each of type $\cos(\theta_l - \theta_s)$ and $\sin(\theta_l - \theta_s)$, thus a total of $n(n-1)$ variables. Note that the 'diagonal' terms are missing from the left side. Let us generalise slightly and assume that there are $n-1$ more variables $\rho_{11}, \ldots \rho_{n-1\,n-1}$ whose coefficients are 0 in the above equation. The reason for this step will become clear in the next chapter when I generalise the interpretation to "mixed" states. The present proof goes over almost *verbatim* to the general case.

We have assumed that $\mathbf{F}_k$ holds for $k \leq n^2 - n + 1$ bases. As we have seen above there is a block B_j of $n-1$ equations corresponding to each

basis. Hence, for $k > n^2 - n + 1$, of the k possible blocks any subset of $n(n-1)+1$ equation has a solution. Now suppose that the entire set of equations does not have a solution. I prove a contradiction. Since there are $n(n-1)+n-1 = n^2-1$ variables in the blocks, the maximum possible rank of the coefficient matrix is $\leq n^2-1$ (actually it is $\leq n^2-n$ in the present case because the last $n-1$ columns are zero). Adding a block to an existing set of blocks, the rank of the corresponding coefficient matrix either remains equal to the rank of matrix of coefficients of A or it increases by at least 1. Moreover, any particular block of equations has rank $n-1$ because the coefficient matrix in this case is given by a set of $n-1$ projection operators $|\alpha_1\rangle\langle\alpha_1|, \ldots |\alpha_{n-1}\rangle\langle\alpha_{n-1}|$ and they are independent. Hence, starting with $n-1$ independent equations with respect to the first block add a new block if the addition of the latter increases the rank. That is, suppose we already have a set S_r of blocks, say $S_r = \{B_0, \ldots, B_r\}$ then add B_{r+1} if the addition increases the rank. Otherwise, the set of equations corresponding to B_{r+1} are linearly dependent on the blocks already in S_r and we discard it. Let rank(S_r) denote the rank of the coefficient matrix of the equations in S_r. By this construction rank$(S_r) \geq n-1+r-1 = n+r-2$ Since the maximum possible rank is n^2-1, $r \leq n(n-1)$ and S_r is consistent by assumption. Continuing this process two things can happen before all k blocks are exhausted.

Case 1. The rank remains less than $n(n-1)$ and all k blocks are exhausted. In this case $r < n^2-n+1$ and by assumption the system S_r has a solution. The discarded blocks must be all consistent with this solution because an arbitrary block B_{r+1} is linearly dependent upon S_r and $S_r \bigcup B_{r+1}$ is consistent (as $r+1 \leq n^2-n+1$).

Case 2. The rank becomes equal to the maximum value $n(n-1)$ for some S_r, $r \leq k$. Then, since $r < n(n-1)+1$ the system S_r has a solution and due to the maximality of rank this solution is *unique*. Now, let B_p be B_q two blocks not in S_r and consider the set of equations $W_1 = S_r \bigcup B_p$ and

$W_2 = S_r \bigcup B_q$. Since $r < n(n-1)+1$ then both W_1 and W_2 have a solution and by the uniqueness of solution it is the common solution of the system S_r. As B_p and B_q are arbitrary, there is a (unique) solution to the whole set. □

I have stated this result as a theorem because I feel that it is of significance for physics as well as the logic of quantum probabilities. It shows that to determine whether a set of numbers can have arisen as the probabilities associated to a set of k bases in n-dimensional Hilbert space, it suffices to check the probabilities associated to every subset of size $n^2 - n + 1$.

Corollary 1 *In the present case where quantum states are required to be pure the above bound on k can be improved to $n^2 - 2n + 1$.*

Proof: In the proof of the theorem $n - 1$ extra variables were added but the actual maximum rank was $n^2 - n$. By an argument very similar to above one obtains the corresponding bound $n^2 - 2n + 1$. □

As for the language $\mathcal{L}_n(P)$, one can also obtain from the completeness proof some complexity bounds for $\mathcal{L}_n(P, \mathbf{m})$.

Theorem 11 *Satisfiability of a formula in $\mathcal{L}_n(P, T)$ can be decided in exponential space. If the formula is quantifier free and the number of square root symbols is bounded by some fixed number then satisfiability can be decided in polynomial space.*

Proof: We note that similar reasoning to that above shows that $\phi'(\mathbf{x}, \mathbf{y}) \wedge \tilde{\Phi}$ is satisfiable iff ϕ is satisfiable(see Lemma 12). This formula may be constructed in time polynomial in $|\phi|$. Thus, the satisfiability problem reduces to that for $\mathbb{RC}$. The theorem then is a simple consequence of the Theorem 5. □

4.5 Alternative Formulations

There are two equivalent descriptions or pictures of dynamic evolution of a quantum system [Per95]. The Schroedinger picture is that of the state evolving in time and the observables or equivalently the bases representing observables fixed. The second view is called the Heisenberg picture in which the state is fixed and the bases of observables evolving in time. The evolution operator in both cases are certain unitary operators. Of course, both views are equivalent and it is easy to change from one to the other. The logical framework developed so far corresponds to the Heisenberg picture: a formula is evaluated at a state in different bases which are connected by unitary transformations. Such a choice of basis is motivated by the loose empiricist view that the properties of a quantum system are inferred from the data obtained by measurement in different bases. The quantum computing community however, usually work with the Scroedinger picture.

There is also a third intermediate picture called the interaction picture. The choice of the picture is dictated by the problem under consideration. For example, for quantum state tomography the language corresponding to Heisenberg picture - that is $\mathcal{L}_n(P, \mathbf{m})$ developed in the previous sections- is more natural. To describe the behaviour of quantum circuits the Schroedinger picture is preferred. We like to think that the quantum gates transform the state of the qubits under observation. In this section I develop alternative formulations of the language which correspond to the other pictures. These will prove useful as the description language of quantum computing.

First note that although we have relation of equality among complex (and real) terms there is no such relation among basis terms. However, by exploiting the (unitary) transformation relations among bases we can add a relation $\sim$ which behaves like equality. Thus, for basis variables $\mathtt{b}$ and $\mathtt{c}$ in

$\mathcal{L}_n(P, \mathbf{m})$ define:

$$\mathsf{b} \sim \mathsf{c} \Leftrightarrow \bigwedge_{i,j \in \{0,\ldots,n-1\}} m_{ij}(\mathsf{b}, \mathsf{c}) = \delta_{ij}. \tag{4.19}$$

Here, I have introduced the useful Kronecker symbol δ_{ij} which is equal to 0 if $i \neq j$ and 1 if $i = j$. For example if $n = 2$ the above expression is a shorthand for $m_{00}(\mathsf{b}, \mathsf{c}) = 1 \wedge m_{01}(\mathsf{b}, \mathsf{c}) = 0 \wedge m_{10}(\mathsf{b}, \mathsf{c}) = 0 \wedge m_{11}(\mathsf{b}, \mathsf{c}) = 1$. This is of course the unit matrix. I also use the following syntactic notation. Let $\Phi[\mathsf{b}]$ be a formula with probability and other terms over b-formulas. It may contain basis variables other than b-formulas for example in the unitary or transition probability terms. Let $\Phi[\mathsf{c}/\mathsf{b}]$ represent the formula obtained from $\Phi[\mathsf{b}]$ by replacing *every occurrence* of b-formula by the corresponding c-formula in $\Phi[\mathsf{b}]$. For example, if $\Phi[\mathsf{b}]$ is $P(\mathsf{b}_0) = x_0 \wedge P(\mathsf{b}_1) = x_1 \wedge (\wedge_{ij} m_{ij}(\mathsf{b}, \mathsf{d}) = x_{ij})$ then $\Phi[\mathsf{c}/\mathsf{b}]$ is $P(\mathsf{c}_0) = x_0 \wedge P(\mathsf{c}_1) = x_1 \wedge (\wedge_{ij} m_{ij}(\mathsf{c}, \mathsf{d}) = x_{ij})$. Now fix the dimension n and the corresponding Hilbert space $\mathbb{C}^n$ with the standard inner product over which the formulas below are interpreted.

Proposition 2 *Let $\pi, \psi \models \mathsf{b} \sim \mathsf{c}$ then for any formula $\Phi[\mathsf{b}]$ we have,*

$$\pi, \psi \models \Phi[\mathsf{b}] \text{ iff } \pi, \psi \models \Phi[\mathsf{c}/\mathsf{b}].$$

Proof: The proposition is obvious since in any interpretation the basis $\pi(\mathsf{b})$ is identical to $\pi(\mathsf{c})$ as the transformation matrix is the unit matrix of order n. 2 □

Define now an extension of the language $\mathcal{L}_n(P, \mathbf{m})$. First, introduce the following notation: U, V, W, U' etc. (possibly with subscripts) be symbols representing unitary matrices of dimension n. Call these unitary symbols. Sometimes, the dimension is explicit as $U^{(n)}$. Thus associated with each unitary symbol U there are n^2 variables(of $\mathbb{RC}$) written U_{ij}. We assume that we have a well-defined set of unitary symbols. We also require that if U is a unitary symbol and b is a basis then $U\mathsf{b}$ is also a basis. Let

$\mathcal{L}_n(P, \mathbf{m}, \mathbf{U})$ be the language obtained by this extension of $\mathcal{L}_n(P, \mathbf{m})$. We use collective quantification over these variables-effectively quantifying over unitary matrices. Thus, $\forall U \Phi$ stands for quantification over all $\mathbb{RC}$ variables U_{ij}. Now fix the notation for the "constant" gates X, Y, Z, H defined in the last section. Indeed, the definition of these gates involve only constants from $\mathbb{RC}$. For every basis symbol $\mathfrak{b}$ and unitary symbol U, $U\mathfrak{b}$ will denote a basis with basis components denoted by $(U\mathfrak{b})_i$ or simply $U\mathfrak{b}_i$ and add the axiom

Un $\quad \bigwedge_{ij} m_{ij}(\mathfrak{b}, U\mathfrak{b}) = U_{ij}$

to $\mathbf{Ax}_n(P, \mathbf{m})$. Let $\mathbf{Ax}_n(P, \mathbf{U}, \mathbf{m})$ denote the modified theory. It is easy to see that $\mathbf{Ax}_n(P, \mathbf{m}, \mathbf{U})$ is a conservative extension $\mathbf{Ax}_n(P, \mathbf{m})$. From the axiom **Un** and unitary matrix axiom **M3** it follows that the U_{ij} constitute a unitary matrix. Now fix a basis symbol, say, $\mathfrak{b}$. Let $\mathcal{L}_{n,\mathfrak{b}}(P, \mathbf{m}, \mathbf{U})$ be the subset of formulas of $\mathbf{TAx}_n(P, \mathbf{U}, \mathbf{m})$ obtained by restricting to the basis symbol $\mathfrak{b}$. This convention is reminiscent of the ubiquitous "computational basis" of quantum computing. Now the state is supposed to change. As discussed above this is the "Schroedinger picture". Since there is only one basis symbol "transition probability" and the transformation matrices m_{ij} do not have much use. We have only probability or *P-terms*. As before the language $\mathcal{L}_{n,\mathfrak{b}}(P, \mathbf{m}, \mathbf{U})$ language is an extension of $\mathbb{RC}$; so all the symbols of $\mathbb{RC}$ are available. The probability terms are of the form $P(\phi)$ where now we have now the restriction that ϕ contains only the basis symbol $\mathfrak{b}$. Given a polynomial relation $p(x_1, \ldots, x_k) \geq 0$ a probability atom is obtained by replacing some of the variables in the relation by probability terms $P(\phi), P(\phi'), P(\phi'')$ etc. with the requirement that the only basis appearing in the probability formulas is $\mathfrak{b}$. The interpretation of the formulas is as before. A general probability formula is built from probability atoms and $\mathbb{RC}$-formulas by using logical connectives and quantifiers over variables from $\mathbb{RC}$. We note that the full language $\mathcal{L}_n(P, \mathbf{m}, \mathbf{U})$ can be considered as an extension of $\mathcal{L}_{n,\mathfrak{b}}(P, \mathbf{m}, \mathbf{U})$. Note also the following:

In the axiomatization of $\mathcal{L}_n(P, \mathbf{m})$ we stipulate that all formulas of

$\mathcal{L}_n(P, \mathbf{m}, \mathbf{U})$ where some basis symbol $\mathfrak{b}$ is replaced by symbols of the $U\mathfrak{b}$ is also an instance of the corresponding axiom. Add the formula **Un** as an axiom and let $\mathbf{Ax}_n(P, \mathbf{m}, U)$ be the corresponding theory.

Lemma 13 $\mathbf{Ax}_n(P, \mathbf{m}, U)$ *is a* conservative *extension of* $\mathbf{Ax}_n(P, \mathbf{m})$. *That is, any formula of* $\mathcal{L}_n(P, \mathbf{m})$ *that is a theorem in* $\mathbf{Ax}_n(P, \mathbf{m}, U)$ *is also a theorem in* $\mathbf{Ax}_n(P, \mathbf{m})$.

Proof: $\mathbf{Ax}_n(P, \mathbf{m}, U)$ has only one extra axiom namely **Un**. Call the theory with the same language as $\mathbf{Ax}_n(P, \mathbf{m}, U)$ but without the axiom **Un** $\mathbf{Ax}'_n(P, \mathbf{m}, U)$. Then, a closed formula $\mathbf{F}$ is a theorem of $\mathbf{Ax}_n(P, \mathbf{m}, U)$ if and only if $\mathbf{V} \Rightarrow \mathbf{F}$ is a theorem of $\mathbf{Ax}'_n(P, \mathbf{m}, U)$ where $\mathbf{V}$ is a conjunction of instances of **Un**. Now for each appearance of a symbols of the form $U\mathfrak{b}$ in $\mathbf{V}$ introduce a new basis symbol $\mathfrak{b}^U$ in $\mathcal{L}_n(P, \mathbf{m})$. Here the superscript U is just an index. Let $\mathbf{V}' = \wedge_{ij}(m_{ij}(\mathfrak{b}, \mathfrak{b}^U) = U_{ij}$. Then $\mathbf{V}' \Rightarrow \mathbf{F}'$, where $\mathbf{F}'$ is obtained by substituting $\mathfrak{b}^U$ for $U\mathfrak{b}$ is provable in $\mathbf{Ax}'_n(P, \mathbf{m}, U)$, as it is a substitution instance of $\mathbf{V} \Rightarrow \mathbf{F}$. Suppose $\mathbf{F}$ is a formula in $\mathcal{L}_n(P, \mathbf{m})$. Then $\mathbf{F}$ and $\mathbf{F}'$ are identical. Since $\mathbf{F}$ is closed by $\exists$-introduction rule [Sho67] $\exists U_{ij} V' \Rightarrow \mathbf{f}$ is theorem. Since $m_{ij}(\mathfrak{b}, \mathfrak{c}) = m_{ij}(\mathfrak{b}, \mathfrak{c})$ for any $\mathfrak{b}$ and $\mathfrak{c}$, $\mathbf{F}$ is a theorem of $\mathbf{Ax}'_n(P, \mathbf{m}, U)$. A similar argument shows that the instances of axioms of $\mathbf{Ax}'_n(P, \mathbf{m}, U)$ with basis symbols of type $U\mathfrak{b}$ may be eliminated and we are left with a proof in $\mathbf{Ax}_n(P, \mathbf{m})$. □

We further extend $\mathcal{L}_{n,\mathfrak{b}}(P, \mathbf{m}, \mathbf{U})$ as follows. If U is a unitary symbol and Φ is formula of $\mathcal{L}_n(P, \mathbf{U})$ then $[U]\Phi$ is also a formula. The interpretation $[U]\Phi$ is as follows. I suppress the fixed Hilbert space H. First, we stipulate that in any interpretation π, $\pi(U)$ is to be interpreted as a unitary matrix. As stated above, corresponding to any unitary symbol U, we have n^2 associated symbols U_{ij} which are interpreted as complex numbers, just like m_{ij}.

$$\pi, \psi \models [U]\Phi \text{ iff } \pi, \pi(U)^{-1}\psi \models \Phi$$

Here the notation means the following. First Let $\pi(\mathsf{b}) = \{\alpha_0, \ldots, \alpha_{n-1}$ be the interpretation of b and let $\psi = \sum c_i \alpha_i$. Then

$$U^{-1}\psi \stackrel{\text{def}}{=} \sum_{ij} \overline{U(ji)} c_j \alpha_i.$$

Let $\mathcal{L}_n(P, \mathbf{U})$ denote the fragment of $\mathcal{L}_n(P, \mathbf{U}, \mathbf{m})$ consisting of all formulas *not* containing any terms involving m_{ij} and only one basis symbol b. The restriction to one basis symbol is important because U acts on the coefficients of the state vector expressed in the basis corresponding to b. We may also wish to express transformation of bases. This is achieved by the symbols $U\mathsf{b}$. Intuitively, $U\mathsf{b}$ is the basis obtained by applying the unitary transformation U to all vectors in the basis b. They are all semantically equivalent in the following sense.

Lemma 14 *Let $\mathcal{L}_n(P, \mathbf{m}, \mathbf{U})$ be the extension of $\mathcal{L}_n(P, \mathbf{m})$ by the addition of the unitary symbols and interpretations as above. For all interpretations for which* **Un** *is valid,*

$$[U]\Phi[\mathsf{b}] \Leftrightarrow \Phi[U\mathsf{b}] \Leftrightarrow \bigwedge_{ij} m_{ij}(\mathsf{b}, \mathsf{c}) = U_{ij} \Rightarrow \Phi[\mathsf{b}/\mathsf{c}]$$

is valid.

Proof: Let $\pi(\mathsf{b}) = \{\alpha_0, \ldots, \alpha_{n-1}\}$ be the interpretation of the basis symbol b. Then, since $m_{ij}(\mathsf{b}, \mathsf{c}) = U(ij)$, $\pi(\mathsf{c}) = \{\beta_0, \ldots, \beta_{n-1}\}$ with $\beta_i = U\psi_i = \sum_k U(ki)\psi_k$. Now it is clear that the lemma has to be proved only for probability atoms. Moreover, since $\wedge_{ij} m_{ij}(\mathsf{b}, \mathsf{c}) = U_{ij} \Rightarrow U\mathsf{b} \sim \mathsf{c}$ is valid and the second assertion follows from interpretation of $U\mathsf{b}$ and Proposition 2. I prove the equivalence of the first two. I prove the validity of the formula when Φ is a probability atom in $\mathcal{L}_n(P, \mathbf{m})$ first. Φ is obtained by substituting probability terms for the variables in some polynomial. From the lemma 6 it suffices to prove that the proposition true when the polynomial substitution is of the form $p(y_0/P(\mathsf{b}_0), \ldots, y_{n-1}/P(\mathsf{b}_{n-1}))$. Now, from 4.3.2

$$\begin{aligned}[\![P([U]\mathsf{b}_i)]\!]_{\pi,\psi} &= [\![P(\mathsf{b}_i)]\!]_{\pi,U^{-1}\psi} = |\langle\alpha_i|U^{-1}\psi\rangle|^2 = |\langle\alpha_i|U^{\dagger}\psi\rangle|^2 \\ &= |\langle\alpha_i|\sum_{jk}\overline{U(jk)c_j}\alpha_k\rangle|^2 = |\langle\sum jU(ji)\alpha_j|\sum_k c_k\alpha_k\rangle|^2 \\ &= [\![P((U\mathsf{b})_i)]\!]_{\pi,\psi}\end{aligned}$$

Intuitively, the formulas above simply express the fact that the valuation of $P((U\mathsf{b})_i)$ at ψ is same as the valuation of $P(\mathsf{b}_i)$ at $U^{-1}\psi$. Now the proposition is obvious since we may switch from one valuation to the other. □

As we will see in the next chapter the new language $\mathcal{L}_n(P,U)$ will prove very useful for quantum circuits. The reason for proving the equivalences in the preceding lemmas is that we do not have to prove to properties like completeness and decidability separately for the different languages.

Chapter 5

Logics for QCI

In this chapter I extend the logics developed in the previous chapter to incorporate notions like tensor product and measurement. The extended logic is expressive enough to deal with quantum circuits as a special case. Yet, it preserves the desirable properties like completeness and decidability. There is also some generalization in the interpretation of the formulas. The language $\mathcal{L}_n(P, \mathbf{m})$ of the previous chapter was restricted to a fixed dimension n. This is adequate for expressing properties of single indecomposable quantum systems. But if the system is composed of parts which are quantum systems of smaller dimensions then the states of the whole system are described by the tensor product of states of the parts. A simple classical analogue is a two bit system. The states of this system are given by the *cartesian* product of the states of individual systems. Consequently, any local operation on one bit does not affect the state of the other bit. However the state space of a two qubit quantum system has the more complicated structure of a tensor or direct product space. This is briefly discussed in the chapter on quantum theory.

The other important physical concept that is dealt with is that of measurement. In classical theory the notion of measurement is implicit in the final 'reading' of the output. It is also implicit that the act of observing or reading the output does not affect the state of the latter or influence

future behaviour. This ideal assumption is of course perfectly justified since the disturbance introduced by the reading process is so small that it does not affect the state. For example, the high/low states of a switch may be distinguished by a few millivolts and a reading apparatus like an accurate voltmeter introduces an error of only few microvolts. But it is in the very nature of objects in the quantum scale that the act of measurement usually introduces an irreversible change which is manifested as a random change in the state. The theory of measurement is an important and deep aspect of quantum theory. However, for the present purposes the simple pragmatic approach outlined in Chapter 2 will suffice. An outline of the chapter follows.

Section 5.1.1 deals with the syntax and semantics of the language. New symbols for tensor product and measurement are introduced. Only these new constructs are described in detail. However, for the semantics, the notion of state is generalised to mixed states. In the last chapter the formulas were interpreted in "pure" states. The motivation for the generalization to mixed states will be explained in this section. I also present the equivalent 'variants' of the language corresponding to the Schroedinger and interaction pictures. (see 4.5).

In Section 5.1.3 some examples from quantum computation and information- the main motivation for developing the logic- are presented. In particular, application to formal reasoning about quantum circuits are dealt with extensively. An algorithm for writing any combinational quantum circuit as a formula in the the language is given. As another application a nontrivial complexity bound of quantum complexity hierarchy [BV97] is proved using different techniques. Further applications to the quantum complexity theory are discussed. Applications to important algorithms are discussed in the next chapter.

In the final section (5.3) I give a sound and complete axiomatization. Some complexity theorems are proved. As a corollary one obtains yet another proof of the complexity upper bound mentioned above. In the conclu-

sion, I discuss some possible extensions of the language and further developments.

5.1 Syntax and Semantics of $\mathcal{L}_n(P, \mathbf{m}, t, M)$

5.1.1 Syntax

In the previous chapter a language $\mathcal{L}_n(P, \mathbf{m})$ for quantum probabilities in a fixed dimension n was developed. However, a composite system is described as the (tensor) product of smaller systems. That is, *different* dimensions may have to be considered. To capture this I extend the language $\mathcal{L}(P, \mathbf{m})$ by first introducing a new syntactic object for tensor product. Notice that the dimension is not explicitly mentioned.

As before symbols $\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}$ etc. will stand for basis variables. These are "irreducible" basis symbols in contrast to another "composite" type introduced below. Sometimes I write $\mathtt{b}^{(n)}$ to indicate that the dimension is n. We will also identify certain fragments of $\mathcal{L}(P, \mathbf{m})$ with $\mathcal{L}_n(P, \mathbf{m})$ in a precise sense described below. As in the previous chapter, associate with each $\mathtt{b}$ in $\mathcal{L}_n(P, \mathbf{m})$ a set of symbols $\{\mathtt{b}_i\}_{i=0}^{n-1}$. Thus a subscript always denotes "components" of $\mathtt{b}$. Semantically, basis variables represent orthonormal bases in a Hilbert spaces and the components represent projections on the corresponding vectors in a basis. But unlike the previous chapter the Hilbert space has more structure: the tensor structure. The basis variable $\mathtt{b}^{(k)}$ will be interpreted in a Hilbert space of dimension k, usually $\mathbf{C}^k$ with the standard scalar product. Recall that the probability operator P is defined over $\mathtt{b}$-formulas constructed out of basis components of a basis variable $\mathtt{b}$. Thus, if $\mathtt{b}$ is a basis variable then $\mathtt{b}_i$ are $\mathtt{b}$-formulas over $\mathtt{b}$. If Φ and Ψ are $\mathtt{b}$-formulas over one basis symbol then so are $\neg\Phi$ and $\Phi \wedge \Psi$. As I will be dealing with several basis variables in possibly different dimensions it is convenient to call the collection of basis formulas as $\mathcal{B}$-formulas. There is a further extension of $\mathcal{B}$-formulas by a tensor operation to be explained soon.

By definition, every $\mathcal{B}$-formula is constructed out of a *single* basis vari-

able. Thus $\mathtt{b}_i \wedge \mathtt{c}_j$ is *not* a $\mathcal{B}$-formula. Sometimes we write $\Phi_{\mathtt{b}}$ for a $\mathcal{B}$-formula in basis variable $\mathtt{b}$. Semantically, $\mathcal{B}$-formulas represent projections on subspaces generated by a basis. Physically, they correspond to the events/propositions generated by choice of atomic events -the possible outcomes of a maximal test i. e. . a measurement whose number of outcomes is maximum possible(equal to the dimension of the quantum system). Different choices of basis will yield different event structures. Each one gives rise to a boolean lattice on which the probability operator is defined. I note that these ideas roughly correspond to the notion of manuals [FR78] which is more general but complicated.

I briefly review the syntax of the language $\mathcal{L}_n(P, \mathbf{m})$ for completeness. For $\mathcal{B}$-formulas Φ, a probability atom is an expression of the form $\sum_i k_i P(\Phi_i) \geq c$ where k_i and c are integers. Typical examples are formulas of the type $P(\Phi_i) = 1$ and $kP(\Phi_i) \geq 1$ where k is a positive integer. The formulas of the language $\mathcal{L}_n(P)$ are all the boolean combinations of linear probability atoms.

For the language $\mathcal{L}_n(P, \mathbf{m})$, add *transition matrix terms*, which are expressions of the form $m(\mathtt{b}, \mathtt{c})$ for basis variables $\mathtt{b}$ and $\mathtt{c}$. Associated to $m(\mathtt{b}, \mathtt{c})$ are a set of n^2 terms $m_{ij}(\mathtt{b}, \mathtt{c})$ of complex sort. Recall that $\mathtt{b}$ and $\mathtt{c}$ represent two orthonormal bases in a Hilbert space. Then, $m_{ij}(\mathtt{b}, \mathtt{c})$ represents the entries of the unitary matrix taking transforming the basis represented by $\mathtt{b}$ to that of $\mathtt{c}$. $\mathcal{L}_n(P, \mathbf{m})$ also incorporates the first order language $\mathcal{L}_{\mathbb{RC}}$ discussed in 4.2. For a pair of 'atomic' basis terms $\mathtt{b}_i$ and $\mathtt{c}_j$ define the transition probability term $T(\mathtt{b}_i, \mathtt{c}_j) \stackrel{\text{def}}{=} |m_{ij}(\mathtt{b}, \mathtt{c})|^2$. Given a formula in $\mathcal{L}_{\mathbb{RC}}$ i. e. polynomial equation or in-equation (possibly with quantifiers and predicates denoting real nature of some variables) $p(y_1, y_2, \ldots, y_k) \leq 0$ an $\mathcal{L}_n(P, \mathbf{m})$ atom is obtained by replacing some variables uniformly by expressions $P(\Phi_i)$ and $m_{ij}(\mathtt{b}, \mathtt{c})$. Thus $\sqrt{3}P(\mathtt{b}_1 \wedge \mathtt{b}_2)^2 + T(\mathtt{b}_1, \mathtt{c}_3)^3 - 1/2 \leq 0$ is an atom.

Recall that the tensor product of an m-dimensional and n-dimensional

vector space is an mn-dimensional vector space. That is, the tensor product connects spaces of different dimension. The basis symbols and formulas in $\mathcal{L}_n(P, \mathbf{m})$ pertain to a fixed dimension n. Hence, to introduce the tensor product we need to consider the collection of basis symbols which will be interpreted in spaces of different dimensions. Thus, for each n call the collection of basis symbols and formulas in $\mathcal{L}_n(P, \mathbf{m})$, the $\mathcal{B}$-terms of sort n. That is, the set of $\mathcal{B}$-expressions is the union of $\mathcal{B}$-formulas and basis symbols b, c etc. in some dimension.

For each positive integer N and every nontrivial factorization $N = mn($ m and $n \neq 1)$ we extend the set of basis terms as follows. Suppose the extension of $\mathcal{B}^m$ and $\mathcal{B}^n$ is already defined and denoted by the same symbols. Then for each pair pair of basis symbols $(X^{(m)}, X^{(n)}) \in \mathcal{B}^m \times \mathcal{B}^n$ we introduce symbols $t(X^{(m)}, X^{(n)})$. This process must stop when we reach a prime factor. Let T_{mn} be the set of all such symbols for all possible pairs $(X^{(m)}, X^{(n)})$ then we extend the basis terms $\mathcal{B}_N$ by adding symbols of the form $t(X^{(m)}, X^{(n)})$. We continue to call the extended set $\mathcal{B}_N$. However, we distinguish the original set of basis symbols by calling them *irreducible*. The terminology is self explanatory since the new class of bases is to interpreted as tensor product(or simply) product bases. Identify $t(\mathsf{b}_i^{(m)}, \mathsf{c}_j^{(n)})$, $0 \leq i \leq m-1$, $0 \leq j \leq n-1$, as a basis component of $t(\mathsf{b}^{(m)}, \mathsf{c}^{(n)})$ such that $t(\mathsf{b}_i^{(m)}, \mathsf{c}_j^{(n)}) = t(\mathsf{b}^{(m)}, \mathsf{b}^{(n)})_{in+j}$. Let $J_k = \{0, 1, \ldots, k-1\}$. For every pair of integers m, n we define the map

$$g_{mn} : J_m \times J_n \to J_{mn} \text{ given by } g_{mn}(i, j) = in + j.$$

It is invertible with inverse

$$g_{mn}^{-1}(k) = (r_1(k), r_0(k)) \stackrel{\text{def}}{=} ([k/n], k \bmod n) \tag{5.1}$$

where [x] is the greatest integer $\leq x$, $t(\mathsf{b}^{(m)}, \mathsf{c}^{(n)})_k = t(\mathsf{b}_{r_1(k)}, \mathsf{c}_{r_0(k)})$. Call this identification tensorial decomposition of $t(\mathsf{b}^{(m)}, \mathsf{c}^{(n)})$ The map g_{mn} introduces an ordering in the set J_{mn}. Recall that associated to each basis

symbol $\mathtt{b}$ in n-dimension, are n symbols $\mathtt{b}_0 \ldots \mathtt{b}_{n-1}$ denoting basis components. For the (compound) basis term $t(\mathtt{b}^{(m)}, \mathtt{b}^{(n)})$ in dimension mn the corresponding basis components are required to be ordered as an array of pairs (i,j), $i = 1 \ldots m$, $j = 1 \ldots n$ by the tensorial decomposition. This ordering is simply the 'dictionary' ordering of the pairs. Note that if $mn = m'n'$ then the map $g_{m'n'}$ introduces a different ordering. However, the map $g^{-1}_{m'n'} g_{mn}$ is a bijection between the two orderings. We extend the operator t to all basis formulas by requiring that

$$t(\Phi_1 \vee \Phi_2, \Psi) = t(\Phi_1, \Psi) \vee t(\Phi_2, \Psi) \text{ and}$$

$$t(\neg\Phi_1, \Phi_2) = \neg t(\Phi_1, \Phi_2) \wedge t(\top, \Phi_2)$$

We define similar formulas for the second argument. The negation of a basis formula is the complimentary formula. That is, if Φ is a disjunction of basis atoms then $\neg\Phi$ is a disjunction of atoms which are indexed by the set complimentary to the set of indices in Φ (see the deductions in Section 4.4.1). But the set of indices depends on the dimension. The set of indices for the tensor product is $\{0, \ldots, mn-1\}$ and these are ordered by the map g_{mn} above. Hence, the conjunction with the term $t(\top, \Phi_2)$ picks up the components constituting Φ_2. Note that, these formulas capture the linearity of tensor product with respect to each factor. We can now recursively define the syntax for the product of more than two basis symbols. Thus, $t(t(\mathtt{b}^m, \mathtt{b}^n), \mathtt{b}^p)$ is basis symbol in dimension mnp. We have the corresponding ordering defined by the map $g_{mnp} \equiv g_{mn,p} \cdot g_{mn}$ in the set J_{mnp} induced by the triplet (m,n,p). The map g_{mnp} is unambiguous due to the "associative' property

$$g_{mn,p} \cdot (g_{mn}, \mathtt{id}_p) = g_{m,np} \cdot (\mathtt{id}_m, g_{np}).$$

Here $\mathtt{id}_p$ is the identity map on J_p. It is straightforward to extend these definitions to arbitrary products. Letting, $N = n_1 \cdots n_k = n'_1 \cdots n'_k$ we see that the map $g^{-1}_{n'_1 \cdots n'_k} g_{n_1 \cdots n_k}$ induces a bijection between the two orderings.

