1
Making Computation Faster and Communication Secure: Quantum Solution

The main stumbling block of quantum information, computation, and to a lesser extent, communication is the lack of a definite hardware. We still do not know whether we are going to compute by ions, or by solid state systems, or by photons, or by quantum electrodynamics, or by superconducting charges … Yet, there are already formalisms, algorithms and theories on quantum "all that." But, didn't we have the same "problem" when we started to compute on classical computers in the forties? What was the hardware then? "Human computers," mechanical gadgets, electromechanical drums, tube-calculators, … And before that, we already had classical formalisms and algorithms and theories. Let us start with a classical story which will help us understand that interplay of software and hardware so that we can better apply it to qubits later on.

1.1
Turing Machine: a Real Machine or …

The *Turing machine* is not a computer and it cannot serve us to build a useful gadget. Yet, there are so many Turing machine applets on the web to help math students to prepare their exams. So, why can't we turn "the machine" into a realistic computing device? The answer is both simple and long.

Alan Turing graduated in mathematics from King's College, Cambridge in 1934 and was elected a fellow there the next year thanks to a paper in which he designed his famous machine. The paper gave a solution to a problem on which the famous mathematician Alonzo Church at Princeton University was also working at the time. So, in 1938, Turing went to Princeton to study under Church and received his Ph.D. from Princeton in 1938.

A few months later, Turing returned to England and started to work part-time at the *Government Code and Cypher School* on German encryption systems. A year later, he joined the wartime station of the school, now famous, *Bletchley Park*. There he soon became a main designer of electromechanical decrypting machines – named *Bombes*, after their predecessor, a Polish *Bomba*. They helped the British to decipher many German messages and gain advantages in many actions and battles.

Companion to Quantum Computation and Communication, First Edition. M. Pavičić.

The story is a paradigm for today's university researchers who are expected to find an application for their research as soon as possible and sign as many contracts with industry as possible. Then, there were many details in Turing's approach to work and people that attract interest of the media today – also a valuable commodity for today's university researchers. For example, he induced his colleagues to see his design of the *Bombe* as follows: "[in its design] he had the idea that you could use, in effect, a theorem in logic which sounds to the untrained ear rather absurd; namely, that from a contradiction, you can deduce everything." This is our main clue and we will come back to it in Section 1.2.

In July 1942, Turing devised a new deciphering technique named *Turingery* against a new German *secret writer*, code-named *Fish* and recommended some of his coworkers for the project of building the *Colossus* computer, the world's first programmable digital electronic computer, which eventually replaced simpler prior machines and whose superior speed allowed the brute-force decryption techniques to be usefully applied to the daily-changing ciphers. Turing himself did not take part in designing the Colossus, but left for America to work on US *Bombes* ($3 \times 2.1 \times 0.61\,m^3$; 2.5 t; 120 of them were made till 1944). When he returned to England, he accepted a position as a general consultant for cryptanalysis at the Bletchley Park. At that time, he also designed a machine for a secure voice communication which has never been put into production.

Details of his work on cryptography during the war remained a secret for many years after it. Eventually, he dropped his cooperation with industry and returned to that lofty realm of science that offers a different history. But, let us first go back to his machine and examine its "simple history." We will learn that the machine is not at all a real machine, but a mathematical procedure.

1.2
. . . a Mathematical Procedure

In the thirties, most leading mathematicians in the field of symbolic logic and related algebras were involved in solving a problem of *decidability* – whether one can decide that a statement (formula, theorem) in a formal system is valid (holds) or not. That a system is *decidable* means that each formula in it is either provable or refutable. That a proof of a formula (*predicate* of the formula) is *effectively decidable* means that for every tree of formulae starting from the axioms of the system we can tell whether it is a proof of the formula, that is, whether it is *recursive*. For *functions* – formulae that depend on arguments – we then say that they are effectively calculable functions. If there is a system of equations that define a function recursively, then the function is *general recursive*.

One of the leading mathematicians who was engaged in these problems and who defined the notion of the general recursiveness of a function was Alonso Church (see above) who also formulated his famous *Church thesis*:

Thesis 1 Church 1936

Every effectively calculable function (effectively decidable predicate) is generally recursive.