At this point it is instructive to consider the possible interpretations of the logic with tensor product notation. If one aims to interpret the theory in its full generality one needs to consider a Hilbert space of dimension n for each natural number and that there is a corresponding set of $\mathcal{B}$-expressions for each n. These, sets are required to be mutually disjoint. The only formulas in $\mathcal{L}(P, \mathbf{m})$ whose valuation is state dependent are the probability formulas. Since we want to consider only formulas pertaining to copies of an arbitrary but fixed *single* system the dimension of the state space remains fixed. The quantum system may be be composed of several subsystems e. g. multiple qubits. But none of these constituent parts are destroyed nor new ones added during the period of interaction and observation. Note however, that we do not rule out operations performed only on a part of the system. As explained in Chapter 2 this may be considered as an operation on the whole system. Hence, although we allow basis symbols of all sorts (=dimensions), the only formulas which will have a proper physical interpretation are the *homogeneous formulas*- defined as formulas in which the probability formulas pertain to an arbitrary but fixed dimension. For example, the formula $P(\mathtt{b}_1) > 1/2 \wedge P(t(\mathtt{b}_1, \mathtt{b}_2)) > 1/3$ is *not* a homogeneous formula since it refers to spaces of different dimension- one for $\mathtt{b}$ and the other for $t(\mathtt{b}, \mathtt{b})$. This restriction on the formulas may be somewhat dissatisfying. Although they may be satisfactorily interpreted in an appropriate Hilbert space the latter does not constitute the state space of quantum systems under consideration. A deeper reason for keeping the inhomogeneous formulas is the following. If one were to deal with physical systems of **quantum field theory** (more precisely a second quantized theory) which allow particle creation and annihilation, then we may require the full generality of the language. But for our purposes it is sufficient to restrict to homogeneous probability formulas. Thus, a probability term refers to some fixed dimension determined by the basis terms over which it is defined. We are therefore in a situation similar to the previous chapter but the difference is that the basis terms appearing

in the formula may be from a composite or product basis. Since the transformation matrix terms are interpreted independently of the state there is no restriction on them. Call the new language $\mathcal{L}(P, \mathbf{m}, t, M, S)$. Note the absence of dimension.

For quantum computation and information (QCI) we do not need even this restricted structure because then one deals with qubits or the more esoteric qunits (n-dimensional quantum systems). Therefore, we only require the tensor powers of a fixed Hilbert space. For example, in case of qubits the space is $\mathbf{C}^2$ with the standard inner product. For generality, fix some positive number $d \geq 2$ as the base dimension. The basis symbols now will refer to spaces of dimension s^k. For example, if $d = 2$, and $\mathtt{b}, \mathtt{c}$ are basis symbols in 2 dimension then $t(t(\mathtt{b}, \mathtt{c}), \mathtt{b})$ will denote a (product) basis in $2^3 = 8$ dimensions. Call the restricted language $\mathcal{L}^d(P, \mathbf{m}, t) = \bigcup_k \mathcal{L}_{d^k}(P, \mathbf{m}, t)$ for a fixed positive number d(d=2 mostly). A $\mathcal{B}$-expression like $t(\mathtt{b}^0, \mathtt{b}^1, \ldots, \mathtt{b}^k)$ stands for $t(\mathtt{b}^0, t(\mathtt{b}^1, \ldots t(\mathtt{b}^{k-1}, \mathtt{b}^k)))$; that is, I assume implicit right associativity. Note an interesting consequence of recursive application of tensorial decomposition: $t(\mathtt{b}^0, \mathtt{b}^1, \ldots, \mathtt{b}^n)_k = t(b_{s_n(k)}, \ldots, b_{s_0(k)})$, where $s_0(k) = r_0(k), s_1(k) = r_0(r_1(k)), \ldots s_n(k) = r_0(r_1^n(k))$ is the representation of k in base d(with each nonnegative integer $\leq d^n - 1$ represented by n "'dits"', padded by 0's if necessary).

The final syntactic constructs are the operators for measurement. In quantum theory it is convenient to classify measurements into two groups. The first group consists of the detection type measurement. This kind of measurement is referred to when we have the system in some state which may be unknown and we empirically calculate the probabilities based on our measurements in some bases. If the state is known beforehand then one may compute the *a priori* probabilities. But to have certain knowledge of the state one has to *prepare* the system in that state. This preparation process constitutes the second kind of measurement. Usually, we do this by starting with an unknown state and then 'filtering' the system through a

selective device. Recall that when we make a maximal test a large number of systems is passed through a a measuring device which separates the systems into orthonormal states in a basis. For example, in a 2 dimensional spin-1/2 system one may choose a basis corresponding to the two spin states in, say, z direction. If a beam of electrons is passed through a device like the Stern-Gerlach apparatus then it splits in two beams corresponding to the two states [Per95]. The selection or preparation is done by choosing a particular basis state in the measuring device and ignoring the rest. This classification of separating measurements into two categories is somewhat artificial as the same physical measurement could be termed a detection or selection depending on the context. In the case of quantum protocols too it becomes important to formally distinguish the two measurements. For example, in the two party teleportation protocols one agent, say Alice, performs the measurement and notes the outcome while the other agent Bob only knows that a particular measurement has been done. Therefore, Alice and Bob's description of the post-measurement situation will differ. I shall come back to the teleportation example later.

It is clear from the preceding discussion that we should distinguish the two 'types' of measurement. First, define the syntactic operator corresponding to (detection type) measurement in a more general setting. Let X be a basis variable in dimension m. It may be an atomic or composite type. Let $\{s_1, \ldots, s_r\}$ be a partition of the set $\{1, \ldots, m\}$ such that $s_k = \{i_1, i_2, \ldots, i_k\}$. Consider the set of basis formulas $\mathcal{F} = \{\phi_{s_1} \ldots, \phi_{s_r}\}$ with $\phi_{s_k} = X_{i_1} \vee \cdots \vee X_{i_k}$. Call $\mathcal{F}$ a complete set of basis formulas. Intuitively, the ϕ_i constitute a set of mutually exclusive events that is complete in the sense that one of them must occur in a quantum measurement in the X basis. For every such complete set $\mathcal{F}$ and for any formula Ψ in $\mathcal{L}_m(P, \mathbf{m}, t, M, S)$, $M_{\mathcal{F}}(\Psi)$ is a formula. The intuition is that $M_{\mathcal{F}}$ corresponds to a quantum measurement which yields one of the of the *subspaces* represented by the $\phi_i's$. An important special case is when

$\mathcal{F} = \{X_0, \ldots, X_{m-1}\}$. The corresponding measurement is called a *maximal test* because its outcomes provide the maximum possible information about the system. In this case we simply use the notation $M_{\mathtt{X}}$ for the measurement operator corresponding to the basis $\mathtt{X}$. The operator for the selection type measurement will be denoted by S_ϕ for a $\mathtt{b}$-formula ϕ. Like M it is applied to any formula Ψ in $\mathcal{L}(P, \mathbf{m}, t)$. The intuition is, $S_\phi(\Psi)$ holds in a state represented by ϕ. The exact meaning will be clarified when I treat the semantics of the formulas. It is important to note that the measurement operators are applied to *homogeneous* formulas and the definition of homogeneity is extended to corresponding formulas with measurement operators.

Let $\mathcal{L}(P, \mathbf{m}, t, M, S)$ be the boolean combinations of probability formulas (possibly with t-operator), transition probability formulas, and measurement formulas. The special case of the language in which the dimension are restricted to tensor powers of a base dimension r I call it $\mathcal{L}^r(P, \mathbf{m}, t, M)$. We finally have the four most important attributes of quantum systems: probability (unconditional), unitary operation and transition probability, tensor product for composite systems, and measurement.

5.1.2 Summary of Syntax

Let us summarize the syntax of the language. Let $S_k = \{m_1, \ldots, m_k\}$ be distinct positive integers such that for each m_i we have atomic basis symbols of that "sort". Intuitively, m_i signifies the dimension of the Hilbert space. Sometimes we indicate the dimension explicitly by writing m_i as a superscript. Thus, $\mathtt{b}^{(2)}$ is a basis symbol in 2 dimensions. Let m be a fixed positive integer and $\{a_1, a_2, \ldots, a_r | a_i \in S_k\}$ be finite sequences such that their product is m. Then,

$$t(\mathtt{b}^{a_1}, \mathtt{b}^{a_2}, \ldots, \mathtt{b}^{a_r})$$

is a composite basis symbol in dimension m. Note that, for convenience of notation I do not introduce the grouping of the symbols. Thus in the

(temporary) generic notation $t(\mathtt{b}^{a_1}, \mathtt{b}^{a_2}, \ldots, \mathtt{b}^{a_r})$ is generic notation for arbitrarily grouped t-operators over the relevant basis symbols. For example, if $r = 3$ it may denote $t(t(\mathtt{b}^{a_1}, \mathtt{b}^{a_2}), \mathtt{b}^{a_3})$ or $t(\mathtt{b}^{a_1}, t(\mathtt{b}^{a_2}, \mathtt{b}^{a_3}))$. The "components" of such basis symbols is given by the ordering map g defined above. We assume that the atomic dimensions m_i are fixed at the outset. For every integer $n = m_1^{j_1} \cdots m_k^{j_k}$ let $\mathcal{B}_n$ denote the basis terms of sort (dimension) n. Let $\mathcal{B} = \bigcup \mathcal{B}_n$ denote the disjoint union. The probability operator P and the operator m_{ij} for transition matrix are applied to the terms in $\mathcal{B}_n$. Let $X_i^{(n)}, X_i'^{(n)}, X_i''^{(n)} \ldots \ 0 \leq i \leq n-1$ denote basis terms in $\mathcal{B}_n$. We remember that if $X^{(n)}$ is a compound term, i.e. a product of basis terms of lower dimension, and the components inherit the ordering induced by the latter. Let $\mathtt{D}, \mathtt{D}', \mathtt{D}''$ be basis formulas corresponding to the above bases. Let $q(x_1, \ldots, x_r)$ be a multivariate polynomial. Then an atomic formula is obtained by substituting $P(\mathtt{D}), P(\mathtt{D}'), P(\mathtt{D}'')$ uniformly for some of the variables in the formula $q(x_1, \ldots, x_r) \geq 0$. We may also substitute matrix terms $m_{ij}(X, X')$. A general formula of sort n is defined recursively. Thus, if Φ_1 and Φ_2 are formulas then, $\Phi_1 \wedge \Phi_2, \neg\Phi_1, M_X\Phi_1$, and $S_X\Phi_1$ are formulas. Let Fm_n denote the formulas of sort n thus obtained. Then the set of formulas of $\mathcal{L}(P, \mathbf{m}, t, M, S)$ is given by $\bigcup_n Fm_n$. Note that, we have ensured that all formulas are homogeneous, that is they belong to some dimension n. Note also that to simplify notation we do not make explicit mention of the "atomic" dimensions $m_1, m_2, \ldots$.

5.1.3 Semantics

In the previous chapter the language $\mathcal{L}_n(P, \mathbf{m})$ was interpreted in a Hilbert space- the state space of a quantum system. The main features of the semantics of last chapter are summarized below.

1. The basis symbols are interpreted as orthonormal bases in the Hilbert space.

2. The real and complex variables are interpreted as usual in the field of

complex numbers considered as an algebraic extension of the field of reals.

3. The matrix terms are interpreted as *unitary* transformation matrices between the bases.

4. The atomic probability formulas are interpreted as expressing the probability of the basis terms in a given state.

Only the probability formulas are state dependent. The notion of state used in the last chapter was that of pure state. This notion is an idealization that assumes that we have complete information about the nature of the system. However, in the case of composite system, possibly interacting with external environment, often one can only access part of the system. Sometimes we even lack the knowledge of what constitutes the system! Therefore, henceforth, the probability formulas will be evaluated in mixed states. Mixed states or density matrices are explained in the chapter on quantum theory. Here I only remark that mixed states reflect a 'classical uncertainty' about the state and this is captured by a classical probability distribution. We may consider classical probabilities over (pure) quantum states and avoid the density matrix formalism. The main reasons for adopting the density matrix semantics are as follows.

1. The pure states are a special type of density matrix. Thus, corresponding to a vector $|\psi\rangle$ we have the density matrix $|\psi\rangle\langle\psi|$. An arbitrary density matrix is a convex linear combination of pure state density matrices. That is, any density matrix may be written as $\sum_i x_i |\psi_i\rangle\langle\psi_i|$ such that $x_i \geq 0$ and $\sum x_i = 1$. We have a uniform interpretation of formulas.

2. The simple formula $P(\mathsf{b}_0) = 1$ is satisfiable only at a pure state. So we may characterize pure states by such a formula. In other words the set of pure states is *definable* [Sho67].

3. A pure state assumes the ideal situation of an isolated system. In some instances it is not reasonable to make this assumption. For instance, when the quantum system is coupled to its environment which is incompletely known. Then the only states that can be attributed to the system are mixed or density matrix states.

4. The semantics of the measurement operators given below is simple and elegant in terms of density operators. Some of the interesting consequences will be discussed.

As before a *structure* for $\mathcal{L}_n(P, \mathbf{m})$ is made up of the n-dimensional complex Hilbert space $H_n \equiv \mathbb{C}^n$ with the standard inner product. Let $L(H_n)$ be the space of linear operators on H_n. The vectors of H_n, written as $|\alpha\rangle$ are $n \times 1$ matrices with components z_i. The dual $\langle\alpha|$ is a row vector with entries $\overline{z_i}$. A basis will continue to mean an orthonormal basis. Two subsets of $L(H_n)$ are of special interest: $\mathcal{L}_n$, the space of hermitian operators and $\mathcal{U}_n$ the group of unitary operators. For a square matrix A of order n, A_{ij} will denote its ij^{th} entry and $\mathtt{Tr}(A) \equiv \sum_i A_{i,i}$. A *state* is an operator $\rho \in \mathcal{L}_n$ with $\mathtt{Tr}(\rho) = 1$ and $\sum_{ij} \overline{x_i}\rho(ij)x_j \geq 0$ for all real vectors $\mathbf{x} = (x_1, \ldots, x_n)^T$, where A^T is the transpose of the matrix A. It is positive semidefinite with trace 1. The interpretation of basis symbols and complex and real variables are identical to the last chapter and I repeat it briefly for completeness.

An *interpretation* of $\mathcal{L}_n(P, \mathbf{m})$ in a structure $H_n \equiv \mathbf{C}^n$ is function π, such that for each basis variable $\mathtt{b}$, $\pi(\mathtt{b})$ is an orthonormal basis $\psi_0, \ldots, \psi_{n-1}$ of H; (we write $\pi(\mathtt{b})_i$ for ψ_i and occasionally suppress the $|\rangle$ notation). If $M = (m_{ij})$ is an $n \times n$ unitary matrix and $B = \psi_1, \ldots, \psi_n$ is a sequence of vectors of H, we write MB for the sequence of vectors $\psi'_1, \ldots, \psi'_n$, where $\psi'_i = \Sigma_{i=1}^n m_{ik}\psi_i$. If B is an orthonormal basis of H then so is MB. To give semantics to formulas of $\mathcal{L}_n(P, \mathbf{m})$, we define a relation of satisfaction of a formula ϕ at a state ρ in a structure H, with respect to an interpretation π, denoted by $H, \pi, \rho \models \phi$. The definitions are straightforward and the reader may refer to the last chapter for details. The only point of departure is that

the formulas are interpreted in general mixed states. Since, so far only the probability formulas are state dependent only those are affected. A formula ϕ of $\mathcal{L}_n(P,T)$ is *satisfiable* (in the n-dimensional Hilbert space H) if there exists an interpretation π and a state ρ such that $H,\pi,\rho \models \phi$. A formula ϕ is *valid* (in H) if $H,\pi,\rho \models \phi$ for all interpretations π and states ρ. We extend the interpretation π to terms $\mathbf{t}$ of various sorts as follows. Given the term $\mathbf{t}$, a state ρ and an interpretation π, we define the interpretation $[\![\mathbf{t}]\!]_{\pi,\rho}$ of X with respect to π and ρ as follows. Basis variables are interpreted as bases: $[\![\mathtt{b}]\!]_{\pi,\rho} = \pi(\mathtt{b})$.

If $\mathtt{b}$ is a basis variable, we interpret $\mathtt{b}$-formulas as projection operators on H_n: $[\![\mathtt{b}_i]\!]_{\pi,\rho} = |\psi'\rangle\langle\psi'|$, where $\psi' = \pi(\mathtt{b})_i$; and $[\![\alpha_1 \wedge \alpha_2]\!]_{\pi,\rho} = [\![\alpha_1]\!]_{\pi,\rho} \cdot [\![\alpha_2]\!]_{\pi,\rho}$ (this is the projection onto the intersection of the subspaces of H that are the images of the projectors $[\![\alpha_1]\!]_{\pi,\rho}$ and $[\![\alpha_1]\!]_{\pi,\rho}$). $[\![\neg\alpha]\!]_{\pi,\rho} = [\![\alpha]\!]^{\perp}_{\pi,\rho}$ is the projection operator projecting onto the orthogonal complement of the image of H under $[\![\alpha]\!]_{\pi,\rho}$. The transformation matrix terms, $m_{ij}(\mathtt{b},\mathtt{c})$, are interpreted as complex numbers: $[\![m_{ij}(\mathtt{b},\mathtt{c})]\!]_{\pi,\rho} = u_{ij}$, where $U = (u_{ij})$ is the unitary matrix such that $U\pi(\mathtt{b}) = \pi(\mathtt{c})$; The interpretation of the two probability terms are:

$$[\![P(\alpha)]\!]_{\pi,\rho} = \mathtt{Tr}([\![\alpha]\!]_{\pi,\rho} \cdot \rho), \text{ and}$$

$$[\![T(\mathtt{b}_i,\mathtt{c}_j)]\!]_{\pi,\rho} = \mathtt{Tr}([\![\mathtt{b}_i]\!]_{\pi,\rho}[\![\mathtt{c}_j]\!]_{\pi,\rho}).$$

Here $\mathtt{Tr}$ denotes the trace operator. Recall that the trace of an operator is the sum of diagonal elements of a matrix representing the operator. It is independent of the representation. For a pure state this interpretation coincides with the one given in the last chapter.

The intended interpretation of the t-operator is as a tensor product. It connects spaces of different dimension. If we are to allow unrestricted tensor product then in contrast to the interpretation of $\mathcal{L}_n(P,\mathbf{m})$ the Hilbert space can not be fixed. Thus, let

$$H \equiv \sum_{n=1}^{\infty} H_n$$

The sum is the direct sum of Hilbert spaces, i. e. $H_i \cap H_j = \{0\}$ for $i \neq j$ and by definition the vectors in H are *finite* sums of vectors from the component subspaces H_i. If $\alpha = \sum \alpha_i$ and $\beta = \sum \beta_i$ are two vectors in H, with $\alpha_i, \beta_i \in H_i$ then define the inner product $\langle \alpha | \beta \rangle = \sum_i \langle \alpha_i | \beta_i \rangle$ in H. We observe that there is a natural injection $i : H_n \to H$ where $\mathbf{x} \in H_n$ is mapped to the vector in H which has all but the nth component zero. Hence, we identify H_n with its image under this map.

Unlike the spaces considered so far H is infinite dimensional. Although, it is not required in what follows H is actually a Hilbert space. There is one technicality required of H to demonstrate this fact viz. that H is *complete* in the sense that every Cauchy sequence in H converges to a limit in the topology induced by the inner product defined in H. The proof is not too difficult but I omit it since I do not use the completeness property in the subsequent analysis. I only remark that H has a resemblance to Fock space- an important infinite dimensional space in quantum field theory.

We now equip each subspace $H_n = \mathbb{C}^n$ with the standard scalar product. Thus, if $\alpha = (y_1, \ldots, y_n)^T$ and $\beta = (z_1, \ldots, z_n)^T$ then

$$< \alpha, \beta > \equiv \sum_i \overline{y_i} z_i$$

We next define the tensor product of two matrices. Let

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ldots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} \text{ and } B = \begin{pmatrix} b_{11} & \cdots & b_{1q} \\ \vdots & \ldots & \vdots \\ b_{p1} & \cdots & b_{pq} \end{pmatrix}$$

be two matrices of order $m \times n$ and $p \times q$ respectively. Then the tensor or Kronecker product [MM92] of A and B is an $mp \times nq$ matrix given by (in block form)

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ldots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{pmatrix}$$

The tensor product of two vectors α and β as above is then given by

$$\alpha \otimes \beta = \begin{pmatrix} y_1 z_1 \\ y_1 z_2 \\ \vdots \\ y_1 z_n \\ \vdots \\ y_m z_n \end{pmatrix}$$

We use a concrete vector space and tensor product for convenience.

The formulas that we consider are homogeneous. That is, the probability operator P acts on basis formulas of some fixed dimension as earlier. However, the basis formulas themselves may be tensor product of basis terms of lower dimension. For formulas in dimension n, the states at which they are interpreted are elements of $\mathcal{H}_n$, density matrices of order n. Define the interpretation of basis terms- both atomic and composite- as follows.

$$\pi(t(\mathsf{b}^{(m)}, \mathsf{c}^{(n)}) = \pi(\mathsf{b}^{(m)}) \otimes \pi(\mathsf{c}^{(n)}) \text{ and}$$
$$\pi(t(\mathsf{b}_i^{(m)}, \mathsf{c}_j^{(n)})) = \pi(\mathsf{b}_i^{(m)}) \otimes \pi(\mathsf{c}_j^{(n)})$$

These equations express that if $\pi(\mathsf{b}^{(m)}) = \{|\alpha_0\rangle, \ldots, |\alpha_{m-1}\rangle\}$ and $\pi(\mathsf{c}^{(n)}) = \{|\beta_0\rangle, \ldots, |\beta_{n-1}\rangle\}$ are two bases in H_m and H_n respectively then $\pi(t(\mathsf{b}^{(m)}, \mathsf{c}^{(n)})$ is the basis in H_{mn} consisting of the vectors $\{|\alpha_i\rangle \otimes |\beta_j\rangle\}$. Recall the notation of Section 5.1.1. By definition, $t(\mathsf{b}^{(m)}, \mathsf{c}^{(n)})_k = t(\mathsf{b}_{r_1(k)}, \mathsf{c}_{r_0(k)})$. That is one specifies an ordering of the mn basis components as pairs $\{r_0(k) = k \bmod n,\ r_1(k) = [k/n]\}$. This ordering is consistent with the fact that $\langle \alpha \otimes \beta | \gamma \otimes \delta \rangle = \langle \alpha | \gamma \rangle \langle \beta | \delta \rangle$. This simple observation has deep physical and computational implication: the probabilities in a product state is the product of probabilities with respect to the factors. But if the state is not a product state but a more general *entangled* state then this is no longer true. In the classical domain too product states are used to describe a complex system in terms of components- for example concurrent systems. There we usually take the state space of the larger system as the *cartesian*

product of the individual systems. A consequence of this view is that the states of the individual systems can be altered independently. Some individual actions may change the global state but local actions leave all states other than the one where action is applied unaffected. In quantum systems the state space of a composite system are given by a tensor product space. A product state like $|\alpha\rangle \otimes |\beta\rangle$ behaves like classical product states in the sense that knowledge of the component states completely determines the global state. This is no longer true in an entangled state. Local action on an individual component can affect the state of others. This is crucial in many quantum cryptographic protocols which I discuss in the next chapter.

Note that terms involving (unitary) transition matrices are independent of the state ρ at which they are interpreted. Further, if $m_{ij}(\mathsf{b}^{(m)}, \mathsf{c}^{(m)}) = u_{ij}$ and $m_{ij}(\mathsf{b}'^{(n)}, \mathsf{c}'^{(n)}) = v_{ij}$ then $m_{ik,jl}(t(\mathsf{b}^{(m)}, \mathsf{b}'^{(n)}), t(\mathsf{c}^{(m)}, \mathsf{c}'^{(n)})) = u_{ij}v_{kl}$. Thus, if $U = (u_{ij})$ $V = (v_{ij})$ are the the transformation matrices of the above bases then the transformation matrix W of tensor products of bases is the tensor product of U and V. I will discuss some examples in the next section to illustrate these constructions.

Finally, for measurements

$$H, \pi, \rho \models M_{\phi_1,\ldots,\phi_k}(\Psi) \text{ iff } H, \pi, \sum_{i=1}^{k} \pi(\phi_i)\rho\pi(\phi_i) \models \Psi. \tag{5.2}$$

Recall that the measurement operator is defined with respect to a complete set of $\mathcal{B}$-formulas $\{\psi_1, \ldots, \psi_k\}$ i.e. formula $M_{\mathsf{b}}(\Psi)$ is true at a state ρ iff Ψ is true in the post-measurement state $\sum_i \pi(\phi_i)\rho\pi(\phi_i)$. Let us take a closer look at the semantics of this operator in an important special case when the measurement is maximal. Then, the 'projection' terms ϕ_i are 1-dimensional projectors onto the subspaces spanned by the basis components. In other words, the measurement is with respect to $\{\mathsf{b}_0, \ldots, \mathsf{b}_{n-1}\}$ in n-dimensions. I write the corresponding operator simply as, M_{b}. Writing the interpretation

$\pi(\mathsf{b}_i) = |\alpha_i\rangle\langle\alpha_i|$, the post measurement state is

$$\sum_{i=0}^{n-1} |\alpha_i\rangle\langle\alpha_i|\,\rho\,|\alpha_i\rangle\langle\alpha_i| = \sum_i (\langle\alpha_i|\rho|\alpha_i\rangle)\,|\alpha_i\rangle\langle\alpha_i| = \sum_i p_i\,|\alpha_i\rangle\langle\alpha_i|$$

where $p_i = \sum_i \langle\alpha_i|\rho|\alpha_i\rangle$ is the probability that the outcome of the measurement is α_i. That is, the post measurement state of an *ensemble* is the "weighted average" of the possible outcomes. If we have a large number of copies of the system, all prepared in the same state, then a maximal measurement or test will roughly distribute the ensemble among the n possible outcome states with fraction p_i in state α_i. It is easy to visualize a parallel *classical* picture. Suppose, we have a boolean algebra $\{B, \oplus, \cdot\}$ with generators $x_1, \ldots, x_n$. Let z be another boolean variable which is set equal to x_i with probability p_i. For example, letting $i = 6$ we could toss a 'loaded' dice with probability of i showing up equal to p_i, $i = 1, \ldots, 6$. Then, we can write z as a *formal sum* $p_1x_1 + \ldots + p_nx_n$. If $z' = p'_1x_1 + \ldots + p'_nx_n$ is another such variable then define $z \oplus z' \equiv \sum_{ij} p_ip'_j(x_i \oplus x_j)$ and similarly for $z \cdot z'$. It is easy to verify that this definition is consistent with the joint probability distributions, that is, $z \oplus z'$ equals $x_i \oplus x_j$ with probability $p_ip'_j$. Of course, this is not a boolean operation. The quantum case is much more subtle and I will not pursue this analogy further. A measurement of the type M_{b} defined above is called a maximal measurement or test. In principle, it is always possible to extend any measurement to a maximal one provided one has complete information about the state space of the system. This assumption is not realistic in some cases, for example, when we have to model the "environment" as part of the system. I will, however, make this assumption in the present chapter as it would make the axiomatization simpler. Hence, a measurement will mean a maximal measurement in this chapter. I note that this assumption is not at all restrictive as far as quantum computation is concerned [NC01].

The semantics of the operator for selective measurements is slightly more complicated. It is given by,

$$H, \pi, \rho \models S_\phi(\Psi) \text{ iff}$$
$$H, \pi, \pi(\phi)\rho\pi(\phi)/[\![P(\phi)]\!]_\rho \models \Psi \text{ and } [\![P(\phi)]\!]_\rho \neq 0$$

Informally, the selection operator picks up those systems in the ensemble whose post-measurement state lies in the subspace denoted by the basis formula ϕ. Obviously, this is not possible if the ensemble is in a state orthogonal to ϕ since then the probability $P(\phi)$ of obtaining a state in ϕ is 0. Recall that an arbitrary basis formula $\phi[X]$ in the basis symbol X(perhaps with tensor operators), can be written as a disjunction over basis components X_i. Thus, let $\phi[X] \equiv X_{i_1} \vee \ldots \vee X_{i_k}$. Then,

$$H, \pi, \rho \models S_\phi(\Psi) \text{ iff}$$
$$H, \pi, \sum_r [\![P(X_{i_r})]\!]_{\pi,\rho}\pi(X_{i_r})/[\![P(\phi)]\!]_{\pi,\rho} \models \Psi \text{ and } [\![P(\phi)]\!]_{\pi,\rho} \neq 0.$$

In particular,

$$H, \pi, \rho \models S_{X_i}(\Psi) \text{ iff}$$
$$H, \pi, \pi(X_i) \models \Psi \text{ and } [\![P(X_i)]\!]_{\pi,\rho} \neq 0.$$

The semantics of $S_\phi(\Psi)$ captures the notion of selection or filtration of a state or a subspace of the state space. Thus, $S_\phi(\Psi)$ is true at a state if after the measurement, the outcome is found to be in the the subspace K generated by the vectors $\pi(X_{i_r})$. As explained above after the measurement when the outcome is found to be in the subspace K, the state is given by a weighted sum over the basis vectors spanning K. The weights are precisely the conditional or relative probabilities assigned *a posteriori* to these basis vectors. If the pre-measurement state happens to be orthogonal to the subspace represented by ϕ then selection of a state in ϕ is not possible. The *irreversible* effect of a measurement is briefly discussed in the chapter on

quantum theory. As in the case of the general measurement M, I restrict to a special type of selective measurements S_{X_i} which corresponds to selecting or filtering *states* not subspaces. We have seen that in the case of *ensembles* one can give a reasonable explanation of post-measurement state. What if we have only a single or a small number of copies of the system and not an ensemble? A measurement in some basis will yield a definite result, not a mixed state. But imagine that between two observers Alice and Bob, Alice is performing measurements on a *single* quantum system **S**. Suppose that, Bob knows what measurement has been performed, that is, the *basis* chosen by Alice for the measurement, but does not know the outcome. Then, he can only assert that the post-measurement state is a mixed state as above. This is subjective probability as opposed to the objective probability based on the frequency/ensemble interpretation. This sort of subjective probability comes into play in the quantum cryptographic protocols. Note also that Alice who knows the outcome of the measurement has definite information about the state. Hence, the operator capturing Alice's epistemic state is the selection operator S_{X_i}. More about this point in the next chapter. We conclude this chapter with a simple but interesting result.

Lemma 15 *Let L denote the either of the measurement operators above. Let Φ and Φ' be two (homogeneous) formulas of same dimension. Then the following are valid. For $L = M_{\psi_1\psi_2\cdots\psi_k}$*

$$L(\Phi \wedge \Phi') \Leftrightarrow L\Phi \wedge L\Phi' \text{ and } \quad L(\neg\Phi) \Leftrightarrow \neg L\Phi$$

For $L = S_\psi$ the corresponding formulas are

$$L(\Phi \wedge \Phi') \Leftrightarrow L\Phi \wedge L\Phi' \text{ and } \quad P(\psi) \neq 0 \Rightarrow L(\neg\Phi) \Leftrightarrow \neg L\Phi$$

Moreover, from the validity of Φ we may infer the validity of $L\Phi$ if $L = M_{\psi_1\psi_2\cdots\psi_k}$. Also the validity of Φ and $P(\psi) \neq 0$ implies that $S_\psi(\Phi)$ is valid.

Proof: The second statement of the lemma is trivial. For by definition, $\pi, \rho \models L\Phi$ iff $\pi, L\rho \models \Phi$ where

$$L\rho = \sum_{i=1}^{k} \pi(\phi_i)\rho\pi(\phi_i) \text{ for } L = M_{\{\phi_i\}} \text{ and}$$

$$L\rho = \pi(\phi)\rho\pi(\phi)/[\![P(\phi)]\!]_{\pi,\rho} \text{ for } L = S_\phi$$

Since $L\rho$ is a state validity of Φ implies that of $L\Phi$.

For the first statement note the following. Once all the variables and basis symbols occurring in a formula are assigned values by the interpretation π then they are like propositional formulas to be evaluated at some state. Let us keep the interpretation π fixed. Then L acts like a *modal* operator with Kripke semantics [Gol92]. We simply define a relation $\sim$ among states by requiring $\rho \sim \rho'$ iff $\rho' = L\rho$ for $L = M_{\psi_1\psi_2\cdots\psi_k}$. For $L = S_\phi$, $\rho \sim \rho'$ iff $\mathtt{Tr}(\rho\pi(\phi)) \neq 0$ and $\rho' = L\rho$. We thus see that

$$\pi\rho \models \Phi \text{ iff } \rho \sim \rho' \text{ implies } \pi\rho' \models \Phi$$

Now, L is a function on the states. That is, the corresponding Kripke frames are *functional.* For functional frames the above formulas are valid (see the reference above) and the theorem is proved.

□

5.2 Examples

5.2.1 Quantum Gates

In this section we discuss some simple examples mainly drawn from quantum computation. The language used to write$\mathcal{L}_2(P, \mathbf{m}, t, M, S)$ as the qubits are 2-dimensional quantum systems. The formulas given for the gates are the definition of some unitary operators acting on on quantum systems composed 2-dimensional subsystems. These basic gates will be used in the next

chapter where the main applications are given including formulas for complex circuits built from these gates. I also add the usual circuit symbols used in quantum computing.

1. **Pauli-X Gate**

$$X \stackrel{\text{def}}{=} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad —\boxed{X}—$$

2. **Pauli-Y Gate**

$$Y \stackrel{\text{def}}{=} \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad —\boxed{Y}—$$

3. **Pauli-Z Gate**

$$Z \stackrel{\text{def}}{=} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad —\boxed{Z}—$$

4. **Hadamard Gate**

$$H \stackrel{\text{def}}{=} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad —\boxed{H}—$$

5. **Phase Gate**

$$S \stackrel{\text{def}}{=} \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad —\boxed{P}—$$

6. $\pi/8$ **Gate**

$$T \stackrel{\text{def}}{=} \begin{pmatrix} 1 & 0 \\ 0 & \frac{1+i}{\sqrt{2}} \end{pmatrix} \quad —\boxed{P}—$$

Since the U-operators on a probability formula Φ are interpreted as U^{-1} acting on the state, the gates presented here are adjoint (or transposed conjugate) of the standard forms[NC01]. Except for the phase gate they are identical to standard form since Pauli and Hadamard gates are both unitary and hermitian. These are single qubit gates. To get two entangled qubits that produce truly non-classical states we need multiple qubit gates. I present below some of the standard ones as a formula in $\mathcal{L}_2(P, \mathbf{m}, t, M, S)$ and the corresponding matrix. I use the notation defined in equation5.1. The pictorial representation is also given.

1. **CNOT Gate** Let us compare the representation of the 2-qubit **CNOT** gate in the two languages. It is a 4×4 matrix. The formula in terms of basis transformation matrix m_{ij} is

$$C \equiv m_{ij}(t(\mathsf{b},\mathsf{b}),\mathsf{c}) = \delta_{r_1(i)r_1(j)}[\delta_{r_1(i)1}(1-\delta_{r_0(i)r_0(j)}) + \delta_{r_1(i)0}\delta_{r_0(i)r_0(j)}].$$

In the matrix notation it may written as follows. I also put the standard circuit symbol for the gate.

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

Intuitively, the **CNOT** gate works as follows. If the top qubit (called the control) is in the state $|0\rangle$ then the second qubit (the target) is unaffected. If the control is in state $|1\rangle$ then the target is inverted. In other words

$$|00\rangle \to |00\rangle\,, \quad |01\rangle \to |01\rangle \tag{5.3}$$

$$|10\rangle \to |11\rangle\,, \quad |11\rangle \to |10\rangle \tag{5.4}$$

Note that the control is not affected. It is easy to translate the formula C into the corresponding matrix C. I have discussed before the interpretation of the functions r_i. The tensor product bases induce an ordering *viz.* the lexicographic ordering on the product space. In the context of 2-dimensional spaces this is the natural ordering with the numbers written in binary provided we follow the convention that all the binary strings representing numbers are of the same length. The functions $r_0(i), r_1(i), \ldots$ are the bits in that representation of i, $r_0(i)$ being the least significant bit. For the C-NOT gate The first delta function represents a formula expressing that in this representation the first qubit is unchanged and the second one flips if the the first qubit is 1. I further elaborate on this point with the Toffoli gate below.

2. **Toffoli Gate.** This is a 3-qubit gate defined as follows.

$$\text{Tof} \equiv m_{ij}(t(\mathsf{b},\mathsf{b},\mathsf{b}),\mathsf{c}) = \delta_{r_2(i)r_2(j)}\delta_{r_1(i)r_1(j)}\cdot$$
$$[\delta_{r_1(i)1}\delta_{r_2(i)1}(1-\delta_{r_0(i)r_0(j)}) + \delta_{r_1(i)0}\delta_{r_0(i)r_0(j)} + \delta_{r_2(i)0}\delta_{r_0(i)r_0(j)}]$$

The corresponding matrix T is 8×8. The rows and columns of the matrix are ordered from 0 to 7 written in binary e. g. $5 = 011 = r_2(5)r_1(5)r_0(5)$. In the matrix below the row and column indices are written at the top and the left of the matrix respectively.

	000	001	010	011	100	101	110	111
000	1	0	0	0	0	0	0	0
001	0	1	0	0	0	0	0	0
010	0	0	1	0	0	0	0	0
011	0	0	0	1	0	0	0	0
100	0	0	0	0	1	0	0	0
101	0	0	0	0	0	1	0	0
110	0	0	0	0	0	0	0	1
111	0	0	0	0	0	0	1	0

Then it is easy to read out the matrix elements from the formula. It is clear that a formula, whose size is linear in the dimension of the space concerned, can always be found to represent any given unitary matrix. This fact will be more formally stated later. Let us now deal with a more complex gate.

3. **Controlled-U.** Let $|i\rangle$ and $|j\rangle$ be m-qubit and n-qubit states respectively in the computational basis. They are vectors in respective dimensions $M = 2^m$ and $N = 2^n$. The integers i and j are written as binary strings of length m and n respectively. Let U be a unitary operator of order N. The controlled-U operation $CN-U$ is then defined

as

$$|i\rangle\,|j\rangle \longrightarrow |i\rangle\,|j\rangle \text{ if } i \neq M-1$$
$$|M-1\rangle\,|j\rangle \longrightarrow |M-1\rangle\,U\,|j\rangle\,.$$

Only in the case when the control qubits are in the state $|M-1\rangle$, that is, if all the individual qubits constituting the control system are in state $|1\rangle$, is the operator U applied to the target system. Otherwise, the whole system is left unchanged. The matrix for the gate is

$$\mathbf{CN}\text{-}U(ij) \equiv \delta_{r_1(i)r_1(j)}[(1-\delta_{i,M-1}\delta_{r_0ir_0j}) + \delta_{i,M-1}U(r_0(i)r_0(j))] \quad (5.5)$$

It is now a simple matter to write it in $\mathcal{L}_2(P, \mathbf{m}, t, M, S)$.

$$\bigwedge_{ij}(m_{ij}(\mathsf{b},\mathsf{c}) = \mathbf{CN}\text{-}U(ij))$$

The $CN-U$ gate is an operator in dimension MN. Recall that the functions r_i, $i = 0, 1$ in this context are defined as $r_0(i) = i \pmod N$ and $r_1(i) = [i/N]$. The controlled operations, although quite simple in appearance, carry one of the most powerful resources of quantum systems - entanglement. The single qubit operations can be efficiently simulated by a classical Turing machine. Hence, we need at least one 2-qubit operation to reach non-classical states. This can be done by the controlled operations. In fact, just the controlled-Not(CNOT) gate along with single qubit gates suffices to approximate an arbitrary unitary operator.

5.2.2 Characterization of states

The notion of entanglement is crucial to many quantum algorithms and protocols. Consider a composite quantum system consisting of two subsystems of dimension N. The dimension of the entire system is N^2. A *pure* state $\rho = |\alpha\rangle\langle\alpha|$ is called a product state if it can be written as

$$|\alpha\rangle = |\alpha_1\rangle \otimes |\alpha_2\rangle$$

otherwise it is an *entangled* state. For mixed states the situation is bit more complex. A mixed state is defined to be separable if it can be written as a convex combination of product states. Thus, a separable state

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i| \,,\; p_i \geq 0 \text{ and } \sum_i p_i = 1$$

and each pure state $|\psi_i\rangle$ is a product state. Let us write formulas characterizing these different types of states in n-qubit system.

1. **Pure state.** $P(\mathsf{b}_0) = 1$ This formula is satisfiable only at pure states.

2. **Pure product states.** Similarly the formula $P(t^n(\mathsf{b}_0)) = 1$ is satisfiable only at pure product states.

3. **Separable states.**

$$\exists_i p_i \forall_j x_j ((p_i \geq 0 \wedge \sum_i p_i = 1) \wedge$$
$$(P(t(\mathsf{b}^i, \mathsf{c}^i)_0) = p_i \wedge P(X_j) = x_j) \Rightarrow x_j = \sum_i T(t(\mathsf{b}^i, \mathsf{c}^i)_0, X_j) p_i$$

The subscripts on the quantifier indicate that the quantification is over all the variables with subscripts ranging over a finite set. The range of the subscript j is the set $\{0, \ldots, N^2 - 1\}$. Now a general mixed state can be written as

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$$

The state ρ is separable if there is such representation such that each ψ_i is a product state. The above formula is satisfiable only at a separable state.

5.3 Axiomatization

In the preceding sections we developed the syntax and semantics of the languages $\mathcal{L}(P, \mathbf{m}, t, M, S)$. We now turn our attention to the proof theory

and questions of complexity. In a loose sense, $\mathcal{L}(P, \mathbf{m}, t, M, S)$ is an extension of the language $\mathcal{L}_n(P, \mathbf{m})$ whose axiomatization was presented in the last chapter. In the latter case the dimension(=n) is fixed. But now, since we have to deal with tensor product, several dimensions may be involved. Therefore, the dimension is explicitly stated in the formulas wherever there is possibility of ambiguity. Further, the axioms of last chapter are assumed for each dimension $n > 0$. However, there are some significant differences even in the case of fixed dimension since the states in which the formulas are now interpreted are general 'mixed states' or density matrices instead of the pure states used in the last chapter. Consequently, I state all the axioms of the last chapter and indicate which ones are modified. But the intuitive meanings are discussed only in the case of modified axioms. It would be convenient to group them under different headings. There are two kinds of basis terms. The atomic basis variables and the composite basis expressions built out of the atomic variables. In case of the latter, the atomic basis variables appearing in the product basis terms determine the dimension and the ordering of the composite bases. For example, $t(\mathtt{b}^{(m)}, \mathtt{b}^{(n)})$ represents a composite basis in dimension mn(see 5.1.1). The symbols $\mathtt{X}, \mathtt{Y}, \mathtt{Z}, \cdots$ will denote basis expressions of either type. Sometimes, the dimension will be indicated explicitly. Hence, $\mathtt{X}_0^{(k)}, \ldots, \mathtt{X}_{k-1}^{(k)}$ will denote the basis components for the expressions of dimension k.

1. **Boolean axioms for basis variables**

   For every basis expression $\mathtt{X}$ in dimension n

   $\mathbf{B}^{(n)}\mathbf{1}$ $\quad \mathtt{X}_0 \vee \ldots \vee \mathtt{X}_{n-1}$

   $\mathbf{B}^{(n)}\mathbf{2}$ $\quad \neg(\mathtt{X}_i \wedge \mathtt{X}_j)$ $\quad$ for $i \neq j$ and $i, j \in J_n$

   As before we say that a $\mathtt{X}$-formula ϕ is a $\mathtt{X}$-tautology, and write $\vdash_{\mathtt{X}} \phi$ if it is a tautology of ordinary propositional logic or it can be derived from the above axioms alone using propositional logic inference rules. Next, the probability axioms.

2. **Probability Axioms**

   P1 $0 \leq P(\phi) \leq 1$

   P2 $P(\phi) = 1$ if ϕ is a X-tautology

   P3 $P(\phi_1 \wedge \phi_2) + P(\phi_1 \wedge \neg\phi_2) = P(\phi_1)$

   P4 $P(\phi_1) = P(\phi_2)$ if $\phi_1 \Leftrightarrow \phi_2$ is a X-tautology

3. **Unitary operator axioms** For a pair of basis expressions X, Y of same dimension we have the following axioms for unitary transformation.

   M1 $m_{ij}(\mathtt{X}, \mathtt{Y}) = \overline{m_{ji}(\mathtt{Y}, \mathtt{X})}$

   M2a $m_{ij}(\mathtt{X}, \mathtt{X}) = 1$ if $i = j$ **M2b** $m_{ij}(\mathtt{X}, \mathtt{X}) = 0$ if $i \neq j$

   M3 $m_{ij}(\mathtt{X}, \mathtt{Z}) = \Sigma_{k=0}^{n-1} m_{ik}(\mathtt{X}, \mathtt{Y}) m_{kj}(\mathtt{Y}, \mathtt{Z})$

   Again the unitary operators contain information about the dimension n, that is, their order as a matrix is unambiguous. All integer constants i, j etc. are assumed to vary in the range $\{0, n-1\}$. They are *not* variables in the object language. The difference due to the more general interpretation in "mixed states" appears in the the axiom $\mathbf{MP}'_k$ (see subsection 4.4.3). This is the basic consistency axiom. Intuitively, the axiom requires that the probability assignments be consistent with quantum theory.

   $\mathbf{MP}'_k$ $\forall x_0 \ldots x_n : \mathbb{R}.(\bigwedge_{i=1}^{n} P(\mathtt{X}_i^0) = x_i \Rightarrow$
   $\exists z_{11} z_{12} \ldots z_{nn} (\bigwedge_{i=1}^{n} z_{ii} = x_i \wedge \forall y_1 \ldots y_n (\sum_{ij} \overline{y_i} z_{ij} y_j \geq 0) \wedge$
   $\bigwedge_{j=1}^{k} \bigwedge_{i=1}^{n} P(\mathtt{X}_i^j) = \Sigma_{r,s=1}^{n} \overline{m_{ir}(X^0, X^j)} z_{rs} m_{is}(\mathtt{X}^0, \mathtt{X}^j)))$

   This is the most complicated axiom scheme. It simply states that given probability distribution of k bases $\{\mathtt{b}^1, \ldots \mathtt{b}^k\}$ along with the transformation matrices there is a state $\rho = (z_{ij})$ which gives rise to

these distributions. This is valid for all k in an n-dimensional quantum system. This can be proved with little modification to adapt to the density matrix semantics as in the last chapter 1. In dimension n one requires that $\mathbf{MP}_k$ be satisfied for all $k \leq n^2 - n + 1$. Then it is satisfied for all positive integers k. A similar result was proved in the last chapter 10 and since that proof can be be quite easily adapted to the density matrix semantics we omit it here. But observe a difference with the case where states are required to be pure states. In that case $\mathbf{MP}'_k$ is required to be satisfied only for $k \leq n^2 - 2n + 1$. This was observed in Corollary 1.

4. The axioms of tensor operator come next.

Tensor1 $\quad t(t(\mathtt{X}^{(m)}, \mathtt{Y}^{(n)}), \mathtt{Z}^{(p)}) \sim t(\mathtt{X}^{(m)}, t(\mathtt{X}^{(n)}, \mathtt{Z}^{(p)}))$

Tensor2 $\quad P(t(\mathtt{X}^{(m)}, \mathtt{Y}^{(n)})_0) = 1 \Rightarrow$
$P(t(\mathtt{X}'^{(m)}, \mathtt{Y}'^{(n)})_j) = P(t(\mathtt{X}'^{(m)}_{r_1(j)}, \top^n)) \cdot P(t(\top^m, \mathtt{Y}'^{(n)}_{r_0(j)}))$

Tensor3 $\quad m_{ij}(t(\mathtt{X}^{(m)}, \mathtt{Y}^{(n)}), t(\mathtt{X}'^{(m)}, \mathtt{Y}'^{(n)})) =$
$m_{r_1(i),r_1(j)}(\mathtt{X}, \mathtt{X}') \cdot m_{r_0(i),r_0(j)}(\mathtt{Y}, \mathtt{Y}')$

Recall that the notation $\mathtt{b} \sim \mathtt{c}$ is a shorthand for $\wedge_{ij} m_{ij}(\mathtt{b}, \mathtt{c}) = \delta_{ij}$. The first axiom expresses the associativity of the tensor product. Note that both sides have the same dimension. Similarly, the intuition behind the second axiom **Tensor2** is the fact that if the system is in a (tensor) product $|\phi\rangle\,|\psi\rangle$ state then the probability distribution of the states in $|\alpha_i\rangle\,|\beta_j\rangle$ in a complete measurement in some other product basis $\{|\alpha_i\rangle\,|\beta_j\rangle\}$ equals the product of the distributions corresponding to measurement of the component systems. Classically, this is a very familiar situation. If we have separate trials of two independent experiments then the sample space of the combined experiments is the

product space of the individual spaces and the joint distribution is the product of the individual distributions. Note also that there is no special significance of the index 0 in **Tensor 2**. It can be replaced by *any* index i and be shown provably equivalent to the original axiom. We simple use a unitary matrix to interchange the basis vectors corresponding to indices and use **Tensor3**. The intuitive meaning of **Tensor3** is similar. If $U(ij) = m_{ij}(\mathsf{b}, \mathsf{c})$ and $V(ij) = m_{ij}(\mathsf{b}', \mathsf{c}')$ are the respective transformation matrices then the transformation matrix connecting the product bases $t(\mathsf{b}, \mathsf{b}')$ and $t(\mathsf{c}, \mathsf{c}')$ is the tensor or direct product $U \times V$.

The final axiom for the tensor operator deals with the commutation property of tensor product. Thus, we want to relate $t(\mathsf{b}^{(m)}, \mathsf{c}^{(n)})$ and $t(\mathsf{c}^{(n)}, \mathsf{b}^{(m)})$. First, note that both are interpreted as basis vectors in the Hilbert space $\mathbb{C}^{mn}$. There is a definite transformation matrix $P^{m,n}$ relating them. If A is any $m \times m$ matrix and B is a $n \times n$ then

$$P^{m,n}(A \otimes B) = (B \otimes A)P^{n,m}$$

Any basis in $\mathbb{C}^m$ may be viewed as a unitary matrix of order m, the basis vectors representing the columns [HJ91]. Thus, the above equation represents the relation between the bases $\pi(t(\mathsf{c}^{(n)}, \mathsf{b}^{(m)}))$ and $\pi(t(\mathsf{b}^{(m)}, \mathsf{c}^{(n)}))$

Hence, for a consistent interpretation we have to incorporate this in the next axiom. The matrix $P^{m,n}$ of order mn may be described as follows [HJ91]. Let I_{mn} be the identity matrix of order mn. We write the elements of $P^{m,n}_{ij}$ as

$$P^{m,n}_{ij} = P^{m,n}_{r_1(i)r_0(i),r_1(j)r_0(j)}, \; 0 \leq r_1(i), r_1(j) \leq m-1 \text{ and } 0 \leq r_0(i), r_0(j) \leq n-1$$

We have encountered the functions r_0 and r_1 before. They simply introduce an ordering in the set $\{0, \ldots, mn-1\}$. Explicitly, for $0 \leq$

$k \leq mn-1$, $r_0(k) = k \bmod n$ and $r_1(k) = \lfloor k/n \rfloor$. Intuitively, we order the indices $0, \ldots, mn-1$ as an $m \times n$ matrix. Dually, we may also order them as an $n \times m$ matrix. Let $r'_0(k) = k \bmod m$ and $r'_1(k) = \lfloor k/m \rfloor$ be the corresponding functions. Let $0 \leq k \leq mn-1$. Then the $(r_1(k), r_0(k))$th row is same as the $(r'_1(k), r'_0(k))$th row of I_{mn}. Explicitly,

$$P^{m,n}_{ij,kl} = \delta_{i,r'_0(nk+l)}\delta_{j,r'_1(nk+l)}$$

Now we write the axiom relating the bases.

Tensor4 $\wedge_{ij}\, m_{ij}(\mathsf{d}, t(\mathsf{b}^{(m)}, \mathsf{c}^{(n)})) = v_{ij}$
$\wedge_{ij}\, m_{ij}(\mathsf{d}, t(\mathsf{c}^{(n)}, \mathsf{b}^{(m)})) = u_{ij} \Rightarrow$
$\exists v_{ij}\mathbf{Unit}(v_{ij}) \bigwedge(\bigwedge_{ij} \sum_{klr} P^{m,n}_{ik} v_{kl} u_{lr} P^{n,m}_{rj} = \sum_{l} v_{il}u_{lj})$

The unitary matrix $V(ij) = v_{ij}$ simply connects an arbitrary basis to the standard basis. The permutation matrix connects the bases $\pi(t(\mathsf{b}, \mathsf{c}))$ and $\pi(t(\mathsf{c}, \mathsf{b}))$ when expressed in the standard basis. We will not use this axiom much as mostly products in definite order will be used in tensor product terms. However, we require this axiom for proving completeness of the axiomatization.

5. Next we come to the measurement axiom. Recall that there are two operators which are associated with measurement. The first is the measurement operator M_X, where X is a basis symbol(atomic or product).

Measure1 $\bigwedge_i P(\mathsf{b}_i) = x_i \Rightarrow \bigwedge_j (M_{\mathsf{b}}(P(\mathsf{c}_j) = \sum_k T(\mathsf{b}_k, \mathsf{c}_j)x_k))$.

First, we note that the measurement corresponds to a *complete* measurement. Recall that, by definition, the transition probability term $T(\mathsf{b}_k, \mathsf{c}_j) = |m_{kj}(\mathsf{b}, \mathsf{c})|^2$. I refer to the chapter on basics of quantum theory for the intuition behind the axiom. The post-measurement state of an ensemble then becomes a mixture (see the discussion on

semantics of measurement). For selective measurements $S_{\mathbf{b}_i}$ we have the axiom schema

Selection $(P(\mathbf{b}_i) \neq 0 \Leftrightarrow S_{\mathbf{b}_i}(P(\mathbf{b}_i) = 1)) \wedge (P(\mathbf{b}_i) = 0 \Leftrightarrow S_{\mathbf{b}_i}(\bot))$

The intuitive meaning is clear: if the observer "selects" a post measurement state corresponding to $\mathbf{b}_i$ the the probability of obtaining a state corresponding to $\mathbf{b}_i$ in a *subsequent* measurement is 1. This is, of course, provably equivalent to asserting that the post measurement *is* $\mathbf{b}_i$. But a state corresponding to $\mathbf{b}_i$ can be selected if and only if the probability of that state is not 0. If it *is* 0, then the set of states in which $S_{\mathbf{b}_i}(\Phi)$ is satisfiable is empty. This is expressed by saying that the unsatisfiable formula $\bot$ holds. Note that in the last two axioms concerning measurement we only require that they be satisfied for *atomic* bases. We will deduce the corresponding formula for product bases as a theorem. Denote by L either of the measurement operators. Then the axiom schema for boolean combinations is:

Measure2 $L(\Phi_1 \wedge \Phi_2) \Leftrightarrow L(\Phi_1) \wedge L(\Phi_2)$

Measure3a $M_{\mathbf{b}}(\neg\Phi) \Leftrightarrow \neg M_{\mathbf{b}}(\Phi)$

Measure3b $P(\mathbf{b}_i) \neq 0 \Rightarrow S_{\mathbf{b}_i}(\neg\Phi) \Leftrightarrow \neg S_{\mathbf{b}_i}(\Phi)$

We have seen in Section 5.1.3 that in the density matrix semantics of the language $\mathcal{L}(P, \mathbf{m}, t, M, S)$ the measurement operators behave like modal operators whose Kripke frames are functional. The last two axioms reflect this. They are not true in the state based semantics. Moreover, we have the following inference rule.

Cons1. From Φ infer $M_{\mathbf{b}}\Phi$.

Cons2. From Φ *and* $P(\mathbf{b}_i) \neq 0$ infer $S_{\mathbf{b}_i}\Phi$.

Let $\mathbf{Ax}\ (P, \mathbf{m}, t, M, S)$be the theory resulting from the above axiomatization of $\mathcal{L}(P, \mathbf{m}, t, M, S)$. We must keep in mind that it is an extension of the theory $\mathbb{RC}$. So, as in the last chapter (see Section 4.4.3) we reduce a formula of the language into an equivalent formula of $\mathbb{RC}$. There are several benefits of

this reduction. First, we deduce properties (completeness, decidability, etc.)of $\mathbf{Ax}\ (P, \mathbf{m}, t, M, S)$from those of $\mathbb{RC}$. As there are several new syntactic operators one has to modify the proofs of the similar results in the last chapter. However, as we shall see the modifications are quite straightforward. Therefore, many of the details will not be repeated. Second, the resulting formula of $\mathbb{RC}$ can easily be further reduced to an equivalent formulas of real closed fields. We have a decision methods for the later. So, the combination of the two algorithms provides us with an algorithm for problems concerning interesting aspects of QCI. This can be, in principle, implemented in a computer/Turing machine. Which brings us to the questions of complexity. Several complexity bounds will be derived and the practical aspects of the implementation will be discussed.

First I prove an important result on reduction of formulas in $\mathcal{L}(P, \mathbf{m}, t, M, S)$ to formulas in $\mathbb{RC}$. We have seen that in the last chapter this was extensively used to prove completeness results, devise decision procedures and find complexity bounds. In the last chapter we dealt with languages in some fixed dimension n. One could then introduce for a basis symbol $\mathtt{b}$, n complex variables representing a vector. Similarly, for matrix symbols we require at most n^2 variables. As a consequence for each formula of Φ of $\mathcal{L}_n(P, \mathbf{m})$ we could write an equivalent formula of $\mathbb{RC}$ whose length was of polynomial order in $|\Phi|$(see 12). The situation is more complicated for the languages in the present chapter. The reason is the presence of tensor product bases. For example, if $\mathtt{b}$ is interpreted as a basis in 2 dimension then $t^k(\mathtt{b})$, is a basis in 2^k dimensions. Thus we have to introduce 2^k variables for the corresponding basis and the $\mathbb{RC}$ formula may become exponentially large. We have to be more careful in formulating the reduction algorithm. The next theorem gives such an algorithm. It requires some restrictions which are stated below. First we recall the following facts. The atomic formulas of the language are obtained by substituting probability terms and matrix terms in the relations $p(z_1, z_2, \ldots, z_k) \geq 0$. The probability terms are of the form

$P(\Psi[X])$, where Ψ is a basis formula in the basis variable X which may be atomic or a tensor product term.

1. The probability formulas $P(\Psi[X])$ are homogeneous.

2. The general tensor term is built out of some fixed irreducible basis symbols $\{\mathtt{b}^1, \ldots, \mathtt{b}^k\}$ which have dimensions $n_1+1 \leq n_2+1 \cdots \leq n_k+1$ and the basis variables in a product term appear such that this order is preserved. Thus, in a term $t(\mathtt{b}^{i_1}, \mathtt{b}^{i_2}, \ldots, \mathtt{b}^{i_r})$ we have $i_1 \leq i_2 \leq \cdots i_r$. Here irreducible means that they are not product bases and that no basis symbol appears whose dimension is factor of any dimension n_i+1. That is, these dimensions are minimal. In QCI usually all the $n_i = 1$. This assumption simplifies the proof of the theorem. It is also the natural situation in the physical context. If we want to talk of a composite system consisting of state space $\mathcal{H}_m$ and $\mathcal{H}_n$ respectively, then the state space of the composite system is $\mathcal{H}_m \otimes \mathcal{H}_n$ *or* $\mathcal{H}_n \otimes \mathcal{H}_m$. For the description and analysis of the composite system the two spaces are perfectly equivalent. If $m = n$ the question is irrelevant.

Theorem 12 *Given a formula Φ of $\mathcal{L}(P, \mathbf{m}, t, M, S)$, satisfying the above conditions, a formula Φ' of $\mathbb{RC}$ can be constructed whose length is bounded by polynomial in the $|\Phi|$ such that Φ is satisfiable if and only if Φ' is satisfiable.*

Proof: The algorithm for constructing Φ' out of a given formula Φ in $\mathcal{L}(P, \mathbf{m}, t, M, S)$ is given below.

1. Scan Φ for all basis expressions. This includes the expressions appearing in the measurement operators. I use the notation $X, Y, Z, \ldots$ to denote basis symbols- both irreducible and composite. The "composite" basis symbols are the tensor operator terms. Let $\mathtt{b}^1, \ldots, \mathtt{b}^r$ be the irreducible basis symbols, with respective dimensions $\{n_1+1, n_2+1, \ldots, n_r+1\}$ which are minimal in the following sense: there is no basis term of dimension k which is a divisor of any n_i+1. The reason

for this distinction will become clear below. Let $B_\Phi \equiv \{X^1, \ldots, X^r\}$ be the set of basis expressions that appear in Φ. Note that the dimension of X^i's may be exponentially large. Let $B'_\Phi \subset B_\Phi$ be the set of basis symbols that appear in probability terms. By our assumption of homogeneity they have common dimension, say N. Further, we assume that the first $s(\leq r)$ of the X^i's are in B'_Φ. There is no loss of generality due to this assumption as it is not difficult to modify the following construction and analysis in the general case. Now, X^i may be an atomic basis symbol or a product.

2. Now scan for the indices of *components* of each of the basis expressions X, that is terms of the form X_i, $0 \leq i \leq \dim(X)$, that appear in Φ. We also record the appearance of the component indices i, j of the matrix symbol $m_{ij}(X, Y)$. Let T_X be the set of all such component indices of the basis expressions X appearing in Φ. The dimension of each of the basis expression is known. For integer k let

$$R^k = \bigcup_{X \in \Phi} \{T_X | \dim(X) = k\}$$

Then $\sum_k |R^k| \leq |\Phi|$.

Similarly, let B_X be the set of components of basis expression that appear in the probability terms. Clearly, $B_X \subset T_X$ for every basis variable X and B_{Xj} is nonempty only if $j \leq s$.

3. Let $m_k = |R^k|$ and $r_k = \min\{m_k + |\Phi|, k\}$. The integer k must be of the form $(n_1+1)^{i_1} \cdots (n_r+1)^{i_k}$. Let L^k be a set of cardinality r_k such that $R^k \subset L^k \subseteq \{0, \ldots, k-1\}$. We also require that for a product basis $Z \equiv t(X, Y)$, if $k \in T_Z$, then $r_1(k) \in T_X$ and $r_0(k) \in T_Y$ since by definition $t(X, Y)_k \equiv t(X_{r_1(k)}, Y_{r_0(k)})$.

4. Now we introduce a distinguished class of basis symbols Z^j for each irreducible dimension $n_j + 1$ appearing in the formula Φ and the cor-

responding "vector" variables

$$\mathbf{z}_i^j;\ 1 \leq j \leq r \text{ and } 0 \leq i \leq n_j$$

Each such vector variable is actually a collection of variables (of $\mathbb{RC}$) such that $\mathbf{z}_i^j$ consists of $n_j + 1$ variables, $z_{i0}^j, \ldots, z_{in_j}^j$. Intuitively, the vectors $\mathbf{z}_i^r$, for fixed r, will constitute a basis in $\mathbb{C}^{n_r+1}$. The collection of variables corresponding to $\mathbf{z}_i^r$ will be denoted by $\{z_{i0}^r, \ldots, z_{in_r}^r\}$.