An interpretation of the thesis is the following one. If we know a recursive procedure for obtaining a function, then, of course, the function is effectively calculable because the procedure itself is the proof that the function is valid. The converse is not obvious. That is, if we know how to decide whether there is a proof that a function is valid, we need not ever be able to find a recursive procedure for obtaining the function or even need not know whether such a procedure exists at all. The Church thesis is a conjecture that it always exists. It has not been proved so far, but there are overwhelming evidences that it is correct and, of course, it has never been disproved.

So, how do we prove that a system is decidable and that all its functions are *computable*, that is, generally recursive? Well, we have to find a procedure which would prove that every function from the system is effectively calculable or every predicate effectively decidable, and then we search for a generally recursive algorithm[1)] for computing the functions. Or, better still, we first find a generally recursive algorithm and constructively prove that all functions from a systems are computable. The effective decidability and calculability then follow as consequences of the computability of the system.

The latter task is exactly what Church's general recursiveness (1933), Kleene's λ-*definability* (1935), Gödel's *reckonability* (1936), Turing's machine (1936/1937) and Post's *canonical* and *normal systems* (Emil Post; 1936; independent discovery) are about and this is why theoreticians like them so much. They, however, prefer the Turing machine over others because it is more intuitive and easier to handle.

The final "output" of any of these procedures is the same. They tell us which theory is decidable and which is not. Then, Church's thesis tells us to assume that any decidable theory is computable and that any undecidable one is not.

- Decidable theories are, for example:
  - *Presburger arithmetic* of the integers with equality and addition (Mojżesz Presburger, 1929);
  - Boolean algebras (Alfred Tarski, 1949);
  - Propositional two-valued classical logic;
- Undecidable theories are, among so many others:
  - *Peano arithmetic* with equality, addition, and multiplication (Kurt Gödel, 1932);
  - Predicate logic including metalogic of propositional calculus;
  - Every consistent formal system that contains a certain amount of finitary number theory there exists undecidable arithmetic propositions and the consistency of any such system cannot be proved in the system (many authors, including A. Turing, from the early thirties until the mid-sixties);

1) Algorithm is a computational method of getting a solution to a given problem in the sense of getting correct outputs from given inputs.

– Functions obeying *Rice's theorem*: Only trivial properties of programs are algorithmically decidable. For any nontrivial property of partial functions, the question of whether a given algorithm computes a partial function with this property is undecidable (H.G. Rice, 1953).

Thus, any of the above procedures, can be used to prove that a 0-1 Boolean algebra or equivalently, a two-valued (*true*, $\top$; *false*, $\bot$) propositional classical logic is decidable and therefore computable. Peano arithmetic and any more complicated systems using real numbers are not. Hence, our standard digital binary universal computer is actually the only one we can build without running into a contradiction sooner or later. This gives insight into Turing's colleagues' remark we cited above: "[Turing] had the idea that [in dealing with the Bombe computer] you could use, in effect, a theorem [which states that] from a contradiction, you can deduce everything." By invoking this well-known *Ex contradictione quodlibet* logical principle, they referred to Turing's checking whether particular code systems were consistent or not.

1.3 Faster Super-Turing Computation

Thus, a Boolean digital binary system is a "safe" ideal algebra for building a universal computer because it is decidable and consistent. But, does that mean that undecidable and inconsistent systems reviewed in the previous section cannot be used for computing and building computers?

When Frege, Whitehead, Russell, and Hilbert attempted to develop logic foundations of mathematics, they stumbled on paradoxes of self-reference such as the famous *Liar paradox*, on inconsistencies. Their attempts to go around such inconsistencies failed, but in the eighties, theoreticians revised such apparently inconsistent theories and saw a possibility to revive the Hilbert Program of consistently building mathematics from its logical foundations.