For each $i \in L^k$ and each irreducible basis variable X of dimension k introduce a "vector" variable $\mathbf{x}_i$ of $\mathbb{RC}$ of dimension r_k. Below, we use the shorthand $\mathbf{x}_i$ for r_k variables $\{x_{ij} | i, j \in L^k\}$. For each basis symbol $X($ of dimension k) let $\mathbf{Bs}(X)$ be the following

$$(\bigwedge_{i,j \in L^k, i \neq j} (\sum_{l \in L^k} \overline{x}_{il} x_{jl} = 0)) \bigwedge (\bigwedge_i (\sum_{l \in L^k} |x_{il}|^2 = 1))$$

The intuition behind the formula $\mathbf{Bs}$ is a necessary and sufficient condition that the vectors $\mathbf{x}_i$ can be extended to an orthonormal basis. Similarly, let $\mathbf{Bs}(\mathbf{z}^r)$ denote the relation expressing orthonormality amongst the vectors $\{\mathbf{z}_0^r, \ldots, \mathbf{z}_{n_r}^r\}$.

5. Recall that the basis variables are sorted according to their dimension k. Assume that the new variables x_{ij} introduced also carry the corresponding dimension r_k. We can achieve this easily by introducing different sets of variables in each dimension. Note that a vector variable $\mathbf{x}_j^i$ is introduced for the basis component X_j^i that appears in Φ. Further, for each dimension kk' corresponding to composite or product bases of dimension k and k' in Φ we introduce the $\mathbb{RC}$-terms as follows. First, we treat $Z^{kk''}$ as the product basis $t(Z^k, Z^{k'})$. For any product basis component $t(X^{(k)}, Y^{(k')})_i$ we introduce $\mathbb{RC}$-terms $\mathbf{x}_i^{mn}$

$$\mathbf{x}_i^{mn} \equiv x^m_{r_1(i)r_1(j)} y^n_{r_0(i)r_0(j)} \tag{5.6}$$

It follows that we have terms $\mathbf{z}_i^{mn} \equiv z^m_{r_1(i)r_1(j)} z^n_{r_0(i)r_0(j)}$ corresponding to the distinguished basis $Z^{(kk')}$.

Recall that given m, n, $r_0(i) = i \bmod n$ and $r_1(i) = \lfloor \frac{i}{n} \rfloor$ for $0 \leq i \leq mn-1$. We are simply introducing the semantics of the tensor product into the formula. Let Φ_1 be the conjunction of all the formulas $\mathbf{Bs}(X)$, X an irreducible basis expression. Then $|\Phi_1| = O(|\Phi|^2)$.

6. Now for the matrix terms. Unlike the probability terms the matrix terms may involve basis variables in lower dimension. The special basis variables $\mathbf{z}_i^j$ that we had introduced may be visualized as the standard basis in dimension r_j. Let X be an irreducible basis variable in dimension $k = (n_{i_1}+1)^{j_1} \cdots (n_{i_r}+1)^{j_r}$. The right hand side is unique. Introduce, new variables $\{u_{ij}(Z^k, X),\ 0 \leq i, j \leq r_k\}$. Note the range of i, j. Although, the dimension of Z^k(and X) as basis expression in the object language $\mathcal{L}(P, \mathbf{m}, t, M, S)$ is k, the vectors representing them have dimension r_k. We continue to write u_{ij} as a function over the original expressions. Intuitively, these variables represent the entries of the unitary matrix relating the bases corresponding to Z^k and X.

$$\mathbf{Un}_k(Z, Y) \equiv (\bigwedge_{i \in L^k} (\sum_{l \in L^k} \overline{u_{il}} u_{il} = 1)) \wedge (\bigwedge_{i \neq j} (\sum_{l} \overline{u_{il}} u_{jl} = 0)) \qquad (5.7)$$

These are the unitary conditions saying that the variables u_{ij}, $1 \leq i, j \leq r_k$ are the entries of a unitary matrix of order $r_k \leq 2|\Phi|$. The important point to note is that we need to consider a matrix of order at most r_k only. Thus, even if k is large we need not examine extension to order k.

7. Next, we introduce a formula expressing the unitary transformation connecting the bases in Φ to the distinguished bases introduced above. Thus, for each pair of vector variables $(\mathbf{z}_i^k, \mathbf{y}_j)$ for basis components Z_i^k and Y_j resp. in dimension k and Y irreducible introduce the formula

$$\mathbf{H}(Z_i^k, Y_j) \equiv \bigwedge_{j,l \in L^k} y_{il} = \sum_{r \in L^k} u_{rl}(Z^k, Y) z_{jr}^k \qquad (5.8)$$

The intuition is that the vectors $\{\mathbf{z}_i^k\}$ and $\mathbf{y}_j$ form bases in $\mathbb{C}^{r_k}$ and $u_{ij}(Z^k, Y)$ is the unitary matrix connecting them.

8. Note first that, general tensor terms like $t(\phi_1, \phi_2)$, where $\phi_1[\mathsf{b}]$ and $\phi_2[\mathsf{c}]$ are basis formulas over some basis variables b and c respectively, can be reduced to a disjunction over terms of the form $t(\mathsf{b}_i, \mathsf{c}_j)$. This follows from the basic relation $t(\phi_1 \vee \phi_2, \psi) = t(\phi_1, \psi) \vee t(\phi_2, \psi)$. Hence, I assume that these reductions have been done. We will deal with negated expressions later.

9. By assumption the probability formulas are homogeneous of dimension N. Hence, they are evaluated at some state in $\mathbb{C}^N$. This number N may be of exponential order in the length of Φ. Therefore, we do not introduce variables for *all* the N possible components. Let $N' = \min\{2m_N, N\}$ and L^N be a set of N' integers containing R^N such that $L^N \subseteq \{0, \ldots, N-1\}$. Let $\{s_{ij} |\ i, j \in L^N\}$ be N'^2 variables of $\mathbb{RC}$ indexed by the integers from L^N. Let $L^N = \{k_1, \ldots, k_{N'}\}$ in some ordering of L^N. Define the formula

$$\begin{aligned} \mathbf{St}_\Phi \equiv & \exists v_{11} v_{12} \cdots v_{N'N'} \Big(\sum_{i,j \in L^N} \overline{v}_{ik} v_{jk} = \delta_{ij} \Big) \wedge \\ & \exists r_j \sum_{lm} v_{il} s_{k_l k_m} \overline{v}_{mj} = \delta_{ij} r_j \wedge r_j \geq 0 \wedge \\ & \sum_j r_j \begin{cases} \leq 1 \text{ if } N' < N \\ = 1 \text{ if } N' = N \end{cases} \end{aligned} \tag{5.9}$$

The last line of the above formula actually represents *two* formulas corresponding to the two possibilities. The formula $\mathbf{St}_\Phi$ is true for some interpretation of the variables s_{ij} if and only if they form a nonnegative definite matrix of order N' since the *unitary* matrix v_{ij} diagonalizes s_{ij} such that the diagonal entries are nonnegative and their sum is at most 1.

10. We make a further assumption on the form of basis formulas that they

are expressed either as a disjunction over non-negated basis variables or as a negation of such formula. From the semantics of the basis formulas we know that any formula can be expressed as a disjunction but for efficiency it is useful to admit the negated form also. For example, suppose we know that the system is in some state which is orthogonal to the first 2 basis vectors in a space of dimension N. Then, it is more efficient to express it as the formula $P(\neg(\mathtt{b}_1 \vee \mathtt{b}_2)) = 1$. If we had stuck to the first form we have to write it as disjunction over all but the first two components. Now define a map σ for the probability terms $P(A)$, A a basis formula, that appear in Φ . The image of σ is in the set of $\mathbb{RC}$ terms generated by the variables we have introduced above. Let Ψ, Ψ_1, Ψ_2, be boolean formulas over relevant basis variables such that Ψ_1, Ψ_2 are in non-negated disjunctive form and let Ψ_{12} be the formula defined as the disjunction of all the basis variables common to Ψ_1 and Ψ_2. Then, define

$$\sigma(P(X_i)) = \sum_{j,k \in L^N} \overline{x}_{ij} s_{jk} x_{ik} \tag{5.10}$$

$$\sigma(P(\Psi_1 \vee \Psi_2)) = \sigma(P(\Psi_1)) + \sigma(P(\Psi_2)) - \sigma(P(\Psi_{12})) \tag{5.11}$$

$$\sigma(P(\neg\Psi)) = 1 - \sigma(P(\Psi)) \quad \sigma(P(\top)) = 1 \text{ and } \sigma(P(\bot)) = 0 \tag{5.12}$$

The motivation for the above formula for $\sigma(P(X_i))$ is clear. If $\mathbf{x}_i$ are basis vectors for X_i in dimension N' and $\rho = (s_{ij})$ is a (mixed) state then $\sigma(P(X_i) = \mathtt{Tr}((\mathbf{x}_i{}^\dagger \mathbf{x}_i)\rho)$ is the probability of a maximal test in basis X yielding the result X_i.

11. Let Φ'' be the formula which is the conjunction of all $\mathbf{Bs}(X)$, $\mathbf{Un}_k(Z, Y)$, $H(X_i, Y_j)$ and $\mathbf{St}_\Phi$ constructed above. From the construction we conclude that $|\Phi''| = O(|\Phi|^3)$. The cube appears because the number of new matrix variables appearing in Φ'' is bounded by a number $b = O(|\Phi|^3)$ and there are at most $|\Phi|$ such variables. Now construct a formula $\Psi = \gamma(\Phi)$ as follows. First, let Φ be a "bare" formula,

that is, without the measurement operators. Then, by definition, Φ is constructed by substituting probability terms $P(\phi)$ and matrix terms $m_{ij}(X,Y)$ for some of the variables in a formula $\mathbf{F}$ of $\mathbb{RC}$. We write $\Phi = \eta(\mathbf{F})$ for this substitution. For each occurrence of $P(\phi)$ substitute the $\mathbb{RC}$ term $\sigma(P(\phi))$.

For the matrix terms first we construct terms corresponding to $m_{ij}(Z^k, Y)$ where Y is a basis symbol in dimension k. If Y is irreducible in dimension k then substitute

$$u_{jk}(X,Y) \equiv \sum_i \overline{u_{ij}(Z^k,X)} u_{ik}(Z^k,Y)$$

for $m_{ij}(X,Y)$. Let $Y = t(Y',Y'')$ be a product basis, where $k = k'k''$ and $\dim(Y') = k'$, $\dim(Y'') = k''$, . Then assuming we have constructed $u_{ij}(Z^k,Y')$ and $u_{ij}(Z^k,Y'')$ we define

$$u_{ij}(Z^k,Y) = u_{r_1(i),r_1(j)}(Z^{k'},Y) u_{r_0(i),r_0(j)}(Z^{k''},Y)$$

This way we obtain $\mathbb{RC}$-variables $u_{ij}(Z^k,Y)$, $i,j \in L^k$ for all basis symbols appearing Φ, substitute $u_{ij}(X,Y)$ in $\mathbf{F}$. Denote the $\mathbb{RC}$-formula so obtained by $\gamma(\Phi)$. Then $\Phi' \equiv \Phi'' \wedge \gamma(\Phi)$. From the construction we have $|\Phi'| = O(|\Phi|^3)$. Note also that Φ'' expresses the conditions of unitarity, orthonormality among bases, conditions on the density matrix(state). It depends on Φ only in the choice of variables.

We prove next that Φ is satisfiable if and only if Φ' is satisfiable. Suppose Φ' is satisfiable. Then, for each variable x appearing in Φ' there is some complex number $\pi(x)$ such that if the latter are substituted in Φ' then the resulting formula is true. To prove that Φ is satisfiable we have to find bases and unitary matrices connecting them in complex Hilbert spaces of appropriate dimension.

1. Now, we deal exclusively in dimension N. All probability formulas are to be computed in this dimension by homogeneity. We have

assumed that Φ is a "bare" formula, that is, without the measurement operators. Also assume that Φ is atomic. Then, by definition, there is some polynomial relation $p(z_1, \ldots, z_a) \geq 0 (\leq 0)$ such that Φ is obtained by substituting probability terms or transformation matrix terms uniformly for the variables z_i. By assumption Φ' is satisfiable. Thus, there is an interpretation π such that the complex numbers $\pi(x_{ij}), \pi(z_{ij})$ and $\pi(u_{ij}(X,Y)$, satisfy Φ'' *and* $\gamma(p)$. Recalling that $\mathbf{x}_i$ corresponds to the basis variable X_i we define the map $\gamma(X_i) = \mathbf{x}_i$ and $\gamma(m_{ij}(X,Y)) = u_{ij}(X,Y)$. Now extend the interpretation function π as follows. Suppose X, Y denote bases in dimension k. Then $\pi(X_i)$ is the projection operator on to the i^{th} vector in the basis corresponding to X. Instead of the projection operator it is convenient to write the corresponding vector $\alpha \in \mathbb{C}^k$ written as a complex vector. Thus $\pi(X) = \alpha^\dagger \alpha$ is a matrix of order k. Below we simply write the entries of α as $\pi(X_i)_j$. For X an irreducible basis symbol let

$$\pi(X_i) \stackrel{\text{def}}{=} \pi(\gamma(X_i)_j) \equiv \begin{cases} \pi(x_{ij}) \text{ if } j \in L^k \\ = 0 \text{ otherwise} \end{cases} \tag{5.13}$$

$$\pi(m_{ij}(X,Y)) \stackrel{\text{def}}{=} \pi(\gamma(m_{ij}(X,Y))) \equiv \begin{cases} \pi(u_{ij}(X,Y)) \text{ if } i,j \in L^k \\ = \delta_{ij} \text{ otherwise} \end{cases} \tag{5.14}$$

Assuming that $\pi(X)$ and $\pi(Y)$ already defined in dimensions k and k' we extend it to the product basis $t(X,Y)$ by

$$\pi(t(X_i, Y_j)) = \pi(X_i) \otimes \pi(Y_j) \tag{5.15}$$

Similarly, we extend π to the matrices relating the special basis Z^k to all other bases. It is convenient to use matrix notation. Thus we let $M(X,Y)$ be the matrix whose entries are given by $\pi(m_{ij}(X,Y))$. Let $\dim(X) = k$ and $\dim(Y) = k'$ Then,

$$\pi(M(Z^{kk'}, t(X,Y))) = \pi(M(Z^k, X)) \otimes \pi(M(Z^{k'}, Y)) \tag{5.16}$$

Note that, strictly speaking, π does not apply to matrices. The above formula is a compact representation of $\pi(m_{ij}(Z^{kk'}, t(X,Y)))$. The right hand side is the tensor product of matrices. For arbitrary bases X, X' in dimension k

$$\pi(M(X,X')) \equiv (\pi(M(Z^k,X)))^\dagger \pi(M(Z^k,X')) \qquad (5.17)$$

Where $A^\dagger$ is the hermitian(transposed) conjugate of the matrix A. Now we consider the state at which the formulas are evaluated. Define ρ, the density matrix state, as follows. First, let

$$\sum_{i \in L^N} \pi(s_{ii}) = a$$

Then

$$\rho_{ij} = \begin{cases} \pi(s_{ij}) \text{ if } i,j \in L^N \\ \frac{1-a}{N-N'}\delta_{ij} \text{ if } N > N' \text{ and } i \text{ or } j \not\in L^N \\ \delta_{ij} \text{ if } N = N' \end{cases} \qquad (5.18)$$

The above construction and the fact that Φ'' is satisfiable for the interpretation π makes the following facts clear.

(a) First we observe that the above interpretation π actually extends to all the basis and unitary symbols appearing in Φ.

(b) If $Z_{ij} \equiv t(X_i, Y_j)$ is a product basis symbol then

$$\pi(t(X_i,Y_j)) = \pi(X_i) \otimes \pi(Y_j).$$

We show that the interpretation is consistent with that of the tensor product of bases. Let the dimension of the bases X and Y be k and k' respectively. Any component in the product basis $t(X,Y)_m$ is of the form $t(X_i,Y_j)$ where $i = r_1(m)$ and $j = r_0(m)$. Hence, if $m \in R^{kk'}$ then $i \in R^k$ and $j \in R^{k'}$. Since the components of the basis variables $X_i(Y_j)$ are already defined by the equations 5.15 as $\mathbf{x}_{ii'}$(resp. $\mathbf{y}_{jj'}$) for $i' \in R^k$(resp. $j' \in R^{k'}$). But these equations define the basis variables $t(X_i,Y_j)$ as the components of the tensor product of the bases X_i and Y_j.

(c) For X irreducible, $\pi(X_i)$ and $\pi(X_j)$ are unit vectors in $\mathbb{C}^k$(k dimension of X) that are orthogonal for $i \neq j$. This is true as the vectors $\mathbf{x}_i$ and $\mathbf{x}_j$are already orthogonal because of the formula **Bs** and we have only added zeros. Since the tensor product of orthonormal vectors yields orthonormal vectors we see that $\pi(t(X_i, Y_j)))$ and $\pi(t(X_{i'}, Y_{j'}))$ are also orthonormal for products of orthonormal vectors $X_i, X_{i'}$ and $Y_j, X_{j'}$, respectively.

(d) Similarly $\pi(M(X,Y))$ are unitary matrices for the following reason. First, $u_{ij}(Z,X)$ constitute a unitary matrix because of the formulas $\mathbf{Un}_k$. Since the tensor product of two unitary matrices is unitary $\pi(M(Z,t(X,Y)))$ is also unitary.

(e) The matrix ρ is non-negative definite with trace 1 due to the formula $\mathbf{St}_\Phi$.

(f) The formulas **H** express that $\pi(m_{ij}(Z,X))$ are indeed the entries of the unitary matrix connecting the bases $\pi(Z)$ and $\pi(X)$ for X irreducible. Moreover, similar formulas hold for the product bases $t(X,Y)$. This follows from the fact that $\pi(Z^{kk'} = \pi(Z^k) \otimes \pi(Z^{k'}))$ by definition and the fact that in any interpretation $\pi(M(t(Z^k, Z^{k'}, t(X^k, Y^{k'}))) = \pi(M(Z^k, X)) \otimes \pi(M(Z^{k'}, Y))$. For arbitrary bases X and X' in the same dimension k we have the correct transformation relation due to the equations 5.17.

Note that the completion of the matrices by 0's in the off-diagonal terms and 1's in the diagonal ensures that the formulas $\mathbf{Un}_k, \mathbf{G}, \mathbf{H}, \mathbf{St}_\Phi$ continue to hold, even for the indices not in some L^k. This is most easily seen if we imagine that the matrices are initially defined for the first $k' \leq k$ rows and columns and then extended by putting 1's in the diagonal and 0's elsewhere to complete it to a matrix of order k. In

the block partitioned form it appears as follows.

$$\left(\begin{array}{ccc:ccc} & & & 0 & \cdots & 0 \\ & A & & \vdots & \vdots & \vdots \\ & & & 0 & \cdots & 0 \\ \hdashline \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \hdashline 0 & \cdots & 0 & 1 & 0 & \cdots \\ \vdots & \vdots & \vdots & 0 & \ddots & \cdots \\ 0 & \cdots & 0 & 0 & \cdots & 1 \end{array}\right)$$

Here A is the matrix originally constructed using the indices provided in Φ. Actually, the indices will not have the ordering implied in the above picture. However, a routine but tedious argument shows that the above statements are correct. In general, if we call the extended matrix of order k, A' then the matrix A is a *principal* submatrix of A. A principal submatrix is obtained by deleting some rows with indices from a given set and also deleting columns with the same indices [MM92]. The concept of the principal submatrix is important for the proof below.

Finally, we consider the interpretation of probability terms. Let X be a basis symbol in dimension N and that $P(X_i)$ appears in Φ. Then, the interpretation of $P(X_i)$ in the state ρ is given by (see 5.1.3)

$$\pi_\rho(P(X_i)) = \sum_{j,k=0}^{N-1} \overline{x}_{ij} s_{jk} x_{ik}$$

Now by our definition $\pi(X_i)_j = 0$ for all $j \notin L_\Phi^N$. Hence, we are only left with the terms whose indices are in L_Φ^N. But this is precisely $\sigma(P(X_i))$. For a general probability term, $P(\alpha)$ where α is a basis formula we reduce it to the atomic case by the equations 5.11 and 5.12. We see that the maps σ and γ give the correct interpretation for probability and matrix terms respectively. Therefore, if Φ' is satisfiable then so is Φ. We observe that any interpretation π of the variables

in Φ' can be extended to an interpretation of the corresponding basis and matrix terms of Φ and that the state ρ can be defined from the interpretation. Moreover, if Φ' evaluates to true for the interpretation π then the extension of π makes Φ true at the state ρ. In particular if Φ' is valid(true for any interpretation) then so is Φ.

Next we prove the converse. That is, if an atomic formula Φ is satisfiable then so is Φ'. This part is somewhat more involved. Since Φ is satisfiable we must have an interpretation π of the basis variables and the matrix terms such that Φ is satisfiable. We define the extension of π to an interpretation of the variables introduced in the construction of Φ'. We continue to call this extension π since these variables do not appear in Φ. We let

$$\pi(\mathbf{z}_i^k) = \epsilon_i$$

where ϵ_i is the i^{th} vector in the standard basis in dimension k. Now fix a dimension k appearing in the formula. Let $X^1, \ldots, X^{i_k}$ be the irreducible basis symbols that appear in Φ. The vectors $\pi(X_j^i)$ the jth vector in the basis X^i has dimension k. We want to construct a corresponding vector $\mathbf{x}_j^i$ in dimension $r_k \leq k$ and an extension of π to interpret $\mathbf{x}_i^j$ with the following properties.

(a) $\pi(\mathbf{x}_i^j)$ are unit vector of $\mathbb{C}^{r_k}$ and

$$\pi(x_{ir}^j) = \pi(X_i^j)_r, r \in R^k \text{ and}$$
$$\langle \pi(\mathbf{x}_i^j) | \pi(\mathbf{x}_k^j) \rangle = \delta_{ik}$$

Here we use the standard inner product in $\mathbb{C}^{r_k}$. The above relations express the fact that r_k-dimensional vectors $\{\mathbf{x}_0^j, \ldots, \mathbf{x}_{r_k-1}^j\}$ form an orthonormal basis *and* for the indices which lie in R^k the interpretation is same as the given one for the corresponding basis variable in Φ. The reader may recall that R^k is the set of indices in dimension k which appear in Φ.

(b)

$$\langle \pi(\mathbf{x}_i^k) | \pi(\mathbf{x}_j^l) \rangle = \pi(m_{ij}(X^k, X^l)), i, j \in R^k$$

This is the crucial condition. It states that the relation between the bases X, Y is given by the interpretation π for the matrix symbols $m_{ij}(X, Y)$ *when* the indices $i, j \in R^k$.

(c) Once we have interpreted the irreducible vectors the tensor product of any of these vectors satisfies appropriate relation by the construction.

(d) Finally, we have a density matrix ρ' in $\mathbb{C}^{r_k}$ such that

$$s_{ij} = \rho'_{ij} \text{ and } \pi(P(X_j^i)) = \sum_{lm \in L^k} \rho'^{lm} \pi(\overline{x}_{ilx_{im}})$$

That is, for the indices appear in in Φ the "probabilities" in $\mathbb{C}^{r_k}$ corresponding to the basis vectors $\mathbf{x}_j^i$ are identical to those of X_j^i.

From the above construction it is clear that the formula Φ' is satisfiable when Φ is. The fact that such a construction is possible is a consequence of the Lemma 16 which follows. Since we have shown satisfiability by extending the given interpretation π of Φ, if Φ is valid (that is, true for *any* π) then so must be Φ'.

The case when Φ is not an atomic formula follows easily by induction. If Φ is of the form $\Phi_1 \vee \Phi_2$, let Φ' be $\Phi'_1 \vee \Phi'_2 \equiv (\Phi''_1 \wedge \gamma(\Phi_1)) \vee (\Phi''_2 \wedge \gamma(\Phi_2))$. If Φ' is satisfiable then at least one of the formulas Φ'_1 or Φ'_2 must be satisfiable. Hence, Φ_1 or Φ_2 must be satisfiable by the induction hypothesis. Similarly, if $\Phi = \neg\Phi_1$ then let $\Phi' = \Phi''_1 \wedge \neg\gamma(\Phi_1)$. Now, if Φ is not satisfiable then Φ_1 must be valid and conversely. Consequently, $\neg\Phi_1$ is satisfiable iff $\Phi''_1 \wedge \neg\gamma(\Phi_1)$ is satisfiable. If $\Phi = \exists x \Phi_1$ then there is some interpretation π such that Φ_1 is true for π, which includes an interpretation of x as well. Let $\Phi \equiv M_X(\Phi_1)$, where X is some basis term (irreducible or composite). Suppose, Φ'_1 has been constructed. Write it as $\Phi'_1[\rho]$, making the dependency on the

collection of **RC** terms representing the quantum state, explicit. Then Φ' is $\Phi'_1[\rho']$ where

$$\rho'(jk) = \sum_i p_i \gamma X_{ij} \gamma X_{ik} \text{ where}$$
$$p_i = \sum_{jk} \gamma(X_i)_j \rho(jk) \gamma(X_i)_k.$$

The p_i are the probabilities of the basis term X_i in state ρ. The second formula is precisely the trace formula for probability. The first formula is the post-measurement distribution of the states. The terms $\sigma X_{ij} \sigma X_{ik}$ are the components of the projection operator corresponding to the state $\sigma(X_i)$.

If Φ is $S_{X_i}\Phi_1$ then, Φ' is $\Phi'_1[\rho']$ with

$$\rho'(jk) = \gamma X_{ij} \gamma X_{ik}$$

From the above construction and the semantics of measurement operators it is clear that Φ is satisfiable iff Φ' is satisfiable. □

Lemma 16 *Let* $B_1 = \{\alpha_1^1, \ldots, \alpha_{a_1}^1\}, B_2 = \{\alpha_1^2, \ldots, \alpha_{a_2}^2\}, \ldots, B_k = \{\alpha_1^k, \ldots, \alpha_{a_k}^k\}$ *be vectors of unit length in* $\mathbb{C}^n$. *Suppose we are given the following data.*

1. *In each of the sets* B_i *distinct vectors are orthogonal. This implies, in particular that* $a \leq n$.

2. $$\langle \alpha_j^i | \alpha_l^r \rangle = c_{jl}^{ik},\ i \neq r,\ 1 \leq i, r \leq k,\ 1 \leq j \leq a_i \text{ and } 1 \leq l \leq a_r$$

3. *A positive semidefinite matrix* ρ *such that*

$$\mathtt{Tr}(|\alpha_j^i\rangle\langle\alpha_j^i| \, \rho) = p_j^i$$

Let $m = \min\{(a = a_1 + a_2 + \cdots + a_k), n\}$. *Then there are* k *bases* $B_i' = \{\beta_j^i | 1 \leq j \leq m\}$, $i = 1, \ldots, k$ *in* $\mathbb{C}^m$ *and a positive semidefinite matrix* ρ' *of order* m *such the*

$$\langle \beta_j^i | \beta_r^l \rangle = c_{lr}^{ij},\ i \neq j,\ \text{and } j \leq a_i \text{ and } r \leq a_l$$
$$\mathtt{Tr}(|\beta_j^i\rangle\langle\beta_j^i| \rho') = p_j^i,\ j \leq r$$

Proof: First we note that if $m = n$ then we simply take $B_i' = B_i$. We therefore assume that $m = a < n$. The subspace spanned S spanned by the vectors α_j^i has dimension at most a. Let $\gamma_1, \ldots, \gamma_k$ be an orthonormal basis in S and let

$$\alpha_j^i = \sum_{r=1}^{m} x_{jr}^i \gamma_r$$

We now define vectors $\beta_j^i \in \mathbb{C}^m$ by

$$\beta_{jr}^i = x_{jr}^i \tag{5.19}$$

Then

$$\langle \alpha_j^i | \alpha_r^l \rangle = \langle \sum_{s=1}^{m} x_{js}^i \gamma_s | \sum_{t=1}^{m} x_{rt}^l \gamma_t \rangle = \sum_s \overline{x_{js}^i} x_{rs}^l = \langle \beta_j^i | \beta_r^l \rangle$$

In particular β_j^i, and β_r^i are orthogonal for $j \neq r$. Hence they can be extended to a basis. The first assertion of the lemma is proved.

To prove the second observe that any positive definite operator on $\mathbb{C}^n$ may be written as

$$\sum_i p_i |\psi_i\rangle\langle\psi_i|\ ,\ p_i \geq 0 \text{ and } \sum_i p_i = 1$$

The operators $|\psi_i\rangle\langle\psi_i|$ correspond to pure states. Hence, if we can find for any unit vector $\psi \in \mathbb{C}^n$ a vector $\psi' \in \mathbb{C}^m$ such that

$$\mathtt{Tr}(|\psi\rangle\langle\psi|\ |\alpha_j^i\rangle\langle\alpha_j^i|) = \langle\psi|\alpha_j^i\rangle^2 = \langle\psi'|\beta_j^i\rangle^2$$

then the lemma would be proved. Write $\psi = \psi_1 + \psi_2$ where $\psi_1 \in S$ and $\psi_2 \in S^{\perp}$, the subspace normal to S. Then, $\langle\alpha_j^i|\psi\rangle = \langle\alpha_j^i|\psi_1\rangle$. Since, $\psi_1 \in S$,

it can be written as linear combination of γ_i. Thus if $\psi = \sum_i y_i\gamma_i$ let ψ' be the column vectors with y_i. We then have, $\langle\alpha_j^i|\psi\rangle = \langle\beta_j^i|\psi'\rangle$ The lemma is proved.

□

The construction given in Theorem 12 provides an algorithm for reducing a formula of $\mathcal{L}(P, \mathbf{m}, t, M, S)$ to a formula of $\mathbb{RC}$. Hence using the results of [BKR86] and [Can88] we deduce, as in the last chapter, the following result.

Theorem 13 *A formula of $\mathcal{L}(P, \mathbf{m}, t, M, S)$ can be decided in exponential space. If it is an existential formula then it can be decided in polynomial space.*

We note that the two preceding theorems hold for general formulas of quantum theory, not just quantum circuits. However, in quantum computation one builds circuits using unitary operators(gates) of some fixed sizes. Then it is easy to see that the length of the $\mathbb{RC}$-formula Φ' of Theorem 12 is $O(|\Phi|^2)$. Moreover, it is also clear that, in this case the *time* complexity is $O(|\Phi|)$. Hence, the Grover algorithm which is $O(\sqrt{N})$ using quantum circuits can be classically simulated in time $O(N)$. The complexity result is in terms of the length of the formula. In general, the length of the formula may be of exponential order in the number of "qubits", or the atomic bases. This is the case in case of the Shor algorithm which we discuss in the next chapter. We next prove completeness of the axiomatization of $\mathcal{L}(P, \mathbf{m}, t, M, S)$.

Theorem 14 *The theory* $\mathbf{Ax}(P,\mathbf{m}, t, M, S)$ *is sound and complete.*

Proof: The axioms of $\mathcal{L}(P, \mathbf{m}, t, M, S)$ are valid for any interpretation. This is a straightforward but tedious verification. We have discussed this in the remarks following the axioms. It is also the case that the inference rules are validity preserving in any model. The proof of completeness is more involved but it is similar to the fixed-dimensional case dealt with in the

last chapter(see lemmas 10, 11, and 12). Consider first a "bare" formula(without measurement operators) Φ. Recall that it is homogeneous, that is, all probability terms refer to a *fixed* dimension. Hence, as in Lemma 12 we can construct **RC**-formulas Φ'' and $\tilde{\Phi}$ such that Φ is satisfiable iff $\Phi'' \wedge \tilde{\Phi}$ is satisfiable. We recall that construction below.

Given a formula Φ of $\mathcal{L}(P, \mathbf{m}, t, M, S)$ we construct a formula $\tilde{\Phi}$ as in Lemma 11. The construction is similar to the one given in Theorem 12. We do not assume any ordering of the atomic basis variables. Moreover, since we are only interested here in proving completeness we ignore complexity issues and give the construction in the full Hilbert space. Let Φ contain basis formulas built from basis expressions $X^1, X^2, \ldots, X^k$ of which the first s are assumed to be bases appearing in the probability terms. Hence, they have dimension N. By assumption, the formula Φ is homogeneous in dimension N. It may contain basis terms of some dimension k dividing N. Now imitating the construction in Theorem 12 we construct a formula Ψ as follows. Let $p_1 < \ldots < p_m$ be the primes dividing N. Let $\mathtt{b}^1, \ldots, \mathtt{b}^m$ be *new* basis symbols such that $\mathtt{b}^i$ has dimension p_i. For each, irreducible basis expression X of dimension a in Φ, let $\mathbf{U}_X$ be the formula

$$\mathbf{U}_X = \bigwedge_{i,j} m_{ij}(t(t^{i_1}(\mathtt{b}^1), \ldots, t^{i_m}(\mathtt{b}^m)), X) = x_{ij}$$
$$\text{where } a = p_i^{i_1} \cdots p_m^{i_m}$$

Let us use the matrix notation for concise representation. Thus, we let U_X denote the above matrix whose entries are given by $U_X(ij) = m_{ij}(Z^a, X)$, where $Z^a \equiv t(t^{i_1}(\mathtt{b}^1), \ldots, t^{i_m}(\mathtt{b}^m))$. We then let Ψ_1 be the formula which is the conjunction of all $\mathbf{U}_X, X \in \Phi$. It is easy to see that Φ is satisfiable if and only if $\Phi \wedge \Psi_1$ is. For, if Φ is satisfiable choose an interpretation π which

satisfies it and choose bases $\mathtt{b}^1, \mathtt{b}^2, \ldots$ which are different from the bases appearing in the interpretation. We continue to call the extended interpretation π. Then

$$\pi(Z^a) = (\pi(\mathtt{b}^i)^{\otimes i_1}) \otimes \cdots \otimes (\pi(\mathtt{b}^m)^{\otimes i_m}) \text{ and} \tag{5.20}$$

$$m_{ij}(Z^a, X) = \langle \pi(Z_i^a) | \pi(X_j) \rangle \tag{5.21}$$

satisfies Ψ_1. The introduction of these distinguished bases makes the bookkeeping easier. We may therefore reason with $\Phi \wedge \Psi_1$ in place of Φ. We continue to call the former formula Φ.

For each such basis $\mathtt{b}^i$ appearing in Φ we introduce vector variables $\mathbf{x}_j^i$ for the basis components $\mathtt{b}_j^i$. Recall that the vector variable $\mathbf{x}_j^i$ is a shorthand for a collection of variables x_{jk}^i, $j, k \in \{0, \ldots, p_i - 1\}$. We denote the collection of basis components $\{\mathbf{x}_j^i\}$ by $\mathbf{x}^i$. For convenience, we suppress the dimensions in the summation and other indexed formulas. Now each dimension M corresponding to some basis variable appearing in Φ, is of the form $p_1^{j_1} \cdots p_r^{j_r}$. Let us denote the corresponding tensor product basis by

$$(\mathbf{x}^1)^{\otimes j_1} \otimes (\mathbf{x}^2)^{\otimes j_2} \otimes \cdots \otimes (\mathbf{x}^r)^{\otimes j_r}$$

Here $(\mathbf{x}^1)^{\otimes j_1}$ is the basis consisting of all j_1-fold tensor product of vectors $\mathbf{x}_j^i$. We recall the definition of tensor product of two vectors $\mathbf{x} \in \mathbb{C}^m$ and $\mathbf{y} \in \mathbb{C}^n$.

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \quad \mathbf{x} \otimes \mathbf{y} = \begin{pmatrix} x_1 y_1 \\ x_1 y_2 \\ \vdots \\ x_m y_n \end{pmatrix}$$

Note that associativity of the tensor product is implicit. We write the special bases $(\mathbf{x}^1)^{\otimes j_1} \otimes (\mathbf{x}^2)^{\otimes j_2} \otimes \cdots \otimes (\mathbf{x}^r)^{\otimes j_r}$ introduced above as $\mathtt{X}(j_1, \ldots, j_r)$. Moreover, since the integers $m_1, \ldots, m_r$ are assumed to be relatively prime, the integers n_i, the dimension of the basis variables appearing in Φ are uniquely expressed as product of the m_i's. Thus we simply write $\mathtt{X}(n_i)$. Note that $\mathtt{X}(j_1, \ldots, j_r)$ is a collection of "vector" $\mathbb{RC}$ terms in a particular

order. Thus, the first one is the vector term

$$\overbrace{(\mathbf{x}_0^1 \otimes \cdots \otimes \mathbf{x}_0^1)}^{j_1 \text{ times}} \otimes \cdots \otimes \overbrace{(\mathbf{x}_0^r \otimes \cdots \otimes \mathbf{x}_0^r)}^{j_r \text{ times}}$$

Recall that $N = p_1^{i_1} \cdots p_r^{i_r}$. Let $0 \leq a \leq N-1$ be an integer. Let $p'_j = N/p_j^{i_j}$. Then, p_i and p'_i are relatively prime. Write $a \bmod p'_1$ in m_1 basis as $a^1_{t_1} mp1^{t_1} + \cdots + a^1_1 p_1 + a^1_0$. Now replace N by $p'_1 = p_2^{i_2} \cdots p_r^{i_r}$, p_1 by p_2 and continue as above. We get a sequence of positive integers $\{a^1_0, \ldots, a^1_{t_1}, a^2_0, \ldots, a^2_{t_2}, \ldots, a^r_0, \ldots, a^r_{t_r}\}$. This is the representation of a in the multibasis $\{p_1, \ldots, p_r\}$. Let $\mathtt{X}^a(N) = \mathbf{x}^1_{a^1_0} \otimes \cdots \mathbf{x}^1_{a^1_{t_1}} \otimes \cdots \otimes \mathbf{x}^r_{a^r_0} \otimes \mathbf{x}^1_{a^r_{t_r}}$. Intuitively, the ordered collection $\mathtt{X}^a(N)$ constitute the (tensor) product basis in $\mathbb{C}^N$. The intuition that the collection of vector variables $\mathbf{x}^i$ constitute a basis is expressed by the following formula

$$\mathbf{Bs} \equiv \bigwedge_{ij} \sum_l \overline{x^i_{jl}} x^i_{ml} = \delta_{jm} \tag{5.22}$$

The formula expresses that for each i the set of vectors $\mathbf{x}^i_j$ constitute a basis in $\mathbb{C}^{p_1}$. Then the "tensor terms" $\mathtt{X}(j_1, \ldots, j_r)$ constitute a basis.

Next, let $J \subset \{1, \ldots, k\}$ be the set of indices such that X^i is irreducible. For each basis symbol X^i, $i \in J$ of dimension n_i appearing in Φ introduce $\mathbb{RC}$-variables y^i_{jl}, $0 \leq j, l \leq n_i - 1$. Intuitively, y^i_{jk} represent the unitary matrix that transforms the basis $\mathtt{X}(n_i)$ to the basis represented by X^i. We assume that the n_i's are ordered so that $n_1 \leq n_2 \leq \cdots \leq n_k = N$. We write the variables y^i_{jl} collectively as $\mathtt{Y}^i$. Let

$$\mathbf{Un}(\mathtt{Y}^i) \equiv \bigwedge_{lm} \sum_j \overline{y^i_{lj}} y^i_{mj} = \delta_{lm} \tag{5.23}$$

be the familiar unitarity conditions. Next introduce N^2 variables $\rho = \{\rho_{ij},\ 0 \leq i, j \leq N-1\}$ corresponding to the state(density matrix) and the formula

$$\mathbf{St} \equiv \sum_i \rho_{ii} = 1 \wedge \bigwedge_{ij} \forall x_i x_j (\overline{x_i} \rho_{ij} x_j) \geq 0 \tag{5.24}$$

Now for each vector variable $\mathrm{X}_j(N)$ $0 \leq j \leq N-1$ we introduce "probability" variables p_j and the formula

$$\mathbf{Prob(p)} \equiv \bigwedge_j (p_j \geq 0) \wedge \sum_j p_j = 1 \tag{5.25}$$

We consider probability variables in N-dimension only because by assumption Φ is homogeneous and all probability formulas pertain to this dimension. Besides the probability variables for the distinguished variables we introduce variables q_j^i for the probability term $P(X_j^i)$ appearing in probability formulas. Let

$$\begin{aligned} \Psi \equiv & \bigwedge_i (\mathbf{Prob}(\mathrm{q}^i) \wedge \mathbf{Un}(\mathrm{Y}^i)) \wedge (\mathbf{Prob(p)} \wedge \mathbf{Bs} \wedge \mathbf{St}) \wedge \\ & \bigwedge_j (p_j = \rho_{jj}) \bigwedge_{ij} (q_j^i = \sum_{kl} y_{jk}^i \rho_{kl} \overline{y_{jl}^i}) \end{aligned} \tag{5.26}$$

Now introduce new basis symbols d^j for each $j \in \{n_1, n_2, \ldots, n_k\}$. Intuitively, we are introducing product bases for each of the dimensions appearing in Φ. Let $\tilde{\Phi}$ be the $\mathbb{RC}$-formula constructed as follows. We now introduce $\mathbb{RC}$ variables z_{ji}^i, $0 \leq j, l \leq n_i$ for each of the basis expression X^i appearing in Φ. $\dim(X^i) = n_i$. If X^i is irreducible then let

$$\mathbf{F}(X^i) \equiv \bigwedge_{jl} z_{jl}^i = y_{jl}^i$$

Otherwise, X^i is of the form $t(X^{i_1}, X^{i_2})$. Then, $n_i = n_{i_1} n_{i_2}$. Define,

$$\mathbf{F}(X^i) \equiv \bigwedge_{j_1 j_0, l_1 l_0} z_{j_1 j_0, l_1 l_0}^i = z_{j_1 l_1}^{i_1} z_{j_0 l_0}^{i_2}$$

Note that in the above formula it is implicit that each index $j(l)$ is decomposed as $j_1 j_0 (l_1 l_0)$ where $j_1 = r_1(j) = \lfloor j/n_{i_2} \rfloor$ and $j_2 = j \bmod n_{i_2}$. Let $\tilde{\Phi}$ be the conjunction of Ψ and all $\mathbf{F}(X^i)$ constructed above. Let $\tilde{\Phi}\theta$ be the formula of $\mathcal{L}(P, \mathbf{m}, t, M, S)$ obtained from $\tilde{\Phi}$ by the following substitutions.

1. For p_j substitute $P(\mathrm{d}_j^N)$ and for q_j^i substitute $P(X_j^i)$. Note that X^i must be basis in dimension N.

2. Substitute $m_{jl}(\mathrm{d}^i, X^i)$ for z^i_{jl} and also put $m_{jl}(\mathrm{d}^s, X^s)$ for y^s_{jl} if $s \in J$.

The first observation is that $\tilde{\Phi}\theta$ is provable in **Ax**(P,**m**, t, M, S). The proof is similar to Lemma 11. We start with the subformula of $\tilde{\Phi}\theta$ pertaining to dimension N. This is a consequence of probability axioms, unitarity axioms and $\mathbf{MP}'_k$. In fact, a similar assertion was proved in the lemma mentioned. The extra complication is the possible presence of matrix terms in dimensions less than N. If X^i is irreducible then we have instances of identity axiom $m_{jl}(\mathrm{d}^i, X^i) = m_{jl}(\mathrm{d}^i, X^i)$ since the same substitution occurs for y^i_{jl} and z^i_{jl}. If X^i is of the form $t(X^{i_1}, X^{i_0})$ then $n_i = n_{i_1} n_{i_0}$ and the corresponding subformula of $\tilde{\Phi}\theta$ is

$$\begin{aligned} m_{jl}(\mathrm{d}^i, t(X^{i_1}, X^{i_0})) &= m_{jl}(t(\mathrm{d}^{i_1}, \mathrm{d}^{i_0}), t(X^{i_1}, X^{i_0})) \\ &= m_{r_1(j),r_1(l)}(\mathrm{d}^{i_1}, X^{i_1}) m_{r_0(j),r_0(l)}(\mathrm{d}^{i_0}, X^{i_0}) \end{aligned}$$

But this is an instance of **Tensor 3**.

Next, we consider the formula Φ. By assumption it is without measurement operators. Hence, it is obtained by substituting probability terms and matrix terms for some variables in an $\mathbb{RC}$-formula **G**. We construct from Φ another $\mathbb{RC}$ formula Φ'' as follows.

1. We assume the basis formulas are in canonical form. For probability terms $P(X^s_{i_1} \vee X^s_{i_2} \vee \cdots \vee X^s_{i_m}$ substitute in Φ, the term $(q^s_{i_1} + \cdots + q^s_{i_m})$, keeping in mind that $\dim(X^s) = N$.

2. For matrix terms $m_{jl}(X^a, X^b)$ we substitute the following

$$m_{lm}(X^i, X^j) \leftrightarrow \sum_s \overline{z^i_{sl}} z^j_{sm}$$

The intuition behind the above formula has been explained before. If $Z^i = (z^i_{lm})$ represents the matrix taking the basis d^i to X^i and similarly Z^j the matrix relating d^i to X^j(since $n_i = n_j$, by construction d^i is identical to d^j) then, $Z^{i\dagger} Z^j$ is the transformation matrix relating X^i and X^j.

We can now show that Φ is satisfiable iff $\tilde{\Phi} \wedge \Phi''$ is satisfiable. This is similar to the analysis in Theorem 12. I omit the details. We may now use arguments almost identical to those in Theorem 9 to show that Φ is satisfiable iff it is consistent.

Now consider first formulas with a single measurement operator. Assume that $M_{\mathfrak{b}}(\Phi)$ is consistent. Then let X^i be the bases in the formula including product bases. If Φ is not satisfiable then, since it is a bare formula we must have $\vdash \neg\Phi$. Hence from **Measure3**, $\vdash \neg M_{\mathfrak{b}}\Phi$. The formula $M_{\mathfrak{b}}\Phi$ is inconsistent, a contradiction. Hence we may assume that Φ is consistent. Let $X^1, \ldots, X^k$ be the basis terms appearing in Φ. Now, let x_i be new real variables not appearing in Φ. Let $\Gamma \equiv \wedge_i P(\mathfrak{b}_i) = x_i$. Then, since clearly Γ is consistent so is the formula $\Phi \wedge \Gamma$. Suppose, $M_{\mathfrak{b}}(\Phi)$ is not satisfiable. Then $M_{\mathfrak{b}}(\Phi) \wedge \Gamma$ is also unsatisfiable. By assumption, $M_{\mathfrak{b}}(\Phi)$ is consistent. Then, $M_{\mathfrak{b}}(\Phi) \wedge \Gamma$ is also consistent. From, **Measure1** it follows that

$$\vdash M_{\mathfrak{b}}(\Phi) \wedge \Gamma \Rightarrow M_{\mathfrak{b}}(\Phi) \bigwedge (\bigwedge_{ij} (M_{\mathfrak{b}}(P(X_i^j) = \sum_k T(X_i^j, \mathfrak{b}_k) x_k))).$$

Since by hypotheses the left hand side of the implication sign is consistent, so is the right side. By **Measure2** and **Measure3** the right hand side is $M_{\mathfrak{b}}(\Phi \wedge (\wedge_{ij} P(X_i^j) = \sum_k T(X_i^j, \mathfrak{b}_k) x_k))$ and it is unsatisfiable since $M_{\mathfrak{b}}(\Phi)$ is. Arguing as above we may assume that the bare formula $\Phi \wedge_{ij} ((P(X_i^j) = \sum_k T(X_i^j, \mathfrak{b}_k) x_k))$ is consistent . Hence it is satisfiable. Write $\sum_k T(X_i^j, \mathfrak{b}_k) x_k = A_i^j$. Let

$$\pi, \rho \models \Phi \wedge (\wedge_{ij} ((P(X_i^j) = \sum_k T(X_i^j, \mathfrak{b}_k) x_k)))$$

Let $\Phi' = \Phi[P(X_i^j)/A_i^j]$, that is, the formula obtained from Φ by the substitution of the terms A_i^j for $P(X_i^j)$. Then from the substitution rule of equality

$$\pi, \rho \models \Phi'$$

Recall the semantics of measurement operator(see 5.2). Thus,

$$\pi, \rho \models M_{\mathfrak{b}}(\Phi) \text{ iff } \pi, \sum_k \pi(\mathfrak{b}_k) \rho \pi(\mathfrak{b}_k) \models \Phi$$

Let $\pi(\mathtt{b}_k) = |\alpha_k\rangle\langle\alpha_k|$ and $\rho' \equiv \pi(\mathtt{b}_k)\rho\pi(\mathtt{b}_k) = \sum_k |\alpha_k\rangle \langle\alpha_k|\rho|\alpha_k\rangle \langle\alpha_k| = \sum_k \pi_\rho((P(\mathtt{b}_k)))\, |\alpha_k\rangle\langle\alpha_k|$. Then, $\pi_{\rho'}(P(X_k^j)) = \sum_k \pi_\rho((P(\mathtt{b}_k)))|\langle\alpha_k|\pi(X_i^j)\rangle|^2 = \sum_k \pi_\rho((P(\mathtt{b}_k)))T(X_i^j, \mathtt{b}^k)$. This means that, for a given interpretation of π of basis terms and all other variables, $\pi, \rho \models M_{\mathtt{b}}(\Phi)$ iff $\pi, \rho' \models \Phi$ iff $\pi, \rho \models \Phi'$. But we have just observed that $\pi, \rho \models \Phi'$. Hence $M_{\mathtt{b}}(\Phi)$ is satisfiable.

A similar proof can be given for the operator for selective measurements.

In the final part of the above proof, where it was shown that formulas of the form $M_{\mathtt{b}}(\Phi)$ are satisfiable if consistent it was assumed that Φ is a bare formula. But from the proof it is easily seen that with hardly any modification satisfiability can be proved for consistent formulas with arbitrary number of measurement operators. Further, instead of a maximal test where all the N(=dimension) outcomes are expected, we can generalise to non-maximal tests(see Subsection 5.1.1 for the definitions).

The proof of the theorem is complete. □

In this chapter we developed the semantics and axiomatization of $\mathcal{L}(P, \mathbf{m}, t, M, S)$, a language which incorporates all the fundamental constructs of quantum theory. In the next chapter we introduce a related language which is more intuitive for expressing important quantum algorithms and protocols. The language $\mathcal{L}(P, \mathbf{m}, t, M, S)$ appears to be richer and we conjecture that it is possible to express *any* assertion of quantum theory of a given finite dimensional system.

Chapter 6

Applications

This chapter is on applications. In the last chapter some simple examples were presented. To deal with complex circuits and protocols we introduce a new language $\mathcal{L}(P, t, M, S, \mathbf{U})$ to confirm to the circuit model of quantum computation. This language is interpreted in the language $\mathcal{L}(P, \mathbf{m}, t, M, S)$ of last chapter. Actually, we modify $\mathcal{L}(P, \mathbf{m}, t, M, S)$ slightly by changing notation. Next, we use the language $\mathcal{L}(P, t, M, S, \mathbf{U})$ to express formulas for quantum circuits and algorithms. We give a general algorithm for quantum circuits including its probabilistic 'output'. In the final section we present formulas for the three important milestones of quantum computing and information: Grover search algorithm, phase estimation algorithm (Shor's algorithm), and quantum teleportation. We also present a formula that asserts the *existence* of a Grover circuit. Therefore, in principle, we could decide questions of existence of circuits with given specifications, at least in small dimensions. This is the first step toward *design* of new quantum algorithms.

6.1 An alternative formulation

In this section some extensions to the logics presented in the last chapter is given. The reasons for such extensions are as follow.

1. It is more convenient to use alternative logics for dealing with differ-

ent classes of problems. These would correspond to fragments of the extended logic.

2. When we deal with axiom systems and corresponding theories the extensions turns out to be conservative [Sho67]. Hence, many properties of the fragments follow once we prove them for the larger theory. It saves the labour of separate proofs.

3. The fragments roughly correspond to alternative pictures of quantum theory.

First, I define everything in a larger language $\mathcal{L}(P, t, \mathbf{m}, \mathbf{U}, M)$. Then I consider different fragments of the theory. Semantically, these fragments correspond to equivalent physical pictures. In the Chapter 4 I had described an equivalent alternative formulation of the logic that corresponds to the Schroedinger picture [Per95](see Section 4.5), as opposed to the Heisenberg picture that is implicit in the formulation so far. Briefly, the former views the state as dynamically changing and the bases/coordinate systems fixed while in the Heisenberg picture the state is fixed and the coordinate systems carry the dynamics. In either case the change is mediated through unitary transformations. The Schroedinger view is more popular amongst the quantum computing community although elegant formulation of models of quantum compute using the Heisenberg picture has been considered before[Got99]. Throughout this section we assume that there is exactly one atomic basis symbol $\mathfrak{b}$ of some arbitrary but fixed dimension m. The other basis terms that may appear in the formulas are generated from this basis symbol via the tensor operator. The quantum systems we wish to describe by this restricted formalism are the ones we encounter in quantum computing and information. For example, a system with many *qubits* is described by an atomic basis symbol in dimension 2.

Recall from 4.5 the definition of equivalent bases:

$$\mathsf{b} \sim \mathsf{c} \Leftrightarrow \bigwedge_{i,j=0}^{n-1} m_{ij}(\mathsf{b}, \mathsf{c}) = \delta_{ij}$$

and the fact that in a formula, basis terms may be uniformly replaced by the corresponding term in an equivalent basis. We adopt the notation of Section 4.5: let U, V, W, U' etc. (possibly with subscripts) denote new symbols representing unitary matrices of arbitrary dimension. Sometimes, the dimension is explicit as $U^{(n)}$. Thus associated with each unitary symbol U in dimension n there are n^2 variables of $\mathbb{RC}$ written as $U(ij)$, $i, j = 0, \ldots, n-1$. In the new notation, for every unitary variable U and basis term b in dimension n, $U\mathsf{b}$ will denote a basis term. The intuition is that, since we are considering only orthonormal bases, they will always be connected by a unitary matrix. Thus, the interpretation of UX is that

$$\pi, \rho \models \bigwedge_{ij} m_{ij}(\mathsf{b}, U\mathsf{b}) = U(ij). \tag{6.1}$$

The components of UX will be denoted by $(UX)_i$. Note that, it is easy to translate into the language $\mathcal{L}(P, \mathbf{m}, t, M, S)$ by simply treating UX as a new basis symbol and adding the formula 6.1. Since we have one atomic basis symbol b all other basis terms built from tensor operations on this *single* basis and its images under unitary operators. Thus, b and its components are such terms and if X is such a term then so are $t(\mathsf{b}, X)$ and $t(X, \mathsf{b})$. All product basis terms are of this type. A general basis term is either a product term as above or constructed from another basis term Y by applying some unitary symbol U of appropriate dimension and is denoted by UY. The interpretation of UY is exactly as above. The number of t-operators appearing in a basis term is called the t-degree of the term.

To summarize, we have three kinds of symbols representing bases. The basis symbols and terms are defined recursively:

1. b is a basis symbol and b_i, $i = 0, \ldots, m-1$ are basis terms.

2. If X and Y are basis symbols and U is a unitary symbol of appropriate dimension then $t(X,Y)$ and UX are basis symbols. The corresponding terms are written as $t(X,Y)_i$ and $(UX)_i$.

The dimensions and the basis components are unambiguously defined as before. For example, $U\mathtt{b}_i$ is to be interpreted as $|(U\pi(\mathtt{b})_i)\rangle\langle(U\pi(\mathtt{b})_i)|$, remembering that if $\pi(\mathtt{b}) = \{\psi_0,\ldots,\psi_{n-1}\}$ then $U\pi(\mathtt{b}) = \{\psi'_0,\ldots,\psi'_{n-1}\}$ with $\psi'_k = \sum_j U(kj)\psi_k$. The important point to keep in mind is that we have only one primitive basis symbol $\mathtt{b}$ in the modified language. The probability terms are defined over basis terms constructed from this symbol. Again, we consider homogeneous formulas, that is, formulas in which all basis terms have the same t-degree. A general formula is obtained by replacing some of the variables in a multivariate polynomials with probability terms. This is exactly as defined earlier but with one stipulation: the probability terms are constructed over basis terms in some fixed dimension. Thus, for example, let basis terms have the form

$$\phi_1 \equiv (Vt(\mathtt{b}^{(2)},\mathtt{b}^{(2)}))_{01} \text{ and } \phi_2 \equiv (Vt(\mathtt{b}^{(2)},\mathtt{b}^{(2)}))_{11}.$$

Then we could have a probability formula

$$\sqrt{2}(P(\phi_1))^2 + 2P(\phi_1)P(\phi_2) + \sqrt{3}(P(\phi_2))^3 < 1.$$

But formulas over different dimension are not mixed (see the preceding section for a discussion). The tensor operator t is interpreted as before. Observe the crucial associative property of the tensor product. In general,

$$\pi(t(X,t(Y,Z))) = \pi(t(t(X,Y),Z))$$

Hence, when we generate product basis terms by iteration of a *single* atomic basis term $\mathtt{b}$, due to the associative property we infer that $\pi(t(\mathtt{b},t(\mathtt{b},t(\mathtt{b}\cdots))))$ is identical to $\pi(X)$ for any product basis X obtained by rearrangement of the t-operator in $t(\mathtt{b},t(\mathtt{b},t(\mathtt{b}\cdots)))$. We write $t^n(\mathtt{b})$ for the latter if there are n, t-operators in the expression.

We introduce one more extension. If Φ is a probability formula and U is a unitary variable then $[U]\Phi$ is a formula of $\mathcal{L}(P, \mathbf{m}, t, M, S, \mathbf{U})$. The semantics of $[U]\Phi$ is given by

$$\pi, \rho \models [U]\Phi \text{ iff } \pi, [[U]]^{-1}\rho[\mathfrak{b}] \models \Phi$$

The notation $[[U]]^{-1}\rho[\pi(\mathfrak{b})]$ expresses the operation of the unitary matrix U^{-1}, whose entries are the variables $U(ij)$, on ρ expressed in the basis $\pi(t^n(\mathfrak{b}))$ where n is the number of t-operators appearing in the formula. We will often drop the square brackets unless we want to emphasize that the syntactic operator $[U]$ depends on the interpretation. Since the formulas are assumed to be homogeneous this is unambiguous. The dimension of the bases is $M \equiv m^n$. Thus, if $U = U(ij)$ and $\pi(\mathfrak{b}) = \{\alpha_0, \ldots, \alpha_{M-1}\}$ is a basis in and $\rho = |\psi\rangle\langle\psi|$ is a pure state with $\psi = \sum_i x_i\alpha_i$ then

$$[[U]]^{-1}\psi[\mathfrak{b}] \equiv |\phi\rangle = \sum_i \overline{U(ij)}x_i\alpha_j \text{ and}$$
$$[[U]]^{-1}\rho = |\phi\rangle\langle\phi|$$

This action of a unitary matrix on a vector representing the basis component reduces to ordinary matrix multiplication of a column vector by an $n \times n$ matrix when the basis symbol $\mathfrak{b}$ is interpreted as the standard basis. In the case of mixed states represented by a density matrix ρ expressed in the standard basis U acts(by linearity) as $\rho \longrightarrow U\rho U^{-1}$. However, I re-emphasize the point that it is the transformation relations (among basis or vectors) which are important not the assigned numerical values i.e. a particular interpretation. Laws of quantum theory, being valid formulas, are of course independent of such interpretations. This essential independence of the theory from choice of basis/coordinate system extends to all physical theories.

As was the case in 4.5 one can write semantically equivalent formulas without the unitary symbols to replace those containing them. But we need some new notation first. Let $t^n(\mathfrak{b})$ be tensor product of n bases in

the above notation. The interpretation is, as usual, in the space $\mathbf{C}^{m^n}$. Let $S = \{s_1, \ldots, s_k\}$ and U be a unitary matrix of order m^k then define $V \equiv U[S]$ to be matrix of order m^n that acts on the first k factors $s_1, \ldots, s_k$ factors of the basis $t^n(\mathfrak{b})$ leaving the rest unchanged. To write an explicit formula let $s_i = i, i = 1, \ldots, k$ for simplicity. Let $i = r_{n-1}(i) \cdots r_0(i)$ and $j = r_{n-1}(j) \cdots r_0(j)$ the base-m representation of the integers i and j,

$$\begin{aligned} V(i,j) &= V(r_{n-1}(i) \ldots r_0(i), r_{n-1}(j) \ldots r_0(j)) \\ &= \begin{cases} U(r_{k-1}(i) \ldots r_0(i), r_{k-1}(j) \ldots r_0(j)) \text{ if } i,j \leq k \\ \Pi_l \delta_{r_l(i) r_l(j)} \text{ otherwise} \end{cases} \end{aligned}$$

The general case may be handled by permuting the indices in the set S to bring them to the first k factors and then applying U followed by the inverse permutation. This notation will become clear when I deal with specific examples. The point is, the matrix U induces the operator $U[s_1, \ldots, s_k]$ on the larger space $\mathbf{C}^{n_1} \otimes \ldots \mathbf{C}^{n_k}$ which acts as unit operator on all but the factors at $s_1, \ldots, s_k$. This is important from the practical point of view. For example, suppose that we have a system consisting of 100 qubits. A general unitary operator on this space has order 2^{100}! To design such a unitary 'gate' we have to entangle 100 qubits, a task far beyond current technology. However, it is an important result that any such gate can be approximated by 2-qubit and single qubit gates. The challenge is to design an efficient circuit which approximates the desired circuit. To continue with discussion on notation, the interpretation of the operator $[U[s_1, \ldots, s_k]]$ on probability formulas Φ requires that the state be expressed in a product basis. Otherwise, it does not make sense. Hence, for formulas of the form $\Phi[t(\mathfrak{b}^1, \mathfrak{b}^2, \ldots, \mathfrak{b}^k)]$ and for every subset $S \subset \{1, \ldots, k\}$, $[U[S]]\Phi$ is a formula of $\mathcal{L}(P, \mathbf{m}, t, M, S, \mathbf{U})$ interpreted as follows

$$\pi, \rho \models [U[S]]\Phi[t(\mathfrak{b}^1 \ldots \mathfrak{b}^k)] \text{ iff} \tag{6.2}$$

$$U^{-1}[S]\rho[\pi(t(\mathfrak{b}^1 \ldots \mathfrak{b}^k))] \models \Phi \tag{6.3}$$

These definitions require some explanation. First, recall that the notation $\rho[\pi(t(\mathsf{b}^1 \ldots \mathsf{b}^k))]$ means that ρ is expressed in the basis given inside the brackets i.e. $\pi(t(\mathsf{b}^1 \ldots \mathsf{b}^k))$. Let $\pi(\mathsf{b}^r) = \{|\alpha_0^r\rangle, \ldots, |\alpha_{n_r-1}^r\rangle\}$, $1 \leq r \leq k$. Then the basis $\pi(t(\mathsf{b}^1 \ldots \mathsf{b}^k))$ consists of vectors of the form $\{\left|\alpha_{j_1}^1\right\rangle \otimes \ldots \otimes \left|\alpha_{j_k}^k\right\rangle$, where the components $\{j_1, \ldots, j_k\}$ vary over appropriate range i.e. $0 \leq j_r \leq n_r$. It is convenient to drop the tensor product symbol $\otimes$ from the formalism. This will be often adopted in the subsequent formulas. First, suppose that ρ is a *pure* state $|\psi\rangle\langle\psi|$. The state vector ψ can be written as

$$\psi = \sum_{j_1,\ldots,j_k} c_{j_1,\ldots,j_k} \left|\alpha_{j_1}^1\right\rangle \ldots \left|\alpha_{j_k}^k\right\rangle.$$

The operator $U^{-1}[S]$ acts on the coefficients $c_{j_1,\ldots,j_k}$ transforming the set of indices contained in S and leaving the rest unchanged. As a simple illustration, let $S = \{1, 2\}$. Then, only the first two indices are affected. The matrix U is of order m^2. Write the entries of U as $U(j_1 j_2, l_1 l_2)$. Then,

$$c_{j_1,\ldots,j_k} \longrightarrow \sum_{j_1 j_2} \overline{U(j_1 j_2, l_1 l_2)} c_{j_1,\ldots,j_k} \tag{6.4}$$

The action of U on a general (mixed) state operator extends by linearity. Explicitly, let $V|\psi[\mathsf{b}]\rangle$ denote the action of a unitary matrix V on the vector ψ defined with respect to the basis $\pi(\mathsf{b})$ for some interpretation. Then the action of V on the density operator $|\psi\rangle\langle\psi|$ is given by

$$V \cdot (|\psi[\mathsf{b}]\rangle\langle\psi[\mathsf{b}]|) = V(|\psi[\mathsf{b}]\rangle\langle\psi[\mathsf{b}]|)V^{-1}.$$

This action reduces to matrix multiplication when the basis variable b is interpreted as the standard basis. The point is, the unitary matrices of interest are constant numerical matrices like the single qubit gate defined in the last chapter but the basis variable can represent any orthonormal basis in the appropriate Hilbert space. Therefore, the transformation are defined with respect to the later. Now a density operator ρ can be written as $\rho = p_1 |\psi_1\rangle\langle\psi_1| + \ldots + p_r |\psi_r\rangle\langle\psi_r|$ with $p_1 + \ldots + p_r = 1$. Then $V \cdot (\rho) =$

$\sum_i p_i V \cdot (|\psi_i\rangle\langle\psi_i|)$. The special type of unitary operations defined above can be written directly by giving the entries explicitly (Eq.6.4). We may treat the new notation $U[S]$ as "syntactic sugar" which expands to the latter. But notation $U[S]$ is a succinct representation telling us which parts are affected.