At the time, David Hilbert gave up his program because Gödel and others proved that the consistency of arithmetics cannot be proved within arithmetics itself. Though recently, the authors, such as R.K. Meyer and C. Mortensen, started from a widely accepted assumption that all negative results would not endanger the correctness of numerical calculations that have been carried before and since the beginning of the twentieth century, and started a new program called *Inconsistent Mathematics*. Originally, it started with a plausible claim that mathematics could be given a trouble-free interpretation if we recognized that mathematics is *not* its foundation.

We shall not elaborate on the foundational and conditional aspect of "inconsistent mathematics" any further. However, we need to discuss several nonstandard approaches in computation science and underlying formalisms, algebras, and even logic to see how it all can be applied in reaching our goal of speeding up computation – both classical and quantum.

Hilbert's program considered whether we can write down algorithms for an automated computation of any expression or carrying out any proof in any mathematical theory. Such algorithms can traditionally and rigorously be obtained only for Boolean algebras. The Church–Kleene–Gödel–Turing–Post proof of this result we can – in *2012 Year of Alan Turing* – express as follows. "A Turing machine calculating any of the 0-1 Boolean algebra problems will halt after a finite number of steps." But, what about standard arithmetics? Theory of rational or real numbers? Can there be *super-Turing* machines that are faster then the Turing ones? We shall see that there are *quantum-Turing* machines that are exponentially faster than the Turing ones and therefore a kind of super-Turing machine. Are there *classical* super-Turing machines?

To answer this question, we first have to answer several other questions.

1. Let us start with our safe "digital 0-1 algebra." Is there a logic behind it which is more general than the usual two-valued (true-false) one? Can it be implemented in a binary computer? Is it important for our purpose of devising a fast quantum computer to find a "quantum logic" behind or "under" the Hilbert space formalism we will use?
2. Can we devise computers that can handle, for example, real numbers directly, analogously to how we humans handle them on paper, that is, without any need to first digitalize them? Can analog computers be universal? Are they faster? What are their limits? Do quantum computers have a theoretical speed limit?
3. Are there other such classical or optical computers that can compute the same problems quantum would-be computers could solve? Can we achieve similar exponential speed-up of computation with such computers? Can we realistically use them to carry out super-Turing computation?
4. How much energy does the computation itself require? Can we reduce heat and energy dissipated in calculation per operation and per calculated bit? Heat is a main problem when we want to pack transistors of ever reduced size. The closer the transistors are to each other, the faster the computation. Are there processors that dissipate orders of magnitude less heat than today's standard Pentiums? What is the theoretical minimum we cannot go beyond? How do classical computers that dissipate a minimum of energy look like? Are quantum computers better?

We shall answer all of these questions in the next sections.

1.4 Digital Computers Do Not Run on Logic

It is often taken for granted that 0-1 Boolean algebra, Boolean logic, and classical propositional logic are all different names for one and the same algebra: 0-1 Boolean algebra. However, that is not the case.

If we browse through books on computation and computer organization, we shall soon notice that these books hardly ever mention *logic*. This is because the theory of classical logic contains various methods of manipulating the propositions and different possible models (semantics). The authors know that almost universally accepted valuation of the logical propositions is a 0-1 bivaluation, that the corresponding semantics is represented by truth tables, and that the only lattice model that corresponds to this bivaluation is a 0-1 Boolean algebra – a Boolean algebra is a distributive lattice. So, they all take for granted that it is OK to deal with 0-1 Bolean algebra instead. Often, Boolean algebra is called *Boolean logic*. Let us take a more detailed look.

Algebra is a mathematical structure (most often a *vector space*, for example, a lattice, a Boolean algebra, a Euclidean space, a phase space, a Hilbert space, ...) over a set of elements (most often a *field*, for example, real or complex numbers, ...). Loosely speaking, algebras describe relationships between things that might vary over time. What interests us the most are algebras that can or cannot be implemented in a computer and algebras that can serve as models of logic.

Thus, although a general Boolean algebra is a vector space over a field or a ring (e.g., set $\{0, 1\}$, for which division is not defined), we shall start with its simplest 0-1 (digital, two-valued) form and define it at first as the set $\{0, 1\}$ on which operations *conjunction* ($\cap$), *disjunction* ($\cup$), and *complement* ($'$) are defined as in Figure 1.1.