Now we define probability atoms by substituting probability terms in some multivariate polynomial as before. A probability formula is a Boolean combination of probability atoms. We write $\Phi[Y]$ for a probability formula over basis expression Y.

Lemma 17 *Let $S \subset J_k = \{1, \ldots, k\}$. Let Φ be a probability formula over the product base $t^n(\mathtt{b})$. Then the formula*

$$[U[S]]\Phi[t^n(\mathtt{b})] \Leftrightarrow \Phi[U[S]t^n(\mathtt{b})]$$

is valid in $\mathcal{L}(P, \mathbf{m}, t, M, S, \mathbf{U})$.

Proof: The lemma is almost immediate from the definitions. We observe that for any basis term X_i

$$[\![P(UX_i)]\!]_\rho = \mathtt{Tr}(\rho U \pi(X_i)) = \mathtt{Tr}(U \cdot \rho\pi(X_i))$$

This is proved first in the case when $\rho = |\psi\rangle\langle\psi|$ is a pure state. Thus let $\pi(X_i) = |\alpha_i\rangle\langle\alpha_i|$. Then by definition $\pi(UX)_i = U|\alpha_i\rangle \equiv \sum_k U_{ki}|\alpha_k\rangle = |\beta_i\rangle$, say. If $\psi = \sum c_i |\alpha_i\rangle$ then

$$U|\psi\rangle = \sum c_i U|\alpha_i\rangle = \sum c_i |\beta_i\rangle$$

Hence

$$\mathtt{Tr}(\rho U|\alpha_i\rangle) = |\langle\psi|\beta_i\rangle|^2 = |\langle\psi'|\alpha_i\rangle|^2$$

where $|\psi'\rangle = U^{-1}\psi$. But $|\langle\psi'|\alpha_i\rangle|^2 = \mathtt{Tr}(U \cdot \rho\alpha_i|)$ and the conclusion of the lemma holds in the case of pure state. The general case follows by linearity since the trace operator is linear. □

The lemma is a formalization of a simple but important notion. As far as the probabilities are concerned we may take one of the two alternative views. We may consider the state as fixed and the measurement is done with respect to the basis $\{|\beta_i\rangle\} = |\alpha_i\rangle\}$. Alternatively, we may view the state transformed by the operator U^{-1}. The probability distributions remain same. The equivalence of the two approaches are rooted in the basic paradigm of quantum mechanics: it is the probabilities which are the direct outcomes of measurement. The state is not directly observable.

For convenience I summarize the languages discussed so far, along with their salient features in a tabular form(see the following table). The language $\mathcal{L}(P, \mathbf{m}, t, M, S, \mathbf{U})$ is very general and as seen from the lemma several fragments of the language are equally expressive. The reason for considering $\mathcal{L}(P, \mathbf{m}, t, M, S, \mathbf{U})$ is precisely this. Because it is an expansion of all these fragments one can easily demonstrate equivalences. Note that, by definition, they are all interpreted in the same structure, *viz.* the infinite dimensional Hilbert space of the preceding chapter. The language $\mathcal{L}(P, \mathbf{m}, t, M, S, \mathbf{U})$ roughly corresponds to the interaction picture of quantum theory (see the chapter on quantum theory). The fragment $\mathcal{L}(P, \mathbf{m}, t, M, S)$ corresponds to the Heisenberg picture. In contrast, the fragment $\mathcal{L}(P, t, M, S, \mathbf{U})$ consisting of the probability, tensor, and measurement operators and unitary symbol corresponds to the Schroedinger picture. The meaning of the word "operator" is overloaded. On one hand it is meant to represent syntactic constructs like the operator t on basis symbols or the operator M on formulas. On the other hand, these are interpreted as algebraic operators on appropriate algebraic structures. Note also that the measurement operators are common to all pictures. The reason is, it is implicit in the present analysis that measurement is to be imagined as a classical operation of 'reading the pointer value' on a *classical* apparatus. That is, although this device interacts with the quantum system to give the pointer reading, the interaction is outside the domain of analysis. We only know that we get definite readings with

Language	Operators	Quantum Picture
$\mathcal{L}(P, \mathbf{m}, t, M, S, \mathbf{U})$	probability, transformation matrix, tensor operator, measurement, selection, unitary	Interaction
$\mathcal{L}(P, \mathbf{m}, t, M, S)$	probability, transformation matrix, tensor operator, measurement, selection	Heisenberg
$\mathcal{L}(P, t, M, S, \mathbf{U})$	probability, tensor operator, measurement, selection, unitary	Schroedinger
$\mathcal{L}(P, \mathbf{m})$	probability, transformation matrix	Heisenberg

Table 6.1: Summary of languages

probability distributions given by quantum theory. This view is a watered down version of the operational viewpoint of quantum theory which avoids the intricacies and pitfalls of quantum measurement theory. Of course, this simplistic version does not stand up to rigorous analysis. I refer the reader to the literature (e. g. [Per95], [BLM91]) for more sophisticated analysis.

The logics developed in this work are motivated by quantum computing and information (QCI). In the framework of QCI we consider several copies of some quantum system. Usually, it is some 2-dimensional systems called qubits. Hence, the above formalism with one atomic basis symbol is adequate. The fragment that we will adopt for discussing several examples of QCI consists of this basis symbol, the tensor operator t, the probability operator P, and the operators $[U]$ over probability formulas for a unitary symbol

U, and the measurement operators. We call this language $\mathcal{L}_n(P, t, M, S, \mathbf{U})$, when the atomic basis symbol is of dimension n. I summarize the semantics below.

1. Let n be a positive integer. Since the formulas are homogeneous a typical formula of t-degree k is interpreted in a space of dimension n^k. Let $H_n = \mathbf{C}^n$ with standard inner product. Then, $\mathcal{L}_n(P, t, M, S, \mathbf{U})$ is interpreted in the space:

$$H = H_n + H_n \otimes H_n + \ldots + H_n^{\otimes k} + \ldots$$

The inner product in $H_n \otimes H_n$ has the following property. If $|\alpha_i\rangle = |\beta_i\rangle \otimes |\gamma_i\rangle$ for $i = 1, 2$ then $\langle\alpha_1|\alpha_2\rangle = \langle\beta_1|\beta_2\rangle\langle\gamma_1|\gamma_2\rangle$.

2. There is only one basis symbol, say $\mathtt{b}$, of dimension n. Now the unitary variables are interpreted as matrices of indicated dimension. Thus, in $[U]\Phi[t(\mathtt{b}, \mathtt{b})]$, where Φ is a probability formula over the product basis $t(\mathtt{b}, \mathtt{b}) \equiv t^2(\mathtt{b})$, U is interpreted as an $n^2 \times n^2$ matrix.

Observe that, because the formulas are homogeneous, the extended logic is similar to the one presented in last chapter for quantum systems of fixed dimension but without tensor product. We will have more about this point later. In the next subsection I use the language $\mathcal{L}_2(P, t, M, S, \mathbf{U})$ to express quantum circuits as formulas.

6.2 Quantum Circuits

Informally, a quantum circuit transforms a given input quantum state into some desired output state. The input is a quantum system of a a finite set of qubits. The desired transformation is then a unitary operation. But this statement is about as informative as saying that an m-input and n-output Boolean circuit is a function $\{0,1\}^m \longrightarrow \{0,1\}^n$. In the case of the Boolean circuits one starts with a finite set or a finite family of gates (boolean functions) and tries to build more complex functions or circuits.

The 'gates' in the quantum case are some basic unitary operations like the ones considered in the previous subsection. But there are some subtleties and novelties in the quantum case. Some of these is discussed below.

First, I give a brief overview of the classical theory of circuit complexity [Vol98]. An n-ary boolean function is function $\{0,1\}^n \longrightarrow \{0,1\}$. A family of boolean functions is a sequence $f = (f^n)$ such that f^n is an n-ary boolean function. For example, if $\wedge^n(x_1, \ldots, x_n) \stackrel{\text{def}}{=} x_\wedge \ldots \wedge x_n$ is the n-input *AND* operation then $(\wedge^n), n = 12, \ldots$ constitute a family of boolean functions. A boolean basis B is a finite set consisting of boolean functions or families of Boolean functions.

Definition 2 *A boolean circuit is a tuple*

$$C = (V, E, B, \alpha, \beta, \omega, \mathbf{In}, \mathbf{Out})$$

The pair (V, E) is a finite directed acyclic graph, with vertices V and edges E, B is a Boolean basis and $\alpha : E \rightarrow N$ is an injective function. **In** *and* **Out** *are two finite sets of Boolean variables, $\beta : V \rightarrow B \bigcup \mathbf{IN}$, and $\omega \rightarrow$* **Out** *are two functions. Further, the following conditions must be satisfied.*

1. *If $v \in V$ has in-degree 0, then $\beta(v)$ is a 0-ary boolean function (i. e. a boolean constant from B) or $\beta(v) \in$* **In**.

2. *If $v \in V$ has in-degree $k > 0$, then $\beta(v)$ is a k-ary boolean function from B.*

3. *Let n and m be the cardinality of the set* **In** *and* **Out** *respectively. For every $i, 1 \leq i \leq n$ there is at most one node $v \in V$ with indegree such that $\beta(v) = x_i \in$* **In**.

4. *For every $i, 1 \leq i \leq m$ there is at most one node $v \in V$ with outdegree 0 such that $\omega(v) = y_i \in$* **Out**. *There is a special symbol $* \in$* **Out** *such that if $\omega(v)$ is not some y_i then it is $*$.*

The sets **In** and **Out** contain the input and output variables respectively. If v has in-degree k_0 and out-degree k_1 then v is said be a *gate* with fan-in k_0 and fan-out k_1. An edge $e = (u, v) \in E$ is called an input wire of v and output wire of u. If $\beta(v) = x_i$ for some i then v is an input gate. If $\omega(v) \neq *$ then v is an output gate. For a fixed node v the function α induces an ordering of the vertices which have edges *into* v. Let the cardinality of **In** and **Out** be m and n respectively. We say that the circuit C with m inputs and n outputs computes the function

$$f_C : \{0,1\}^m \longrightarrow \{0,1\}^n$$

defined recursively as follows. If v is an input node then the fanin $k = 0$. Given $(a_1, \ldots, a_m) \in \{0,1\}^m$ define $Val_v(a_1, \ldots, a_m) = a_i$ for an input node v with $\beta(v) = x_i$. If v is not an input node let it have fanin k and let $v_1, \ldots, v_k$ be the nodes corresponding to the edges directed *into* v, such that $\alpha(v_1) < \ldots < \alpha(v_k)$. Then,

$$Val_v(a_1, \ldots, a_m) = f(Val_{v_1}(a_1, \ldots, a_m), \ldots, Val_{v_k}(a_1, \ldots, a_m))$$

where $\beta(v)$ is a boolean function f. Let w_i be the unique output node with $\omega(w_i) = y_i$. The function f_C computed by the circuit is defined by

$$f_C(a_1, \ldots, a_m) = (Val_{w_1}(a_1, \ldots, a_m), \ldots, Val_{w_n}(a_1, \ldots, a_m)).$$

Definition 3 *The size of a circuit is the number of nodes in the circuit. The depth of circuit is the length of a longest directed path.*

The definition of quantum circuits is similar to the classical except in two crucial aspects. The quantum "gates" are unitary operators and hence invertible. This implies that the 'fanin' and 'fanout' degrees of a node that is not an input/output are equal. The second difference will be explained after the following:

Definition 4 *A U-basis is a finite set of unitary matrices.*

Note that U-basis contains no family of circuits. Now, for the second point of difference between classical and quantum circuit theory. A function f_C computed by a classical circuit C may be written as a composition of circuit functions corresponding to the basis. That is a general function $f : \{0,1\}^m \to \{0,1\}^n$ is obtained *exactly* as composition of some basis functions. As both the domain and range are finite sets it is possible to have a finite circuit basis. In the case of quantum circuits the functions computed are unitary operators mapping $\mathbf{C}^n \to \mathbf{C}^n$. The cardinality of the set of unitary operators is $2^{\aleph_0}$, the cardinality of the real axis. A finite basis is thus ruled out since a finite basis will generate at most a *countable* set of operators. What one may hope to accomplish with a finite basis is to approximate, within any prescribed error margin, any given unitary operator with a finite basis. A formal definition will be given shortly.

First, the reason why a U-basis is required to be finite. A general unitary operation on n qubits involves entangling the qubits. Entanglement of more than 10 qubits poses enormous technical problems and more than, say, 1000 qubits may prove to be impossible due to decoherence effects. In any case, it has been demonstrated that a 2-qubit gate and a finite number of single qubit gates are adequate to approximate an arbitrary unitary operator.

Definition 5 *Let H be a Hilbert space and A a linear operator on H. Given a real number $\varepsilon > 0$, a linear operator B on H is ε-close to A if $E(A,B) < \varepsilon$, where*

$$E(A,B) = sup\{\|(A-B)\,|\psi\rangle\,\| \mid \|\,|\psi\rangle\,\| = 1\} \tag{6.5}$$

The term $E(A,B)$ is called the error in the approximation of A by B. The function $A \to \sup\{\|(A-B)\,|\psi\rangle\,\| \mid \|\,|\psi\rangle\,\| = 1\}$ is a *norm* on the set of operators on H [HJ90]. Hence, $E(A,B)$ may be viewed as a distance between the matrices A and B. One may also use other matrix norms to define approximations. But in a real experimental situation in regard to a quantum systems we usually deal with probability distributions. Suppose, we have an n-dimensional quantum system S and want to verify a claim that a unitary

operator B approximates A within an error bound less than 10^{-6}. If we have efficient algorithms for computing entries of A and B then an error estimate is easy. But if we do not have such knowledge and want to use our ensemble of quantum systems to test the claim, then a direct approach would be to prepare copies of S in some initial states chosen from a basis $\mathcal{B}$ and apply A to half of the copies and B to other half. A complete measurement in $\mathcal{B}$ would yield probability distributions corresponding to A and B applied to the initial state. Therefore, it would be useful to define approximation directly in terms of probability distributions. After all, the unitary operators have to be realized in a real physical situation and such that their application have observable consequences consistent with the theory. A simple example will illustrate my point. The unitary operator $e^{ia}I_n$ for real a multiplies every vector by a phase e^{ia}. We do not distinguish between a state $|\psi\rangle$ and $e^{ia}|\psi\rangle$ since the probability distributions in the two situations are identical. Therefore, no measurement can possibly detect that a unitary operation has been performed. A deeper reason for this invariance follows from the fundamental principle that shifting the origin of the time coordinate cannot have any observable consequence in an isolated system. The operator corresponding to this constant translation of time coordinate is precisely of the form $e^{ia}I_n$. Setting $a = \pi$ we get the operator $-I_n$. This is identical to the identity operator as far as the probability distributions are concerned. But according to the definition 5, since $\|(I_n - (-I_n))|\psi\rangle\| = 2$ the two operators are not even close!

I therefore, propose below an alternative definition in terms of probability distributions. In the following U and V are unitary operators on some Hilbert space H of dimension n.

Definition 6 *Given $\varepsilon > 0$, two unitary operators U and V are said to be ε-approximate (ε-app in short) if for every unit vector $|\psi\rangle$*

$$0 \leq 1 - |\langle\psi|V^{\dagger}U|\psi\rangle|^2 = 1 - |\langle\psi|V^{-1}U|\psi\rangle|^2 < \varepsilon. \tag{6.6}$$

Note that

$$|\langle\psi|V^\dagger U|\psi\rangle| \leq \|V|\psi\rangle\| \|U|\psi\rangle\| = 1$$

The first inequality is an instance of Cauchy-Schwartz inequality and the equality follows from the fact that U and V are unitary operators and hence norm preserving. Consequently, the first inequality in 6.6 always holds. For any two unitary matrices U and V on a Hilbert space H and $\psi \in H$, let

$$D_\psi(U,V) \stackrel{\text{def}}{=} 1 - |\langle\psi|V^\dagger U|\psi\rangle|^2 \tag{6.7}$$

We first note that if $U' = e^{ix}U$, for some real x, then $D_\psi(U',V) = D_\psi(U,V)$. In particular, $D_\psi(U,U') = 0$. This implies that two unitary operators differing only by a phase factor are arbitrarily close. Let us define an equivalence relation on the set of unitary operators,

$$U \sim U' \text{ iff } U' = e^{ix}U \text{ for some real } x$$

Let $\mathcal{U}_n$ be the set of equivalence classes. From the above discussion we see that D_ψ is well defined on $\mathcal{U}_n$. Let

$$D(U,V) = \max\{\sqrt{D_\psi(U,V)} | \|\psi\| = 1\}$$

Then we have the following

Theorem 15 *For any vector $\psi \in H$, D is a metric on $\mathcal{U}_n$.*

I omit the proof since it is not essential to later developments (see [Pat06]). Also note that the definition of ε-approximation for two unitary operators may be recast as

$$D(U,V) < \sqrt{\varepsilon}$$

First, I prove that this definition for approximating one unitary operator by another is equivalent to the standard definition 5 , provided we make a minor modification stated below. Call two unitary operators U and V ε-close up to a phase, if there is a complex number e^{ia}, a real such that when V is replaced by $e^{ia}V$ equation 6.5 holds.

Theorem 16 *If two unitary operators U and V are ε-app then they are $c\sqrt{\varepsilon}$ close up to a phase for some $c > 0$, independent of ε and the dimension of the space. Conversely, if they are ε-close then they are ε^2 approximate.*

Proof: Let the unitary operators U and V be ε-approximate. First some immediate consequences of the definition:

Let $|\psi\rangle \in H$ and put $|\phi\rangle = V^\dagger U |\psi\rangle \equiv W' |\psi\rangle$. Let $\mathcal{B} = \{\alpha_i\}_{i=0}^{n-1}$ be any basis. Since $\sum_i |\langle\alpha_i|\phi\rangle|^2 = 1$, putting $|\psi\rangle = \alpha_i$ in equation 6.6 we get, for $0 \leq i \leq n-1$

$$0 \leq 1 - |\langle\alpha_i|W'|\alpha_i\rangle|^2 < \varepsilon \Rightarrow \sum_{j,j\neq i} |\langle\alpha_j|W'|\alpha_i\rangle|^2 < \varepsilon S \tag{6.8}$$

Also note that $|\langle\psi|U^\dagger V|\psi\rangle| = |\overline{\langle\psi|U^\dagger V|\psi\rangle}| = |\langle\psi|V^\dagger U|\psi\rangle|$.
Now we choose a special basis. An operator A is called normal if $AA^\dagger = A^\dagger A$. Hermitian and unitary operators are normal. It is a standard result in linear algebra that there is an orthonormal basis such that each basis vector is an eigenvector of the normal operator A [HJ90]. Call this the diagonalizing basis. Since W' is unitary it possesses such a diagonalizing basis, say $\{\beta_i\}_0^{n-1}$. It is easy to see that eigenvalues of any unitary operator lie on the unit circle in the complex plane. Hence, let

$$W' |\beta_i\rangle = e^{i\theta_i'} |\beta_i\rangle \,,\ i = 0, \ldots, n-1.$$

Using $e^{-i\theta_0'}V$ instead of V in the approximation we get a new operator $W = e^{-i\theta_0'}V^\dagger U = e^{-i\theta_0'}W'$. It follows immediately that

$$W |\beta_0\rangle = |\beta_0\rangle \text{ and} \tag{6.9}$$

$$W |\beta_i\rangle = e^{i(\theta_i' - \theta_0')} |\beta_i\rangle \equiv e^{i\theta_i} |\beta_i\rangle \text{ for } i = 1, \ldots, n-1. \tag{6.10}$$

For any vector $|\psi\rangle = \sum_{i=0}^{n-1} x_i |\beta_i\rangle$,

$$\langle\psi|W|\psi\rangle = \sum_i |x_i|^2 e^{i\theta_i},\ \theta_0 = 0. \tag{6.11}$$

As U and V are ε-app it follows that

$$\begin{aligned} 1-|\langle\psi|W|\psi\rangle|^2 = 1-|\sum_i |x_i|^2 e^{i\theta_i}|^2 = \\ = 1-(\sum_i |x_i|^2\cos(\theta_i))^2+(\sum_i |x_i|^2\sin(\theta_i))^2 < \varepsilon \end{aligned} \tag{6.12}$$

Now apply the above relation to a unit vector $|\phi\rangle = x_0\,|\alpha_0\rangle + x_i\,|\alpha_i\rangle\,,\ i\neq 0$. First, $|x_0|^2+|x_i|^2=1$ and the other coefficients are zero. Recall that $\theta_0=0$.

$$\begin{aligned} |\langle\phi|W|\phi\rangle|^2 &= (|x_0|^2+|x_i|^2\cos(\theta_i))^2 + +(|x_i|^2\sin(\theta_i))^2 \\ &= |x_0|^4+|x_i|^4+2|x_0|^2|x_i|^2\cos(\theta_i) = 1-2|x_0|^2|x_i|^2(1-\cos(\theta_i)) \end{aligned} \tag{6.13}$$

To derive the last equality we use $|x_0|^2|x_i|^2=1$. By hypothesis $1-|\langle\phi|W|\phi\rangle|^2 < \varepsilon$. Hence, the above equation implies

$$2|x_0|^2|x_i|^2(1-\cos(\theta_i)) < \varepsilon. \tag{6.14}$$

This relation must hold for any choice of x_0 and x_j as long as they satisfy the constraint $|x_0|^2+|x_i|^2=1$. In particular, it must be true for the maximum value $1/4$, of $|x_0|^2|x_i|^2$. Hence,

$$1-\cos(\theta_i) < 2\varepsilon. \tag{6.15}$$

Now, let $|\psi\rangle = \sum_i x_i\,|\beta_i\rangle$ be an arbitrary unit vector. According to the definition 6.5 the error in approximating U by another unitary operator V 'ε- close' to it is

$$\begin{aligned} E(U,V) &= \max\,\|(U-V)\,|\psi\rangle\,\| \\ &= \max\,(\langle\psi|(U-V)^\dagger(U-V)|\psi\rangle)^{1/2} \\ &= \max\,(2(1-\mathrm{Re}(\langle\psi|V^\dagger U|\psi\rangle)))^{1/2} \\ &= \sqrt{2}\max\,(1-\mathrm{Re}(\langle\psi|W|\psi\rangle))^{1/2} \\ &= \sqrt{2}\max\,(1-(\sum_{i=0}^{n-1}|x_i|^2\cos(\theta_i)))^{1/2} \\ &= \sqrt{2}\max\,(\sum_i |x_i|^2(1-\cos(\theta_i)))^{1/2} \end{aligned} \tag{6.16}$$

Hence, using the estimate 6.15 we get

$$E(U,V) = \sqrt{2}\max(\sum_i |x_i|^2(1-\cos(\theta_i)))^{1/2} < 2\sqrt{\varepsilon}\sum_i |x_i|^2 = 2\sqrt{\varepsilon}$$

The first statement of the theorem is proved. To prove the second statement assume that $E(U,V) < \varepsilon$. Then, from the equations 6.16 it follows that $1 - \mathrm{Re}(\langle\psi|W|\psi\rangle) < \varepsilon^2/2$. Therefore,

$$\begin{aligned} 1 - |\langle\psi|W|\psi\rangle|^2 &= 1 - (\mathrm{Re}(\langle\psi|W|\psi\rangle))^2 - (\mathrm{Im}(\langle\psi|W|\psi\rangle))^2 \\ &\leq 1 - (\mathrm{Re}(\langle\psi|W|\psi\rangle))^2 = (1 + \mathrm{Re}(\langle\psi|W|\psi\rangle))(1 - \mathrm{Re}(\langle\psi|W|\psi\rangle)) \\ &\leq 2(1 - \mathrm{Re}(\langle\psi|W|\psi\rangle)) < \varepsilon^2 \end{aligned} \tag{6.17}$$

The theorem is proved. □

We thus see that the present definition of unitary approximation, based on observable probability distributions, is weaker($O(\sqrt{\varepsilon})$ than the standard one. The reason is in the definition of ε-app involves the *square* of amplitudes $\langle\psi|W|\psi\rangle$. It is also clear that the verification of two definitions, ε-close and ε-app has the same *classical* complexity. However, in a hypothetical quantum computer we may simulate the unitary operation W as follows. Recall that $W \equiv V^\dagger U$. Suppose we are given two circuits C_1 and C_2 which implement unitary operators U and V respectively. Now, prepare ensembles in some basis as input and apply them to the *output* nodes of C_2 and the resulting output at the input node of C_2 is applied to C_1's input nodes and a measurement in the same basis is done at the latter's output nodes. Do these for different *independent* bases. If the probability distributions satisfy the condition for ε-approximation we are done. The definition 6 fits in neatly with the logics developed in this work. Its usefulness will be further demonstrated in the developments of quantum circuits below.

Definition 7 *A quantum circuit or Q-circuit, for short, like its classical counterpart is defined as a directed acyclic graph with the nodes labeled by*

unitary gates from some U-basis. More precisely, let $T_n \equiv \{x_1, \ldots, x_n\}$ *and* $S_n \equiv \{y_1, \ldots, y_n\}$ *be two sets of boolean variables. Then a Q-circuit with n inputs is a tuple,*

$$(V, E, U_B, \mathbf{M}_B, \alpha, F, \beta, \omega, \{G_k : 0 \leq k < 2^n\}$$

Here, (V, E) *is a finite directed acyclic graph,* $\alpha : E \to \{1, \ldots, n\}$, U_B *is a U-basis,* $\mathbf{M}_B$ *is a set of measurement operators,* $F : \{0,1\}^n \to \mathbb{C}$ *and* $G_k : \{0,1\}^k \to \mathbb{C}$, $\beta : V \to U_B \cup \{x_1, \ldots, x_n\}$, *and* $\omega : V \to \{y_1, \ldots, y_n\} \cup \{*\} \cup \mathbf{M}_B$ *such that the following are satisfied. We write,* $F(x_1, \ldots, x_n)$ *and* $G(y_1, \ldots, y_k)$ *for the two functions.*

1. If a node has non-zero in-degree and out-degree then they are equal.

2. First we define the level of node in $v \in V$. If it has in-degree 0 then its level $l(v)$ is zero otherwise it is the length of the longest path from some input node. Then, for nodes at a fixed level α is injective when restricted to edges going into those nodes.

3. If a node $v \in V$ has in-degree 0, then $\beta(v) \in \{x_1, \ldots, x_n\}$. If it has in-degree $k > 0$ then $\beta(v) \in U_B$ is a unitary operator of order k.

4. There is exactly one node with $\beta(v) = x_i$. Similarly, there is exactly one node w with $\omega(w) = y_i$. However, there could be several nodes with $\omega(w) = M \in \mathbf{M}_B$. Each such measurement M is actually a collection of projection operators (we consider only such measurements) $\{\Pi_i\}$ such that $\sum_i \Pi_i = 1$.

5. $$\sum_{x_1,\ldots,x_n=0,1} |F(x_1, \ldots, x_n)|^2 = 1 \text{ and } \sum_{y_1,\ldots,y_n=0,1} |G_k(y_1, \ldots, y_n)|^2 \leq 1$$

   If there are no measurements and the function ω is *onto* then

   $$\sum_{y_1,\ldots,y_n} |G_k(y_1, \ldots, y_n)|^2 = 1$$

The function α induces an ordering among the vertices which have edges going *into* given vertex v. The function F is the input function that assigns to each value of the boolean tuple a complex number such that the above equation is satisfied. That is, it defines a unit vector in $\mathbb{C}^{2^n}$. Similarly, the family of functions $\{G_j\}$ define a *basis* in which the final readout is carried out. The condition that in-degree (if not zero) be equal to outdegree reflects the fact that all gates are reversible. The function ω labels the outputs which are measured. It is assumed that the operators I_2, the identity operator of order 2 belongs to U_B so that we may add a node which is labeled by a gate that leaves the input unaffected. For simplicity, I have assumed that all possible n outputs are measured. Let us now define the function computed by the Q-circuit *before* the final measurements. This is done recursively on the levels. Define

$$C_0(x_1, \ldots, x_n) = F(x_1, \ldots, x_n)$$

Let $v_1, \ldots, v_k$ be nodes at level r and let $v_{ij_1}, \ldots, v_{ij_i}$ be the nodes that precede v^i. Let $\beta(v_i) = U_i$. Recall that for a unitary operator operator of order $m \leq n$ and $\{i_1, \ldots, i_m\} \subseteq \{1, \ldots, n\}$ the notation$U[i_1, \ldots, i_m]$ means that U is applied to the qubits at $\{i_1, \ldots, i_m\}$. Then,

$$C_r(x_1, \ldots, x_n) = U_1[\alpha(v_1, v_{1j_1}), \ldots, \alpha(v_1, v_{1j_i})] \cdots U_k[\alpha(v_k, v_{kj_1}), \ldots, \alpha(v_k, v_{kj_k})]$$
$$C_{r-1}(x_1, \ldots, x_n)$$

Let m be the highest level. Then the function computed by C is given by

$$C_m(x_1, \ldots, x_n)$$

Figure 6.2 gives the diagram of the teleportation circuit with measurement at the end. We will discuss teleportation later.

The size and depth of a Q-circuit is defined as for the classical circuit. The number of nodes (gates) in the circuit is its size and the depth is the length of a maximal path from the input to the output nodes. From the above definitions the maximum level of a circuit equals its depth. All the gates are reversible. Let us focus on a special class of classical circuits.

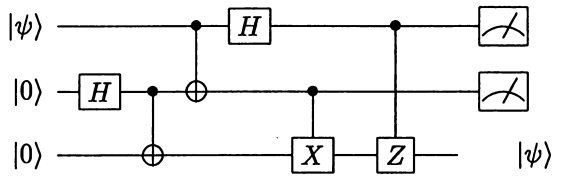

Figure 6.1: Quantum teleportation circuit with measurement at the end

Recall that a classical gate is a function $g : \{0,1\}^m \longrightarrow (\{0,1\}^n$. The gate g is called reversible if it is a bijective or invertible function. It is a well known result that any boolean circuit can be efficiently constructed using reversible gates only. We may have to use some auxiliary bits and read the output at some of the nodes only. For example the Fredkin gate is reversible, conservative and universal [FT82]. If we include all the auxiliary bits then a circuit comprised of reversible gates computes a bijective function $g : \{0,1\}^n \longrightarrow \{0,1\}^n$, for some n equal to the number of inputs (and the number of outputs). The set of all such invertible functions constitute a group G. A finite set of invertible functions is universal if it generates G. Since G is finite we always have a finite set of generators. Analogously, in the quantum case the relevant group is the group of unitary matrices of some fixed dimension. It has uncountably many elements. Hence it cannot be finitely generated. Therefore, with a finite set of unitary matrices (the U-basis) we can only approximate an arbitrary unitary matrix. That is, an arbitrary unitary matrix of order 2^n (n-input circuit) can be approximated as a product of matrices from the U-basis. Some examples of U-bases are

$$UB_1 = \{\text{H, S, C, T}\} \tag{6.18}$$

$$UB_2 = \{\text{H, S, C, Tof}\} \tag{6.19}$$

Note that the gates in the above case are 1 qubit(H, S), 2-qubit(C), or 3-qubit(Tof). When such gates are applied to $n \geq 3$ qubits it is implicit

that the other qubits are unchanged which is pictorially represented by a 'quantum wire'. A wire is therefore, a unit matrix.

6.2.1 Formulas for Quantum Circuits

In this subsection the developments of the preceding sections are used to write formulas for quantum circuits. The most convenient language for the task is $\mathcal{L}_2(P, t, M, S, \mathbf{U})$. Broadly, there are three stages in the construction of a quantum circuit. The steps in the following algorithm for translating a quantum circuit into a formula of the language corresponds to these. Assume we are given some U-basis UB. We are going to construct a circuit with n inputs.

Algorithm 1 (Representing Q-circuits as formulas) *We use the notation in Definition 7.*

1. *We recall that the notation* $t^n(\mathsf{b})$ *is a shorthand for the basis* $t(\mathsf{b}(\mathsf{b}(\cdots)))$. *The formula for the initialisation is simply*

$$\mathbf{In} \equiv P(t^n(\mathsf{b})_0) = 1 \tag{6.20}$$

This corresponds to the input gates. That is, the formula corresponding to level 0 is defined. Note also that from the semantics of the language $t^n(\mathsf{b})_0$ *can be interpreted as the first vector of* any *(ordered) basis.*

2. *Next we define formula for the level 1.*

Let $(v_1^1, v_2^1, \ldots, v_{i_1}^1)$ *be the nodes at level 1 and let* $\beta(v_j^1) = U_j^{(1)}$, $1 \leq j \leq i_1$. *Further, if* $e_{j1}^1, e_{j2}^1, \ldots, e_{jr_j}^1$ *be the edges directed* into (v_j^1) *let* $S_j^1 = \{\alpha(e_{j1}^1), \ldots, \alpha(e_{jr_j}^1), j = 1, \ldots, i_1\}$. *Then,* $\{S_1^1, \ldots, S_{i_1}^1\}$ *are disjoint subsets of* $J_n = \{1, \ldots, n\}$

The formula corresponding to output of level 1 is

$$\Phi_1 \equiv [U_1^{(1)\dagger}[S_1^1]] \ldots [U_{r_1}^{(1)\dagger}[S_{i_1}^1]](\bigwedge_j (P(t^n(\mathsf{b})_j) = x_j)) \tag{6.21}$$

Here, the index j, as usual, indicates basis components. The operator corresponds to $U^\dagger = U^{-1}$ instead of U because then the latter operates on the state. The meaning of the syntactic operator $[U^\dagger]$ is as follows. Recall that $[U]$ is a an ordered collection of variables $U(ij)$. Then, $[U^\dagger]\Phi$ is a shorthand for $[V]\Phi \wedge (\bigwedge_{ij} V(ij) = \overline{(U(ji))}$. The second conjunct is a formula of $\mathbb{RC}$. The range of j is as yet unspecified as are the real *variables x_j. This range corresponds to the qubits on which the final observation/measurement is made.*

3. *Suppose Φ_k is the formula corresponding to the output of the circuit at level k and let $\{S_1^k, \ldots, S_{r_k}^k\}$ be disjoint subsets of J_n such that S_i^k is the range of the mapping α when applied to the edges going into the gate U_i^k at level k. Then,*

$$\Phi_{k+1} = [U_1^{(k)\dagger}[S_1^k]] \ldots [U_{r_k}^{(k)\dagger}[S_{r_k}^k]]\Phi_k \tag{6.22}$$

An important point to be noted is that the matrices $U_i^{(k)}[S_i^k]$, $i = 1, \ldots, r_k$ in level k commute among themselves. Therefore, the ordering of the matrices in a fixed level does not matter. Let Φ be the final formula after all the levels, i.e. , when the output nodes have been reached.

4. *The final step is a measurement (if necessary) in some basis $V\mathfrak{b}$, usually the computational basis. In that case, the unitary operator V is the unit operator. Otherwise, the function ω specifies the measurement basis. However, in all the examples that we deal with it the measurement is done in the computational basis $\mathfrak{b}$. For simplicity, we will assume this. I assume that measurement is deferred till the end. This does not change the statistics of relevant qubits [NC01]. The circuit C is represented by*

$$C \stackrel{\text{def}}{=} \mathbf{In} \wedge \Phi \wedge \mathbf{X}, \tag{6.23}$$

where $\mathbf{X}$ is a formula of $\mathbb{RC}$. We do not specify $\mathbf{X}$ at this stage and allow it to be any formula of $\mathbb{RC}$. However, in a concrete situation

it express some relation among the $\mathbb{RC}$*-terms that appear in* Φ*. Intuitively,* $\mathbf{X}$ *would express relations among the probabilities and entries of the transition matrix.*

Several observations regarding the algorithm are in order.

1. A quantum circuit may be considered as the approximation of a unitary matrix V of large order ($\sim 2^n$) by gates from the U-basis (usually single or 2-qubit gates). A classical circuit computes a boolean function $\{0,1\}^n \rightarrow \{0,1\}^m$ whereas a quantum circuit computes a function $\mathbb{R}^n \rightarrow \mathbb{R}^n$ before any measurements. The formula representing a circuit computes precisely this function. However, the formula in fact expresses more. The output of a quantum circuit is not a deterministic real or boolean vector but *random* vector. If we consider the outcomes of final measurements we can only talk about probabilities. The formula given by the above algorithm actually gives the probability distribution as the result of applying the unitary gates in the circuit. I will further illustrate this when we discuss concrete cases.

2. The measurement operator at the final stage of the algorithm is often not necessary precisely because it is the last operation. What is of interest is the probability distributions of the outcomes- the states in the measurement basis. These probabilistic predictions are adequately described by the probability operator. The actual circuit consists of the unitary operations. The probability operator captures assertions about the behaviour of the circuit. In the rest of the section it is assumed that formulas for quantum circuit are without measurement operators unless stated otherwise.

3. As mentioned earlier, the logics developed in this work express much more than conventional quantum circuits. For example, it is possible to do universal quantum computation using measurements as basic resource [Nie03]. Behaviour of such models can also be expressed in

the logics developed. Besides, the measurement operators enable one to express some sequential circuits. I will come back to this point later.

4. The unitary operators are the basic building blocks of Q-circuits. We have symbols $[U]$ for an operator in the logic which abstracts away the essential properties of unitary operators, somewhat resembling modal operators corresponding to atomic programs in dynamic logic [Har79]. The operator $[U]$ has associated with it n^2 operators $[U(ij)]$ of complex sort in dimension n. Unfortunately, this reference to individual entries cannot be avoided as the probabilities depend directly on them. This is included in the **RC** formula **X**. For example, we always have the unitarity condition

$$\sum_i \overline{U(ij)}U(ik) = \delta_{jk}$$

The matrices/gates corresponding to the U-basis given above (the Pauli matrices, Hadamard matrix, CNOT-gate etc.) will be treated as defined constants.

Two special type of formulas corresponding to different aspects of the behaviour of circuits are given below.

Circuit Equivalence

Suppose we want to capture the fact that two quantum circuits have identical behaviour with respect to all inputs. Recall that a quantum circuit without measurements is a composition of unitary operations and hence itself a unitary operator. Further, I restrict to qubit circuits and hence the dimension is 2^k if the number of qubits is k. We introduce the following formula for a given a quantum circuit C. From the developments of the previous section a circuit C (without measurement) is of the form

$$[U_1]\ldots[U_k](\Phi) \wedge \mathbf{X}$$

Where, Φ is a probability formula and $\mathbf{X}$ is a formula of $\mathbb{RC}$. Let C_u denote the 'unitary' part of C. That is, $C_u \equiv [U_1] \ldots [U_k]$. That is,

$$C_u\Phi \equiv [U_1] \ldots [U_k]\Phi$$

Then $C_u^\dagger$ is the expression $[U_k^\dagger] \ldots [U_1^\dagger]$. One may visualize $C_u^\dagger$ as C_u reversed, that is, the output nodes of C are the input nodes of $C^\dagger$ and vice versa. Of course, to preserve the reversible nature of Q-circuits *all* output nodes of C must be used, including the ancillary 'garbage' qubits. This notation will prove very useful for expressing circuit behaviour. We should consider two circuits C and C' equivalent if and only if for any probability formula Φ,

$$\mathbf{D} \wedge \mathbf{D}' \Rightarrow ([C_u]\Phi \Leftrightarrow [C'_u]\Phi),$$

is valid. Here, $\mathbf{D}$ (resp. $\mathbf{D}$') are $\mathbb{RC}$-formulas that include definitions of the unitary operators in C (resp. C'). It is intuitively obvious that this must be the case since the unitary operators corresponding to C and C' transform any given state into the same state. Assuming that both are k input circuits, in the language $\mathcal{L}_2(P, t, M, S, \mathbf{U})$ this is elegantly captured by the validity of the following:

$$\bigwedge_i \left((P(t^k(\mathfrak{b})_i) = 1 \Rightarrow [C_u^\dagger][C'_u](P(t^k(\mathfrak{b})_i) = 1))\right) \tag{6.24}$$

It is clear that this formula follows from equivalence of C and C' since they must be the same unitary operator. Conversely, if formula 6.24 is valid then it must be true for any interpretation π of the basis of basis $\mathfrak{b}$ in 2 dimensions. Now for every such interpretation $t^k(\mathfrak{b})$ is interpreted as the product basis. Since the formula is assumed to be valid it must be true in every state. The left side of the implication in exactly one conjunct, say $P(t^k(\mathfrak{b}))_i = 1$, is true at the state $\pi(\mathfrak{b}_{i_{n-1}}) \otimes \cdots \otimes \pi(\mathfrak{b}_{i_0})$, where $i_{n-1} \cdots i_0$ is the binary representation of the integer i (possibly padded with 0's). Thus, evaluating the formula successively at the basis states it is clear that $C_u^\dagger C'_u$ acts as the unit operator on basis states and hence all states. The validity of

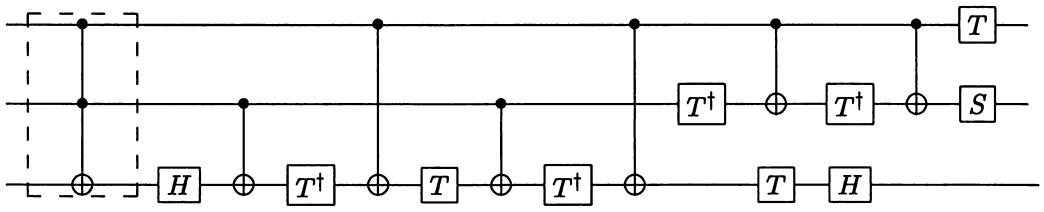

Figure 6.2: Equivalent circuits

formula 6.24 will be taken as the definition of equivalence of two circuits C and C'. Verification of equivalence of two circuits then reduces to verifying if the matrix product $C^\dagger C'$ reduces to the unit matrix.

Here is an example. The 3-qubit Toffoli(Tof) gate can be implemented using the Hadamard(H), phase(S), the $\pi/8$(T), and CNOT(C) gates.

$$\bigwedge_i P(t^3(\mathsf{b})_i) = 1 \Rightarrow$$
$$[Tof^\dagger][H[3]][C[2,3]][T^\dagger[3]][C[1,3]]$$
$$[T[3]][C[2,3][T^\dagger[2]]T[3][H[3]][C[1,2]][T^\dagger[2]][C[1,2]][T[1]][S[2]](P(t^3(\mathsf{b})_i)) = 1 \quad (6.25)$$

The figure 6.2.1 is the circuit digram showing this equivalence. The boxed circuit is the Toffoli gate, which happens to be its own inverse, and the circuit on the right of the box is its equivalent. Of course, verifying exact circuit equivalence of two circuits on a classical computer is simply a matter of multiplication of matrices. In general for an n-qubit circuit the number of basic computations (addition and multiplication) may be of the order $\exp(n)$, since the order of the matrices is 2^n. But with a quantum computer it can be done in $O(n)$ steps with provided we can implement all the gates exactly.

Approximate Equivalence

The equivalence of two circuits discussed above is exact. Far more interesting is the notion of circuit approximation or approximate equivalence Definition 6. Two circuits C and C' with k inputs are defined to be ε-approximate, for some given $\varepsilon \geq 0$ if the following formula is valid. First, let us fix some

notation. Recall that in a formula $[U]\Phi$, the operator $[U]$ is interpreted as an ordered collection of variables. In the context of a quantum circuit we assume that all the unitary operators(=gates) are from a given collection, the unitary base. Therefore, we may treat $[U]$ as collection of variable free terms of $\mathbb{RC}$. More accurately,

$$\pi, \rho \models [C]\Phi \text{ iff } \pi, \rho \models [U_1] \cdots [U_k]\Phi \wedge \mathbf{D}_C$$

where $\mathbf{D}_C$ is an $\mathbb{RC}$-formula which specifies the entries $U_i(lm)$ of the unitary operators. That is, it substitutes $\mathbb{RC}$ terms for the variables corresponding to $[U_i]$.

$$[U](P(t^k(\mathtt{b})_0) = 1)) \Rightarrow [C^\dagger][C'][U](P(t^k(\mathtt{b})_0) > 1 - \varepsilon) \tag{6.26}$$

In the above formula $t^k(\mathtt{b})$ can be interpreted as a k-fold tensor product of the basis $\pi(\mathtt{b})$ where π is an interpretation in 2 dimensions. Therefore, to ensure that some probability formula be valid in *all* bases we have to transform the product basis to an arbitrary basis. The formula 6.26 is indeed equivalent to the formula 6.6 for circuit approximation. The proof is formalized below.

Proposition 3 *Let C and C' be two circuits with k inputs and let C_u(resp. C'_u) denote the corresponding unitary operators. If the formula 6.26 is valid then for any state $|\psi\rangle \in (\mathbb{C}^2)^{\otimes k}$*

$$|\langle\psi|C_u'^\dagger C_u|\psi\rangle|^2 > 1 - \varepsilon.$$

is true.

Proof: The formula

$$\mathbf{F} \equiv [U](P(t^k(\mathtt{b})_0) = 1) \Rightarrow [C_u^\dagger][C'_u][U](P(t^k(\mathtt{b})_0) > 1 - \varepsilon)$$

is valid iff it holds for all interpretation π of the basis symbol $\mathtt{b}$, and the variables x_{ij} and for all states ρ. Write $\pi(\mathtt{b}_0) = \alpha_0$, $\pi(\mathtt{b}_1) = \alpha_1$. To simplify

notation, if $a_{k-1}\ldots a_0$ be the unique binary representation of length k, for an integer i, $0 \leq i \leq 2^k$, let $|i\rangle$ denote the vector $|\alpha_{a_{k-1}}\rangle \otimes \ldots \otimes |\alpha_{a_0}\rangle$. As long as the atomic basis vectors α_0, α_1 are clear from the context this notation is unambiguous. Now

$$\pi, \rho \models [U](P(t^k(\mathfrak{b})_0) = 1) \Rightarrow [C_u^\dagger][C_u'][U](P(t^k(\mathfrak{b})_0) > 1 - \varepsilon)$$

implies

$$\pi, U^\dagger \cdot \rho \models P(t^k(\mathfrak{b})_0) = 1 \Rightarrow \pi, U^\dagger C_u C_u'^\dagger \rho \models P(t^k(\mathfrak{b})_0) > 1 - \varepsilon$$

Now, $U^\dagger \cdot \rho \models P(t^k(\mathfrak{b})_0) = 1$ iff $U^\dagger \cdot \rho = |0\rangle$ or $\rho = U|0\rangle\langle 0|U^\dagger$. This simply means that, ρ is the pure state obtained by applying the unitary U to $|0\rangle$. Let $\psi \equiv U|0\rangle$. Then, from the semantics of the probability operator P it follows that

$$U^\dagger C_u C_u'^\dagger \cdot (|0\rangle\langle 0|) \models P(t^k(\mathfrak{b})_0) > 1 - \varepsilon \text{ iff}$$
$$0 \leq 1 - |\langle\psi|C_u C_u'^\dagger|\psi\rangle|^2 < \varepsilon$$

Any vector $|\psi\rangle$ can be reached by a unitary 'rotation' of the vector $|0\rangle$. Since the formula 6.26 is assumed to hold for any unitary operator U the proposition is proved. □

6.3 Quantum Algorithms

In this section we discuss the representation of some of the most important algorithms in quantum computation and information. First, observe that we can generalize the notion of circuit approximation, formula 6.6 by demanding it be true only for some particular inputs, say $|0\rangle$. Explicitly, let $|\alpha\rangle$ be a fixed state. Define two circuits to be ε approximate *in* state $|\alpha\rangle$ if

$$0 \leq 1 - |\langle\psi|V^\dagger U|\psi\rangle|^2 = 1 - \langle\alpha|V^{-1}U|\alpha\rangle < \varepsilon. \tag{6.27}$$

The Grover circuit [Gro96] **G** is an example of this sort of approximation. I discuss the circuit below.

The Grover algorithm

In the language $\mathcal{L}_2(P, t, M, S, \mathbf{U})$, the formula representing the Grover search circuit [Gro96] **G** is

$$\begin{aligned}\mathbf{G} \equiv & P(t^{n+1}(\mathfrak{b}_0)) = 1 \wedge \mu(V, H, O_q) \Rightarrow \\ & [V[S_n]O_q]^k[H[1]] \ldots [H[n]][HX[N+1]](P(t^{n+1}(\mathfrak{b})_{a0} \vee t^{n+1}(\mathfrak{b})_{a1}) > 3/4)\end{aligned} \tag{6.28}$$

Strictly we should call it the the Grover theorem since it is an assertion about the probabilistic output of the circuit. It is a quantum search algorithm. It works as follows. Given a number a in the range $[0, N]$, and an oracle which 'recognises' the number, the algorithm transforms a fiducial quantum state $|0\rangle$ into the state $|a\rangle$ with probability $> 1/2$ in time $O(\sqrt{N})$(=the number of calls to the the oracle). The oracle's function is to transform a particular state($|a\rangle$) leaving the others unchanged. Let us see some plausible implementation of the oracle in the classical context. Let us suppose we want to solve the satisfiability of a boolean formula F of n variables by brute search through a solution space. It is a very well-known NP-complete problem. Whatever the algorithm, the verification of an alleged solution can be done in polynomial time, given that the length of the formula is bounded by some polynomial in n. Thus, we have an oracle Turing machine which takes as input a number x_0 between 0 and 2^n and answers whether x_0, written as binary string of length n, is a truth assignment such that F evaluates to true. One may assume that the oracle performs this verification in a single step($O(1)$). This assumption does not affect the essential feature of the problem since we only have to multiply the number of times the oracle is called with an appropriate estimate of the actual time the oracle takes to get the final complexity. The details about the oracle are unimportant as long as we are assured that a polynomial time oracle *exists*. Coming back to

the search problem we simply assume that the classical oracle "recognises" the number a. That is, O_c is a boolean function $\{0,1\}^n \to \{0,1\}$, such that

$$O_c(x) = \delta_{xa}$$

That is $O_c(x) = 0$ for all x except a for which it is 1.

The Grover algorithm **G**, makes use of a quantum version of such an oracle. In more detail let, $N = 2^n$ and $a \in [0, N]$ be given. Let $B = \{|0\rangle, \ldots, |N-1\rangle\}$ be the computational basis for a system of n qubits. Recall that, when a nonnegative integer i in the interval $[0, N-1]$ is written as a binary string of length n, $|i\rangle$ is the vector which is the tensor product of the corresponding states of the individual qubits. For example, if $n = 3$, $|2\rangle = |010\rangle = |0\rangle \otimes |1\rangle \otimes |0\rangle$. So far, the situation is identical to the classical description. The quantum nature of the system emerges when we admit states which are superposition of classical states. The quantum oracle is a *unitary* operator O_q of order $2N = 2^{n+1}$ such that,

$$O_q |x\rangle |r\rangle = |x\rangle |r + O_c(x)\rangle . \tag{6.29}$$

Here, $|x\rangle$ and $|r\rangle$ are n-qubit(dimension=2^n) and single qubit vectors respectively. The matrix O_q is $2N \times 2N$ given by

$$O_q(ij) = \delta_{r_1(i)r_1(j)}[\delta_{r_0(i)r_0(j)}(1 - \delta_{r_1(i)a}) + (1 - \delta_{r_0(i)r_0(j)})\delta_{r_1(i)a}]$$

Here $r_0(i) = i \mod 2$ and $r_1(i) = [i/2]$, the greatest integer $\leq i/n$. The Grover circuit is first described informally[NC01]. Although, popularly known as Grover algorithm it really is a circuit with probabilistic output.

1. Prepare an initial state $|0\rangle^{\otimes(n+1)}$.

2. Apply $H^{\otimes n}$ to the first n qubits(reading from right) and HX to the last. Recall that H is the Hadamard matrix and X is the Pauli-X gate, both of order 2. The notation $|0\rangle^{\otimes(n+1)}$ stands for the $(n+1)$-fold tensor product of $|0\rangle$. Similarly, $H^{\otimes n}$ is the n-fold tensor product of the *matrix* H. Simply put, we apply H to each of the first n qubits and HX to the last.

3. Let $|\psi\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle$ be the state which is an equal superposition of all the states in computational basis. As I have explained above the numbers i are written as binary strings of length n and $|i\rangle$ is the product of the corresponding single qubit states. Equal superposition implies a measurement in the computational basis will yield the results $|i\rangle$ with equal probability($=1/N$). Define the operator

$$V = 2|\psi\rangle\langle\psi| - I_N.$$

which is a reflection about the plane perpendicular to $|\psi\rangle$. Its matrix with respect to the computational basis is given by

$$V(ij) = 2N^{-1} - \delta_{ij},$$

that is, V has off-diagonal entries $2N^{-1}$ and diagonal entries $2N^{-1}-1$. Now apply O_q to the output of step 2 and then V to the first n qubits. Call this combined application of O_q followed by V the Grover operator G.

4. Measure the output $|\alpha\rangle$ after $\sqrt{N}$ applications of G.

Assertion. The result of the measurement in the computational basis will yield $|a\rangle$ with probability $\geq \frac{3}{4}$ for $N > 200$. These bounds can be improved depending on the number of iterations but for the present they suffice for the purpose of illustration.

Now apply Algorithm 1 to construct a formula for the Grover *theorem.* We write theorem instead of circuit because the formula actually expresses the probabilistic predictions of the circuit, that is, the assertion above. Let us construct the circuit in line with the circuit construction algorithm given in the section.

1. The initial state is specified with certainty or in other words with probability 1. There are $n+1$ qubits including one ancillary qubit.

$$\mathbf{In} \equiv P(t^{n+1}(\mathsf{b})_0) = 1.$$

The basis is the $(n+1)$-fold tensor product basis of the 2-dimensional basis denoted by $\mathfrak{b}$. The unitary base consists of the operators H, V, and O_q. All the operators are their own inverses since they are hermitian.

2. The Hadamard operator H is applied to first n qubits and XH applied to the last.

$$[H[1]]\dots[H[n]][H\cdot X[n+1]]\Phi,$$

where $H\cdot X$ is the product of the matrices H and X and the unspecified formula Φ is the probabilistic assertion.

3. Apply O_q followed by V to the first n qubits. Repeat this $k \equiv [\sqrt{N}]$ times. Let $S_n = \{1,\dots,n\}$.

$$[H[1]]\dots[H[n]][H\cdot X[n+1]]\underbrace{[V[S_n]O_q]\dots[V[S_n]O_q]}_{k \text{ times}}\Phi$$

4. The formula Φ asserts that the probability of obtaining $|a\rangle$ is $\geq 3/4$. Hence Φ is simply,

$$P(t(t^n(\mathfrak{b})_a, \top)) \geq 3/4.$$

There are a couple of points worth noting. The first is that we do not require that an actual measurement be performed, only the probability of a particular outcome *if* such a measurement were to be performed. The second point is that, since we started with $n+1$ qubits and measure only the first n, the state of the last qubit is immaterial.

5. Putting it all together, the formula for Grover theorem is given by

$$\begin{aligned}\mathbf{G} \equiv & P(t^{n+1}(\mathfrak{b}_0)) = 1 \wedge \mu(V,H,O_q) \Rightarrow \\ & [V[S_n]O_q]^k[H[1]]\dots[H[n]][HX[N+1]](P(t^{n+1}(\mathfrak{b})_{a0} \vee t^{n+1}(\mathfrak{b})_{a1}) > 3/4)\end{aligned} \tag{6.30}$$

Note the difference in the probability term $P(t^{n+1}(\mathfrak{b})_{a0} \vee t^{n+1}(\mathfrak{b})_{a1}$ from that given above. We are only interested in the first n qubits, the last one being

an ancillary qubit. Hence, $P(t^{n+1}(\mathtt{b})_{a0} \vee t^{n+1}(\mathtt{b})_{a1}) \equiv P(t(t^n(\mathtt{b})_a, \mathtt{b}_0 \vee \mathtt{b}_1)) \equiv P(t(t^n(\mathtt{b})_a, \top))$. This conforms to our stipulation that the probability formulas must be homogeneous. Here, the formula $\mu(V, H, O_q)$ is an $\mathbb{RC}$-formula specifying the matrices V, H, and O_q. Let U_a be the unitary transformation of order 2^n which swaps the vectors $|0\rangle$ and $|a\rangle$ and leaves the other basis vectors unchanged.

$$U_a |0\rangle = |a\rangle \text{ and } U_a |a\rangle = |0\rangle .$$

The operator U_a is different from the oracle as the latter is only a recognizer while the former actually transforms into the desired state. It is the exact operator we are looking for. Let G_c denote the composition of the unitary operators in the formula 6.28. That is,

$$[G_c] \equiv [V[S_n]O_q]^k[H[1]] \ldots [H[n]][HX[n+1]]. \tag{6.31}$$

Then write the formula $\mathbf{G}$ as,

$$\mathbf{G} = P(t^{n+1}(\mathtt{b})_0) = 1 \Rightarrow [U_a[S_n]][G_c](P(t^{n+1}(\mathtt{b})_{a0} \vee t^{n+1}(\mathtt{b})_{a1}) > 1 - 1/4). \tag{6.32}$$

Note the following. The operator U_a is defined as a numerical matrix. Thus, given a positive integer a, and an ordered basis $\mathcal{B}$ in a 2^n-dimensional Hilbert space U_a permutes the first and the $(a+1)th$ elements of $\mathcal{B}$. We can also quantify over a and since $0 \leq a < N$ this is actually bounded quantification. It follows from the formula 6.27 for approximation of a circuit in some fixed state that, the Grover algorithm is actually an approximation algorithm. The Grover circuit G_c approximates the operator U_a in state $|0\rangle$. It is clear that, a fixed state approximation like the Grover algorithm can be verified in time $T(N)$, which is a of polynomial order in N. We will come back to the complexity issues in a later section.

Let us explore further the expressivity of the logics to see whether we can go beyond mere verification. Suppose we do *not* have knowledge of Grover's clever algorithm. But we do have the oracle O_q. This requires an extra

ancillary qubit. That is, we start with the initial state

$$|0\ldots0\rangle\,|0\rangle$$

in the space $(\mathbf{C}^2)^{\otimes n}\otimes\mathbf{C}^2$. Since the 'database' of $N=2^n$ items is supposed to be unstructured the initial complete shuffling of the first n qubits, so that all the states are equiprobable, is reasonable. Moreover, assume that the oracle O_q's action to distinguish state $|a\rangle$ from the rest is implemented using the ancillary qubit and the application of the Pauli-X gate as above. Now consider the *existence* of operators V. For simplicity, I put quantifiers over unitary operators with implicit understanding that they quantify over all the variable entries in the corresponding matrix. Thus, the formula

$$\exists V\mathbf{Un}(V)$$

is a shorthand for

$$\exists x_{00}\ldots x_{n-1\,n-1}\wedge_{i,j=0}^{n-1}(V(ij)=x_{ij}\wedge\sum_k\overline{x_{ik}}x_{jk}=\delta_{ij})$$

I will consistently use this notation in what follows. Now consider the formula

$$\begin{aligned}(Pt^n(\mathsf{b}_0))=1\wedge\exists V_1\ldots V_k([V_1O_q]\ldots[V_kO_q][H[1]]\ldots[H[n]][HX[n+1]]\\ (P(t^{n+1}(\mathsf{b})_{a0}\vee t^{n+1}(\mathsf{b})_{a1})>1/2)).\end{aligned}\tag{6.33}$$

If the closed formula is satisfiable we know the existence of a circuit which transforms the state $|0\rangle$ to the state $|a\rangle$ with probability $>1/2$ and using only $k=O(\sqrt{n})$ oracle calls. The appearance of O_q along with the $V_i's$ is reasonable since at each stage we query the oracle if the search has succeeded. We can do better. We query,

$$\begin{aligned}\exists V(P(t^{n+1}(\mathsf{b}_0))=1\wedge\underbrace{[VO_q]\ldots[VO_q]}_{k\text{ times}}[H[1]]\ldots[H[n]][HX[n+1]]\\ (P(t^{n+1}(\mathsf{b})_{a0}\vee t^{n+1}(\mathsf{b})_{a1})>1/2)).\end{aligned}\tag{6.34}$$

That is, whether there is *one* matrix which, if applied k times in conjunction with the oracle, does the job. The point is, the logics developed in this work are all decidable. So both the formulas can be decided. The last chapter gives an algorithm for the general decision procedure. Although, the complexities are rather high, one can still implement these in small dimension (say $\leq 2^6 = 64$). However, we may use the inherent symmetry to try reasonable guesses. Since the circuit, if it exists, must work for a state $|a\rangle$ with any number $0 \leq a \leq N$ the matrix V must treat all the states in the basis $\{|0\rangle, |1\rangle, \ldots, |N-1\rangle\}$ on equal footing. If we then assume that the matrix V has all off-diagonal entries equal to some number and the diagonal entries equal to some other number then we get back the Grover circuit. It is then a matter of verification whether the circuit works. One may then combine the Grover circuit and the formula 6.33 to verify whether the former is the best. In principle, these questions can decided in a given dimension although not efficiently. The general algorithm is a systematic reduction of the formulas of the languages developed here to that of real closed fields. As stated earlier, this is unavoidable in any probabilistic logic of reasonable expressivity. However, the best existing algorithms for real closed fields are exponential in the number of variables. A deeper question would be, whether there are quantum algorithms for real closed fields that perform better than the classical algorithms including probabilistic and approximation algorithms.

Quantum Phase Estimation and the Shor Algorithm

Perhaps the most widely known fact about quantum computing is Shor's algorithm for factoring a large number[Sho94]. I will not go into the intricate details of the algorithm since there are very good accounts of it [NC01]. Shor's factoring algorithm is based on a quantum algorithm for finding the order of a number x modulo a number N coprime to it, that is the least positive number r such that $x^r = 1 \bmod N$. This, in turn, is based on estimation of phase of the eigenvalues of a unitary operator given the corresponding eigenvector and unitary operators corresponding to finite Fourier

transformation. We remind the reader that the eigenvalue of any unitary operator U is of the form e^{ix}, x real . The number x, determined up to a multiple of 2π is called a phase of U.

Definition 8 *Let a positive integer N be given. For any complex vector $\alpha = \begin{pmatrix} x_0 \\ \vdots \\ x_{N-1} \end{pmatrix}$ its quantum Fourier transform(QFT) is the vector*

$$F(\alpha) = \widetilde{\alpha} \stackrel{\text{def}}{=} = \begin{pmatrix} y_0 \\ \vdots \\ y_{N-1} \end{pmatrix} \text{ such that}$$

$$y_j = \sum_{k=0}^{N-1} e^{2\pi ijk/N} x_k.$$

The quantum or finite Fourier transform is a unitary operator [NC01]. The inverse is given by

$$x_j = F^{-1}(\widetilde{\alpha})_j = \sum_{k=0}^{N-1} e^{-2\pi ijk/N} y_k. \tag{6.35}$$

To express the QFT in $\mathbb{RC}$ one must be able to write the N^{th} roots of unity in $\mathbb{RC}$. This can be done in a very general setting [Zil03]. However, for the present purpose we need only a restricted expressiveness. Explicitly, for each constant integer $N > 2$ with $N \neq 4$,define a new constant as follows. First consider the formulas

$$\mathbf{A}_N(x) \stackrel{\text{def}}{=} x^N = 1 \wedge \text{Im}(x) > 0 \wedge \forall y(y^N = 1 \wedge \text{Im}(y) > 1 \Rightarrow \text{Re}(x) \geq \text{Re}(y)). \tag{6.36}$$

Intuitively, we can see it as follows. The Nth-roots of unity are distributed uniformly on the unit circle. The number 1 is always a root. Starting from the real axis if one moves counterclockwise along the circle the first root one encounters in the first quadrant is the unique number that satisfies $\mathbf{A}_N$.
One can prove the following within the theory $\mathbb{RC}$.