Operations in logics are defined equivalently, only it is taken that 1 means *true* and 0 *false*. These values are called *truth values* and are denoted as $\top$ and $\bot$, respectively. The tables from Figure 1.1 are called *truth tables*.

What is characteristic of both $\{0, 1\}$ Boolean algebra rules and classical logic truth tables is that by starting from the definite initial values for all variables, we will define values of all intermediary combinations of the values until we reach a final result of our calculation as shown in Table 1.1. When the expressions become huge, evaluation of the intermediary expressions take exponentially more time. And, these intermediary expressions are exactly what quantum computers should get rid of. How?

Both classical and quantum computers require the so-called *logic gates*, which we can understand as switches or ports through which electrons, photons, ..., information flow. Schematics of some classical ones are shown in Figure 1.2 where, for example, XOR is the electrical current equivalent of the negation of the logical biconditional "$\leftrightarrow$:" It represents the following electric current behavior in a tran-

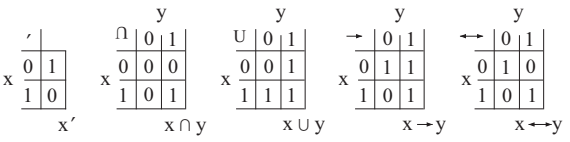


Figure 1.1 Boolean and logical binary operations. Boolean operations $'$, $\cap$, $\cup$, $\rightarrow$ and $\leftrightarrow$ we denote in logic as $\neg$, $\wedge$, $\vee$, $\rightarrow$, and $\leftrightarrow$.

Table 1.1 Logical truth table. The more complicated the expression, the more intermediary valuations we have to make to evaluate the final expression. The complexity of its evaluation grows exponentially in time with its size.[2)]

a	b	c	$a \wedge b$	$\bar{a}$	$\bar{c}$	$\bar{a} \wedge \bar{c}$	$(a \wedge b) \vee (\bar{a} \wedge \bar{c})$	$b \equiv c$	$(\bar{a} \wedge \bar{c}) \supset (b \equiv c)$
$\top$	$\top$	$\top$	$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\top$
$\bot$	$\top$	$\top$	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\top$
$\top$	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$
$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$
$\top$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\bot$
$\bot$	$\top$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$	$\bot$	$\bot$
$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$	$\top$	$\top$
$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$

sistor: "The output is low when both inputs A and B are high and when neither A nor B is high."

That low and high voltage, 0 and 1, a binary digit, a unit of information, a *bit* for short, is what makes up every number, word, program, pixel, sound, image in a classical computer. When we want to represent a number in a binary form, we soon realize how many physical hardware elements we need to implement any input. For example, eight-bit binary forms of the first 256 nonnegative integers are 00000000, 0000001, ..., 11111111. To carry out the addition of these digits (other operations can be reduced to addition), a classical digital computer uses an eight bit binary adder. It consists of eight full adders, each full adder consists of two half adders and an OR gate, and each half adder of an XNOR (negation of XOR) and an AND gate which altogether makes 40 gates (see Figure 1.71).

Thus, a computation of problems or manipulation of images whose complexities grow exponentially require an exponential increase of the number of transistors. Today, the number of transistors (gates) in a classical processor already reached 5 billions and still a quantum processor with only several hundred quantum gates would outdo it. The reason is that quantum gates can work with an arbitrary continuous combination, we call it a *superposition*, of elementary states – *quantum*

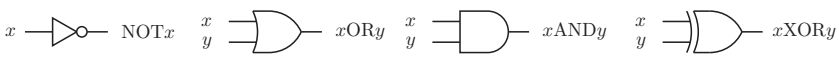


Figure 1.2 Logic gate symbols and operations. Notation used for operations: $\overline{x}$ (NOT x), $x + y$ (x OR y), xy (x AND y), $x \oplus y$ (x XOR y) (XOR is addition (+) modulo 2: $1 \oplus 1 = 0$; also $A \oplus B = (A + B)\overline{AB} = A\overline{B} + \overline{A}B$). See Figure 1.16.

2) To that, we can add the *satiability problem* (SAT) and the *isomorph-free generation of graphs and hypergraphs*. SAT problem consists in verifying whether Boolean expressions like that one shown in Table 1.1 are satisfied, that is, true, that is, equal to 1 in a Boolean algebra. SAT belongs to EXPTIME and that will help us understand why the factoring number problem computed in a digital computer belongs to EXPTIME too. Graphs and hypergraphs can be used to map nonlinear equations into hypergraphs and filter out equations that have solutions. They also belong to EXPTIME. We shall use them later on to generate the so-called Kochen–Specker sets.

bits, *qubits*. Yet, a working quantum computer still does not exist and therefore we should first explore whether we can exponentially speed-up computation in some other way.

The first option is to see whether we can implement our classical logic into some other kind of hardware instead of a binary computer. Since the models of both classical and quantum logic that we use to implement logics into computers are lattices (a distributive lattice (Boolean algebra) and a Hilbert lattice (underlying the Hilbert space)), the option boils down to a question of whether there are other lattices that can model classical logic. The answer is positive.

Let us look at the lattice shown in Figure 1.3. (A *lattice* is a partially ordered set with unique least upper and greatest lower bounds.) Here, a valuation of the proposition: "A particle is detected at position (4,3,5)" can be not only 1 (true) and 0 (false), but also a, b, a', or b'.

At the first glance, this seems acceptable as a kind of multivalued logic. One is tempted to consider a proposition to which value 1 is assigned, as an "always true" one; then, ones with values a and b as, say 66 and 33%, respectively; and a 0 one as "always false." But, we soon realize that such and actually any numerical valuation is impossible. To see this, it suffices to recognize that any numerical valuation would make a and b comparable with $\bar{a}$ and $\bar{b}$ and that is in contradiction with the main property of o6 lattice – that a and b are incomparable with $\bar{a}$ and $\bar{b}$.

That also means that one cannot construct a simple chip for o6 where we would just have different voltages for 0, a, b, and 1 as in a multivalued logic (different voltages are comparable to each other and that is precluded in o6). Actually, such a chip should have a nonclassical, nonnumerical ports and that is directly correlated with the above property that we must be able to represent propositions that are mutually incomparable, that is, nonordered.

To see how that would work for classical logic (CL), let us consider the following expression, namely,

$$\vdash_{\mathrm{CL}} (A \wedge B) \vee C \equiv (A \vee C) \wedge (B \vee C) \tag{1.1}$$

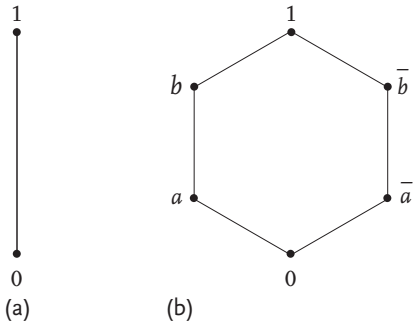


Figure 1.3 (a) Boolean lattice model of classical logic; (b) hexagon lattice model of classical logic. Also called o6 [244, 245, 277].

where $\vdash_{CL}$ denotes provability of an expression from the axioms of CL or simply that an expression is *true* in CL.

Let us consider possible interpretations of CL, that is, possible semantics or models. To map propositions and expressions formed by propositions, we use a *semantic valuation* (v): a function from the set of all formulas of CL to a set of all formulas of its model. If a model is a lattice, we will have $v(A) = a$, $v(A \wedge B) = a \cap b$, and so on.

Now, if the model is a Boolean algebra, the valuation of the valid CL formula given by expression (1.1) is a well-known property of Boolean algebra (BA) – the *distributivity*:

$$(\forall a, b, c)[(a \cap b) \cup c = (a \cup c) \cap (b \cup c)] \tag{1.2}$$

because if $\vdash A \equiv B$ is true in a BA-valuation, and then $v_{BA}(A) = a = v_{BA}(B) = b$ holds in this valuation.