Lemma 18 *For every positive integer $N > 2$ the following are theorems of* $\mathbb{RC}$

$$\exists x(\mathbf{A}_N(x)) \text{ and } \mathbf{A}_N(x) = \mathbf{A}_N(y) \Rightarrow x = y. \tag{6.37}$$

Proof: Since $\mathbb{RC}$ is complete we may only prove the validity of the formula in some model. The proof is easy, using properties of sine and cosine functions. □

The predicates $\mathbf{A}_N(x)$ define a primitive N^{th} root x of 1. One can then generate all other roots by taking successive powers of x. It is then clear that if we extend the theory $\mathbb{RC}$ by adding a new constant ω_N for each positive integer N along with the following axioms.

$$\omega_1 = 1 \quad \omega_2 = -1 \quad \omega_4 = i \tag{6.38}$$

$$\forall x(x = \omega_N \Leftrightarrow \mathbf{A}_N(x)),\ N \neq 1, 2 \text{ or } 4. \tag{6.39}$$

For the special values of N, *viz.* 1,2 and 4, the roots (including the primitive roots) lie along one of the axes and these are already defined within the (unextended) $\mathbb{RC}$.

Lemma 19 *The formula*

$$x^N = 1 \Rightarrow \vee_{k \leq N}(x = \omega_N^k)$$

is a theorem of $\mathbb{RC}$.

In Section 4.2 it was shown that $\mathbb{RC}$ is a complete theory. Consider the closure $\mathbf{F}$ of the above formula. As $\mathbb{RC}$ is a complete theory $\mathbf{F}$ or $\neg\mathbf{F}$ must be a theorem. Hence, if $\mathbf{F}$ is true in one model it must be true in all models and therefore a theorem. We now consider the complex plane $\mathbf{C}$ as a model of $\mathbb{RC}$. The Nth roots of unity in $\mathbf{C}$ are given by $e^{2\pi ik/N} = \cos(2\pi k/N) + \sin(2\pi k/N)$, $k = 0, \ldots, N-1$. It is intuitively clear that

$\omega_N = e^{2\pi i/N}$ in the complex plane and generates all the Nth roots of unity. The closed formula **F** is true and the lemma follows.

Extending the theory $\mathbb{RC}$ by adding the new constants and the formulas 6.38 as axioms is an extension by definition [Sho67]. Hence, any formula containing the new constants can be replaced by an equivalent formula without them. I continue to call the extended theory $\mathbb{RC}$. Let F be a unitary matrix of order N defined by

$$F_{jk} = \omega_N^{jk}. \tag{6.40}$$

Now we are in a position to write the formula for the phase estimation algorithm. The algorithm for phase estimation is quite simple. Suppose we are given a unitary operator U of order s and an eigenvector $|u\rangle$ of U with eigenvalue u'. The eigenvalues of a unitary operator have modulus 1. That is,

$$U|u\rangle = u'|u\rangle = e^{2\pi i\varphi_u}|u\rangle$$

for some *real* φ_u. The phase estimation algorithm gives an estimate of the phase φ_u. It is outlined below.

Assume we are given a unitary matrix U and one of its eigenvectors $|u\rangle$ as above. Assume also that we have access to an oracle or "black box" which performs controlled-U^j operations for integers j in some given range. Recall that in a controlled operation there are some control qubits and some target qubits of appropriate dimension (see 5.2.1). If the controls are in some particular state the operation U is performed on the target qubit otherwise it is left unchanged.

Let $\varepsilon = 2^{-n}$ and $m = 2n$. An easy calculation shows that for $n > 2$, $m = n + \lceil \log_2(2 + \frac{1}{2\varepsilon}) \rceil$, where $\lceil x \rceil$ denotes the least integer greater than or equal to x. Prepare m control qubits in the initial state $|0\rangle$ in dimension $M \equiv 2^m$, that is, a m-qubit state all initialised to $|0\rangle$. As before, I assume that the integers are all written as binary strings of length m. Thus 0 represents a string of m zeros. The initial state of the system is then $|0\rangle|u\rangle$.

1. Use appropriate Hadamard gates to transform the control qubits so that the resulting state of control + target is,

$$\frac{1}{\sqrt{M}} \sum_{j=0}^{M-1} |j\rangle |u\rangle .$$

2. Next apply the oracle for controlled-U^j operations. The resulting state is

$$\frac{1}{\sqrt{M}} \sum_{j=0}^{M-1} e^{2\pi\iota j\varphi_u} |j\rangle |u\rangle .$$

3. Apply the inverse QFT to the control system(the first m qubits). Let the state obtained be

$$|\tilde{\varphi}\rangle |u\rangle .$$

4. A measurement in the original computational basis yields a state $|s\rangle$ such that $|s2^{-n} - \varphi| < \varepsilon$ with probability $1 - \varepsilon$.

The proof of the last statement can be found in [NC01]. It is also demonstrated there how the controlled-U^j operations can be implemented efficiently. The algorithm in the present form cannot be expressed in the languages developed here. The reason is, the phase is the exponent in the eigenvalue $u = e^{2\pi i\varphi_u}$. Hence, $\varphi_u = \ln u/(2\pi i)$. But the logarithm function is multiple valued. Therefore, it is much simpler to give the estimate directly in terms of the eigenvalue u itself. First I prove a simple but useful lemma.

Lemma 20 *For real x, y and $0 < \varepsilon < 1/2$,*

$$|x - y| < \varepsilon \Rightarrow |e^{ix} - e^{iy}| < \varepsilon$$
$$|e^{ix} - e^{iy}| < \varepsilon \Rightarrow |x - y| \pmod{2\pi} < \sqrt{2}\varepsilon$$

Proof: The following estimate for the cosine function is well-known and can easily be proved using the simple formulas $f(x) \geq 0 \Rightarrow \int_a^b f(x) \geq 0$ and $-1 \leq \cos x \leq 1$.

$$\frac{x^2}{2} - \frac{x^4}{24} \leq 1 - \cos x \leq \frac{x^2}{2} \tag{6.41}$$

Since

$$|e^{ix} - e^{iy}| = \sqrt{2(1 - \cos(x-y))},$$

the hypotheses $|x-y| < \varepsilon$ implies

$$|e^{ix} - e^{iy}| = \sqrt{2(1 - \cos(x-y))} \leq \sqrt{2(x-y)^2/2} < \varepsilon \tag{6.42}$$

Conversely,

$$|e^{ix} - e^{iy}| = \sqrt{2(1 - \cos(x-y))} < \varepsilon$$
$$\Rightarrow \sqrt{(x-y)^2(1 - \frac{(x-y)^2}{12})} < \varepsilon \tag{6.43}$$

Since $1 - \cos(x-y) < \varepsilon^2/2$, the assumption that $\varepsilon \leq 1/2$ implies that $\cos(x-y) > 3/4$. From elementary properties of the cosine function it follows that $|x-y| \pmod{2\pi} < \pi/4$. Since the phase is determined only up to a multiple of 2π, we must have $|x-y| \pmod{2\pi} < \pi/4 < 1$. Therefore, $(1 - \frac{(x-y)^2}{12} \pmod{2\pi}) > 1/2$ and the second statement of the lemma follows. □

Of course, we can get much better bounds but $\sqrt{2}$ will suffice. Using the Lemma 20 one can now write the formula for the phase estimation algorithm.

Phase Estimation Formula

1. Let a positive integer $n > 2$ be given and $m = 2n$. The initial state is $|0\rangle |u\rangle$. The vector $|0\rangle$ is an m-qubit state and $|u\rangle$ is an s-qubit state. The formula expressing that the initialization of the system is $|0\rangle |u\rangle$ is,

   $$\mathbf{In} \equiv V[m, m+1, \ldots, m+s-1](P(t(t^m(\mathsf{b})_0, t^s(\mathsf{b})_0)) = 1) \tag{6.44}$$

   The operator V is arbitrary. It acts only on the last s qubits and rotates the vector represented by $t^s(\mathsf{b})_0$ to the vector $|u\rangle$. Given this vector and the basis represented by $t^{n+s}(\mathsf{b})$, the entries of a corresponding matrix can be written. It is not unique. Since we only need its effect on the vector $\pi(t^s(\mathsf{b})_0)$ we need only specify the first column of the matrix leaving the rest unspecified (as variables).

2. The fact that $|u\rangle$ is an eigenvector of U can be elegantly expressed by the formula

$$\mathbf{In} \wedge U[m, m+1, \ldots, m+s-1]\mathbf{In}.$$

This formula is true only in case both **In** *and* $U[m, m+1, \ldots, m+s-1]$**In** are true. That is, in the state ρ which satisfies the following. If we write the interpretation of $\mathtt{b}$ as the computational basis and the state $|u\rangle$ is the vector obtained by applying V to $t^n(\mathtt{b})_0$ then, in a measurement in the appropriate basis, the 'event' $|0\rangle|u\rangle$ is certain in the state ρ *and* $U \cdot \rho$. This can only happen if the application of U gives rise to a state which is a multiple of the original state ρ. Note that **In** is true for a pure state only.

3. Apply the Hadamard operator on each of the control qubits. This is the randomization operation. As in the case of Grover algorithm the operation may be expressed as

$$H[0]H[1]\cdots H[m-1].$$

The formula on which these operators act is yet to be specified. I will come to it last.

4. Apply next the CN-U^j operations to the target qubits. The corresponding operator in the language is

$$[\text{CN-}(U^\dagger)^j[m, m+1, \ldots, m+s-1]].$$

It is assumed that the operator U is given.

5. Now apply the inverse Fourier transform $F^\dagger$ to the control qubits. The operator is

$$F[0, \ldots, m-1].$$

In these two operations the operators in the formula are the inverse (= hermitian conjugate) operators since they will again be inverted when applied to the state vector.

6. Finally, we come to the formula to which the above operators are applied. As in the case of Grover algorithm this is the assertion that the algorithm produces the desired approximation with high probability. This is the trickiest part. First define the phase operator in the space $(\mathbb{C}^2)^{\otimes m}$ and for $M = 2^m$,

$$\mathbf{Ph}(jk) = \delta_{jk}\omega_M^j \tag{6.45}$$

The matrix for **Ph** is diagonal. In the computational basis it simply sends $|j\rangle \rightarrow e^{2\pi ij/M}|j\rangle$. The specification of phase estimation algorithm asserts that the output state of the control qubits is such that a measurement in the computational basis yields with high probability an eigenstate of **Ph** whose eigenvalue approximates the eigenvalue of the original operator U. That is, if we write the successive composition of all the operators in the algorithm as a circuit C (recall that a quantum circuit is also a unitary operator) then Ph composed with C approximates UV. One can express this as formula for circuit approximation but in rather cumbersome way. I follow a different approach. Let $S = 2^s$. In what follows it would be convenient to use bounded quantifiers over the positive integers. This can, of course, be avoided by replacing the bounded quantifiers by appropriate disjunction (for $\exists$) or conjunctions (for $\forall$). We now write the formula for phase estimation.

$$\begin{gathered}\mu(U,V,H,CN-U,F)\wedge\mathbf{In}\Rightarrow[U[m,m+1,\ldots,m+s-1]]\mathbf{In}\wedge\\ (\exists x\forall(i<S)(\sum_k U(ik)\overline{V(k0)}=x\overline{V(i0)}\wedge|x|^2=1)\wedge\\ \forall(j<M)\wedge(|x-\omega_N^j|<1\Rightarrow\\ [H[0]\cdots H[m-1]\text{CN-}(U^\dagger)^j[m,\ldots,m+s-1]]F(P(t(t^m(\mathsf{b}))_j,\top)>1-\varepsilon)))\end{gathered} \tag{6.46}$$

Let us take a close look at the formula. First, consider the subset of vectors defined by the subformula $\mathbf{In} \wedge [U[m, m+1, \ldots, m+s-1]]\mathbf{In}$ for some interpretation π of the basis variable $\mathtt{b}$. Let $\pi(t^n(\mathtt{b})_i) = |\alpha_i\rangle$. Clearly **In** is true in a state ρ iff $\mathtt{Tr}(\rho\,|\alpha_0\rangle\langle\alpha_0|) = 1$. This is possible iff $\rho = |\alpha_0\rangle\langle\alpha_0| \otimes V\,|\alpha_0\rangle\langle\alpha_0|\,V^\dagger$. Similarly, $[U[m, m+1, \ldots, m+s-1]]\mathbf{In}$ is true in the state $|\alpha_0\rangle\langle\alpha_0| \otimes UV^\dagger\,|\alpha_0\rangle\langle\alpha_0|\,V^\dagger U^\dagger$. These two states must coincide which is possible iff $UV^\dagger\,|\alpha_0\rangle = xV^\dagger\,|\alpha_0\rangle$ for some x such that $|x| = 1$. That is, $V^\dagger\,|\alpha_0\rangle$ is an eigenvector of U with eigenvalue x. Then, $x = e^{2\pi i\phi}$ for some real ϕ. The formula $\exists y \forall (j < M)\omega_N(y) \wedge (|x - y^j| < 1/N \Rightarrow [H[0] \cdots H[m-1]\text{CN-}(U^\dagger)^j[m, \ldots, m+s-1]]F(P(t(t^m(\mathtt{b}))_j, \top) > 1 - \varepsilon))$ then asserts that there is a primitive Nth root of unity ω_N such that if ω_N^j is close to x then after the application of the inverse Fourier transform $F^\dagger$ with high probability ($> 1 - \varepsilon$) we obtain the state state $\pi(t^m(\mathtt{b}))_j$ when we measure the first m bits. In other words, j/N is an estimate of the phase ϕ.

The phase estimation algorithm described above is the basis of important algorithms for two important problems: finding the order of an element in a commutative group and the related factorization of a number (the famous Shor algorithm) [NC01]. I will not deal with these in the present work but conclude this section with the remark that it is possible to implement the controlled-U^j operations efficiently.

Quantum teleportation

In the examples considered above I have not used the measurement operators M_X and S_{X_i}. They can be used to express circuits in which measurements, not unitary gates, are used as the primitive unit of computation[Nie03]. They are also necessary in the discussion of quantum protocols [NC01]. For example, in the teleportation protocol Alice and Bob share a qubit each of an entangled pair. Then Alice applies a unitary transformation to her share of the pair and another qubit in unknown state and then performs a measurement. Due to entanglement Bob's qubit gets affected. By applying

unitary transformations depending on Alice's measurement outcome Bob can change the state of his qubit to that of the unknown one. Since the qubits are entangled Bob's qubit gets affected by the measurement. Alice knows the outcome of the measurement. The corresponding formula is given below. Let $\mathbf{A} \stackrel{\text{def}}{=} H[1]C[1,2]C[2,3]H[2](P(t^3(\mathsf{b})_0)) = 1$ and $\mathbf{B} \stackrel{\text{def}}{=} P(t(\top, \top, \mathsf{b}_0)) = 1$.

$$\Phi \equiv [U](P(t(\mathsf{b}_0, \top, \top)) = 1) \Rightarrow (S_{00}\mathbf{A} \Rightarrow [U]\mathbf{B}) \wedge (S_{01}\mathbf{A} \Rightarrow [U][X]\mathbf{B}) \wedge (S_{10}\mathbf{A} \Rightarrow [U][Z]\mathbf{B}) \wedge (S_{10}\mathbf{A} \Rightarrow [U][ZX]\mathbf{B}) \tag{6.47}$$

This formula is quite easy to understand actually. Alice has qubits 1 and 2 and Bob has the third. Alice and Bob start with the entangled pair 2 and 3, which is achieved by applying $C[2,3]H[2]$. Then Alice applies $H[1]C[1,2]$ and the four alternatives correspond to the four outcomes of a measurement in the computational basis. after running the protocol the state of the third qubit (Bob's) is identical to that of the unknown qubit they started with. Although this formula is valid it does not capture the actual knowledge or information that the agents have at each stage. Some preliminary work in this direction may be found in [MP03a] where knowledge and temporal operators are introduced. Let $\Phi(V)$ be the formula obtained from Φ by replacing $H[1]C[1,2]$ by V a "unitary" variable. We may ask whether $\exists V \Phi(V[1,2])$. That is, whether there exist unitary operation on Alice's qubit such that "teleportation" takes place. In fact it can be verified quite easily by a simple procedure. We may formulate and verify more complicated questions. The point is, since the theory is decidable it is not too hard to devise algorithms to answer such questions. Alternatively, it is in principle possible to utilise theorem provers like PVS or Isabelle using the axiomatization in the preceding chapters.

Chapter 7

Conclusion

In this chapter we make some concluding remarks and discuss future prospects. In the preceding chapters we presented logics for dealing with a variety of problems in quantum theory, especially, quantum computation and information. The examples of chapters 4 and 5 and the applications in the chapter 6 show that the logics can be used to represent and analyze important concepts in quantum theory and nontrivial algorithms and protocols in quantum computing. Let us summarize the salient features of the work.

1. A first order theory $\mathbb{RC}$, incorporating the theory of real closed fields *and* its algebraic closure. Proof that it is a complete theory allowing elimination of quantifiers.

2. Syntax and semantics for the language $\mathcal{L}_n(P, \mathbf{m})$, extending $\mathbb{RC}$. It is language for reasoning about probabilities in quantum systems of dimension n,

3. The axiomatization of $\mathcal{L}_n(P, \mathbf{m})$ and proof of some important features of the axiomatization: soundness, completeness, and decidability.

4. Derivation of some complexity results and bounds.

5. Syntax and semantics of the language $\mathcal{L}(P, t, M, S, \mathbf{U})$, which extend the languages $\mathcal{L}_n(P, \mathbf{m})$ by incorporating tensor product and measure-

ments. The semantics extended to the general mixed states. Examples of important sets of quantum states *definable* in the language.

6. Axiomatization of $\mathcal{L}(P, t, M, S, \mathbf{U})$ and proof of soundness, completeness, and decidability of the resulting theory.

7. Development of efficient algorithms for satisfiability. derivation of some complexity bounds. Characterization of quantum circuits that can be efficiently simulated. I feel that this is an interesting development giving a descriptive characterization of the quantum circuits whcich can be efficiently simulated.

8. Syntax and semantics of a related language $\mathcal{L}_2(P, t, M, S, \mathbf{U})$ with application to quantum circuits.

9. Applications to important quantum algorithms and protocols: Grover search algorithm, Shor algorithm and the quantum teleportation protocol. We may also mention that certain questions about the synthesis of quantum algorithms have been demonstrated to be decidable.

Some of the problems which are *not* addressed are discussed below. I aim to address these very interesting issues in future.

1. Expressive power of the language(s): How expressive is the language $\mathcal{L}(P, t, M, S, \mathbf{U})$? I conjecture that it is as expressive as the standard Hilbert space language for quantum systems whose dimensions are bounded in the sense that they do not have concepts like "in all dimensions n". Since we do not have quantification over *integers* which are treated as constants.

2. We may introduce a new *sort* of variables for natural numbers and include some axiomatization of the latter. By Godel's famous theorem on the incompleteness of arithmetic the rsulting theory is no longer decidable. However, one could look for decidable fragments of the

logic, that is, class of formulas which are decidable. This could help us in our search for new quantum algorithms.

3. Independence of the axioms of the theories of $\mathcal{L}_n(P, \mathbf{m})$ and $\mathcal{L}(P, t, M, S, \mathbf{U})$.

4. Practical implementation of the algorithms. Although simple fragments of the logics have been implemented in Matlab efficient implementation of the full language poses interesting challenges.

5. We have focussed on quantum systems composed of subsystems of fixed dimension ("qunits"). One could generalize to arbitrary *finite* dimensional subsystems. Such a language is possible and in fact is discussed in Chapter 5 but the semantics and axiomatization becomes more involved mainly because the ordering factors in a tensor product is a nontrivial issue. Moreover, such languages dealing with "all" finite dimensional vector spaces is more naturally interpreted in a categorical setting, for example, the category of finite dimensional Hilbert spaces which is a compcat closed category [AC04b].

6. One could also generalize in another direction, namely, quantum systems with infinite dimensional Hilbert space. A finite dimensional quantum system is an idealization and really does not exist in nature. The semantics of infinite dimensional systems is in some sense simpler since we do not have to keep track of dimensions. But the axiomatization poses difficult challenges. One possibility is to consider languages allowing countably infinite conjunctions and disjunctions (infinitary logics).

7. It is possible to extend the languages to deal with quantum cryptographic protocols [NC01] and quantum games [EW00].

We have already made some progress in regard to the last two items. As mentioned earlier, we have implemented some algorithms for simple quantum circuits in MatLab. Specifically, we have "verified" the probabilistic

output of some quantum circuits in small dimensions. The results are encouraging but it is still very early to report. There are also problems of optimization which have not been addressed. The algorithms presented here are efficient but not necessarily optimal.

We have made some progress on the problems of dealing with complex quantum cryptographic protocols. In this regard we introduced the notion of "quantum knowledge" [MP03a]. However, in applying these notions to concrete protocols one encounters problems of a physical and philosophical nature. Issues like-what is a reasonable model for an adversary? are quantum probabilities objective(as limits of frequency of occurrence) or *subjective*? - come up. These issues seem to depend on the context. We have made some progress in this direction. It appears useful to extend the logics with temporal modal operators and knowledge seems to play an important role.

Bibliography

[AC04a] S. Abramsky and B. Coecke. A categorical semantics of quantum protocols. Research Report RR-04-02, Oxford University Computing Laboratory, 2004.

[AC04b] S. Abramsky and B. Coecke. A categorical semantics of quantum protocols. In *Proceedings of the 19th IEEE conference on Logic in Computer Science (LiCS'04)*. IEEE Computer Science Press, 2004.

[AH94] M. Abadi and J.Y. Halpern. Decidability and expressiveness for first-order logics of probability. *Information and Computation*, 112(1):1–36, 1994.

[AR02] M. Abadi and P. Rogaway. Reconciling two views of cryptography. *Journal of Cryptology*, 5:103–127, 2002.

[Bac90] F. Bacchus. *Representing and reasoning with probabilistic Knowledge*. MIT Press, Cambridge, Mass., 1990.

[BAN90] M. Burrows, M. Abadi, and R. Needham. A logic for authentication. *ACM Trans. Comp. Sys.*, 8:18–36, 1990.

[BCS01] S. Bettelli, T. Calarco, and L. Serafini. Towards an architecture for quantum programming. arXiv:cs.PL/0103009 v2, November 2001.

[BKR86] M. Ben-Or, D. Kozen, and J. H. Reif. The complexity of elementary algebra and geometry. *Journal of Computer and System Sciences*, 32(1):251–264, 1986.

[BKR94] M. Ben-Or, D. Kozen, and J. H. Reif. A logic for reasoning about time and probability. *Formal aspects of computing*, 6:513–535, 1994.

[BLM91] P. Busch, P. J. Lathi, and P. Mittelstaedt. *The Quantum theory of measurement.* Springer, Berlin, 1991.

[BPR03] S. Basu, R. Pollack, and M-F. Roy. *Algorithms in real algebraic geometry.* Springer, 2003.

[BS69] J.L. Bell and A. B. Slomsen. *Models and ultraproducts.* North-Holland, Amsterdam, 1969.

[BS04] A. Baltag and S. Smets. A logic for quantum programs. In *Proc. of QPL 2004*, pages 39–56, 2004.

[Bub97] J. Bub. *Interpreting the quantum world.* Cambridge University Press, Cambridge, 1997.

[BV97] E. Bernstein and U. V. Vazirani. Quantum complexity theory. *SIAM. J. Computing*, 26(5):1411–1473, 1997.

[BvN36] G. Birkhoff and J. von Neumann. The logic of quantum mechanics. *Annals of Mathematics*, 37:823–843, 1936.

[Can88] J. F. Canny. Some algebraic and geometric computations in PSPACE. In *Proc. 20th ACM Symp. on Theory of Computing*, pages 460–467, 1988.

[Car50] R. Carnap. *Logical foundations of probability.* University of Chicago Press, Chicago, 1950.

[CBG+92] E. M. Clarke, J. R. Budch, O. Grumberg, D. E. Long, and K. L. McMillan. Automatic verification of sequential circuit design. *Phil. Trans. of R. Soc. Lond. A*, 339:105–120, 1992.

[Chu56] A. Church. *Inroduction to mathematical logic.* Princeton University Press, Princeton, 1956.

[Dal94] D. van Dalen. *Logic and structure.* Springer-Verlag, Berlin, 1994.

[dE76] B. d' Espagnat. *Conceptual foundations of quantum mechanics.* W. A. Benjamin, Reading, Mass., 1976.

[Dru77] M. Drummet. *Elements of intuitionism.* Clarendon Press, Oxford, 1977.

[Ebb96] H-D. Ebbinghaus. *Mathematical logic.* Springer, New York, 1996.

[EW00] A. Eisert and M. Wilkens. Quantum games. *J. Mod. Opt.*, 47:2543, 2000.

[Fel57] W. Feller. *Introduction to probability theory and its applications, Vol. 1.* Wiley, New York, 1957.

[Fey63] R. P. Feynman. *The Feynman lectures on physics III.* Addison-Wesley, Reading, Mass., 1963.

[FHM90] R. Fagin, J. Y. Halpern, and N. Megiddo. A logic for reasoning about probabilities. *Information and Computation*, 87(1/2):78–128, 1990.

[FHMV95] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning about knowledge.* MIT Press, Cambridge, Mass., 1995.

[FR78] D. J. Foulis and C. H. Randall. Manuals, morphisms, and quantum mechanics. In A. R. Marlow, editor, *Mathematical foundations of quantum theory.* Academic Press, 1978.

[FT82] E. Fredkin and T. Toffoli. Conservative logic. *International Journal of Theoretical Physics*, 21(3/4), 1982.

[Gol92] R. Goldblatt. *Logics of time and computation.* CSLI, Stanford, 1992.

[Got99] D. Gottesman. The heisenberg representation of quantum computers. In S. P. Corney, R. Delbourgo, and P. D. Jarvis, editors, *Group 22: Proc. of the XXII Int. Conf. on Group Theoretical Methods in Physics*, Cambridge, MA, 1999. International Press.

[Got03] K. Gottfried. *Quantum mechanics: fundamentals.* Springer, New York, 2003.

[Gre75] W. Greub. *Linear algebra.* Springer-Verlag, New York, 1975.

[Gro96] L. Grover. A fast quantum mechanical algorithm for database search. In *Proc. 28th. annual ACM symposium on Theory of computing*, pages 212–210, New York, 1996. ACM.

[Hal50] P. R. Halmos. *Measure Theory.* Springer, New York, 1950.

[Har79] D. Harel. *First-order dynamic logic, LNCS 68.* Springer-Verlag, Berlin, 1979.

[HJ90] R. A. Horn and C. R. Johnson. *Matrix analysis.* Cambridge, New York, 1990.

[HJ91] R. A. Horn and C. R. Johnson. *Topics in matrix analysis.* Cambridge, New York, 1991.

[HK00] D. Harel and D. Kpzen. *Dynamic logic.* MIT Press, Cambridge, Mass., 2000.

[Hoa85] C.A.R. Hoare. *Communicating sequential processes.* Prentice-Hall Int., Englewood Cliffs, NJ, 1985.

[Hod97] W. Hodge. *A shorter model theory.* Cambridge, New York, 1997.

[HT93] J. Y. Halpern and M. R. Tuttle. Knowledge, probability, and adversaries. *Journal of the ACM*, 1993.

[HU79] J. E. Hopcroft and J. D. Ullman. *Introduction to automata theory, languages and computation.* Addison-Wesely, Reading, Mass., 1979.

[Kar02] N. Karmakar. A new polynomial time algorithm for linear programming. *Combinatorica*, 4:373–395, 2002.

[KG99] C. Kern and M. R. Greenstreet. Formal verification in hardware design: a survey. *ACM Trans. on Design and Automation of Electronic Systems*, 4:123–193, 1999.

[Kha79] L. G. Khachiyan. A polynomial algorithm in linear programming. *Soviet Mathematics Doklady*, 1979.

[Kni96] E. H. Knill. Conventions for quantum pseudocode. LANL report LAUR-96-2724, 1996.

[Kol56] A. N. Kolomogorov. *Foundations of probability theory.* Chelsea, New York, 1956.

[Kri59] S. Kripke. A completeness theorem in modal logic. *J. Symbolic Logic*, 24:1–14, 1959.

[Kri63] S. Kripke. Semantic analysis of modal logic. *Zeit. Math. Logic. Grund. Math.*, 9:67–96, 1963.

[Low96] G. Lowe. Breaking and fixing the needham-scroeder public-key protocol using fdr. *Softwarae-Concepts and Tools*, 17:93–102, 1996.

[Mac71] S. MacLane. *Categories for the working mathematician.* Springer-Verlag, New York, 1971.

[MM92] M. Marcus and H. Minc. *A survey of matrix theory and matrix inequalities.* Dover, New York, 1992.

[MP03a] R. van der Meyden and M. Patra. Knowledge in quantum systems. In *Theoretical aspects of knowledge and rationality*, Bloomington, 2003. ACM.

[MP03b] R. van der Meyden and M. Patra. A logic for probability in quantum systems. In *Proc. Computer Science Logic and 8th Kurt Gdel Colloquium*, Vienna, 2003. Springer-Verlag.

[MS04a] P. Mateus and A. Serandas. Exogeneoous quantum logic. In *Proc. of CombLog04*, pages 141–149, 2004.

[MS04b] P. Mateus and A. Serandas. Reasoning about quantum systems. In *Logics in Artificial Intelligence JELIA04*, pages 239–251. Springer-Verlag, 2004.

[NC01] M. A. Nielsen and I. L. Chuang. *Quantum computation and information.* CUP, 2001.

[Nie03] M. Nielsen. Universal quantum computation using only projective measurement, quantum memory, and preparation of the 0 state. *Physics Lett. A*, 308(2-3):96–100, 2003.

[Nil86] N. Nilsson. Probabilistic logic. *Artificial Intelligence*, 28:71–87, 1986.

[Ö98] B. Ömer. A procedural formalism for quantum computation. Master's thesis, Dept. of Theoretical Physics, Technical Univ. of Vienna, 1998. (http://www.tph.tuwien.ac.at/ oemer/qcl.html).

[Pap94] C. M. Papadimitriou. *Computational Complexity.* Addison-Wesely, Reading, Mass., 1994.

[Pat05] M. Patra. A logic for quantum circuits and protocols. In *Theoretical Aspects of Computing*, volume 3722 of *Lecture Notes in Compter Science*, page 424. Springer-Verlag, 2005.

[Pat06] M. K. Patra. Projective invariant measures and approximation of quantum circuits. quant-ph/0604104, 2006.

[Per95] A. Peres. *Quantum Theory: concepts and methods.* Kluwer Academic Publishers, 1995.

[Pir76] C. Piron. *Foundations of quantum physics.* W. A. Benjamin, Reading, 1976.

[RS00] J. P. Rawling and S. A. Selesnick. Orthologic and quantum logic: models and computational elements. *Journal of the ACM*, 47(4):721–751, 2000.

[Sel04] P. Selinger. Towards a quantum programming language. *Math. Struct. in Comp. Science*, 14:527, 2004.

[Sel05] P. Selinger. Dagger compact closed categories and completely positive maps. In *3rd. Int. Symp. on Quantum Progamming Languages(QPL05)*, 2005.

[Sho67] J. R. Shoenfield. *Mathematical Logic.* Addison-Wesely, 1967.

[Sho94] P. W. Shor. Algorithms for quantum computation, discrete logarithms and factoring. In *Proc. 35th. Symp. on Foundations of computer science*, Los Alamitos, 1994. IEEE.

[Smu96] R. Smullyan. *Theory of formal systems.* Princeton University Press, Princeton, N.J., 1996.

[SP00] J. W. Sanders and P.Zulliani. Quantum programming. In *Mathematics of program construction, LNCS 1837*, pages 80–99. Springer, 2000.

[Sto36] M. H. Stone. The theory of representations of boolean algebras. *Transactions of American Mathematical Society*, 40:37–111, 1936.

[Tar56] A. Tarski. *Logic, semantics, metamathematics:papers from 1923 to 1938.* Clarendon Press, Oxford, 1956.

[Vol98] H. Vollmer. *Introduction to circuit complexity.* Springer, Berlin, 1998.

[Wae53] B. L. van der Waerden. *Modern Algebra.* Ungar, New York, 1953.

[Wil91] D. Williams. *Probability with martingales.* Cambridge University Press, Cambridge, 1991.

[Zil03] B. Zilber. Complex roots of unity on the real plane. http://www.maths.ox.ac.uk/ zilber/complexrootsonrealplane.dvi, 2003.

Index