However, if the model is an o6, then the following holds

$$v_{o6}(A \equiv B) = 1 \Rightarrow v_{o6}(A) = a \neq v_{o6}(B) = b\,, \tag{1.3}$$

and as a consequence, we have

$$(\exists a, b, c)[(a \cap b) \cup c \neq (a \cup c) \cap (b \cup c)]\,. \tag{1.4}$$

To prove this, let us take $a = v_{o6}(A) = b$, $c = v_{o6}(C) = a$, and $b = v_{o6}(B) = \overline{a}$, in Figure 1.3b. We obtain $a \cap b = 0$ and $(a \cap b) \cup c = 0 \cup c = c$. On the other hand, we have $a \cup c = a$, $b \cup c = 1$, and $(a \cup c) \cap (b \cup c) = a \cap 1 = a$. Since $c \neq a$, we do not have $(a \cap b) \cup c = (a \cup c) \cap (b \cup c)$.

Nevertheless, in this model, one can prove all the tautologies (theorems) and all the inference rules that are valid in the standard two-valued classical logic [244, 245, 277, pp. 272, 305].

Taken together, logic is a much wider and weaker theory than its lattice models – Boolean algebra, o6, and so on – through which logic can or cannot be implemented in a hardware. Two-valued Boolean algebra is definitely the simplest model for which such an implementation is possible and this determines the choice of the hardware. We can say that the computation is *physical*. Physical hardware determines how fast we can compute a problem, which algebra we shall use for the purpose, and how we can *translate* our problem into the chosen algebra and therefore hardware. This *physical* aspect of computation is of utmost importance for any attempt to speed-up computation, classical or quantum. In the following sections, we will discuss some of them.

1.5
Speeding up Computation: Classical Analog Computation ...

In the previous section, we showed that the logic we use for reasoning on propositions and operations carried on them can have nonbinary models. On the other

hand, although real numbers that we use for everyday calculations can be based on "binary" (two valued) logic. When we want to carry out a calculation, we first have to translate every real number to a binary one which has got more digits. Then, we have to carry out complex gate manipulations in order to apply algorithms that translate otherwise simple operations with real numbers into operations with binary digits. In the end, we have to translate the result back into real numbers. Would it not be faster to make a *real computer* that would be able to deal with real numbers directly?

In the theory of computation, real computers are hypothetical computing machines which can use infinite-precision real numbers. These hypothetical computing machines can be viewed as idealized analog and parallel computers which operate on real numbers. Realistic analog computers existed in the past, but they were abandoned for the following two reasons

1. digital (binary) computers proved to be faster;
2. analog computers have never been developed to fully universal machines.

Although, the latter reason is apparently only a consequence of the former one.

Both digital and analog computational devices are very old. For instance, Chinese *counting rods* and *abacus* digital "computers" (know in practically all ancient civilizations), are over 2000 years old. Analog *Antikythera mechanism* and *astrolabe* for calculating astronomical positions are nearly as old.

What is important for us, though, is that the analog/parallel computers in the "predigital" time were more efficient then digital for particular tasks simply because a special design of hardware enabled faster and more efficient calculation then by means of a universal digital machine. It is important, because, on the one hand, known quantum algorithms (mostly based on the Fourier transform) determine which feature quantum hardware must possess, and on the other, as opposed to current classical computers, massive parallel computation is what characterizes would-be quantum computers and is likely to make them universal. We can say that both analog classical and quantum computers perform a physical calculation.

To better understand what that means, let us have a look at Figure 1.4.

The examples show how we can calculate even irrational (π) using geometrical and physical features of our "hardware" as suitable algorithms for solving particular problems. More sophisticated examples of analog computational devices based on such algorithms are, for example, slide rule and the *Water integrator* shown in Figures 1.5 and 1.6, respectively.

Of course, the analog/parallel computers that were in use after World War II were electronic ones, but the principle stayed the same – physical calculation. With the help of the so-called *operational amplifier* (op-amp)[3] we can add, subtract, multiply, and divide number as well as obtain derivatives and integrals of a chosen function in one step by simply reading output voltages. For example, if we want to divide two numbers, we use a circuit shown in Figure 1.7.

3) The first vacuum tube op-amp was built in 1941.

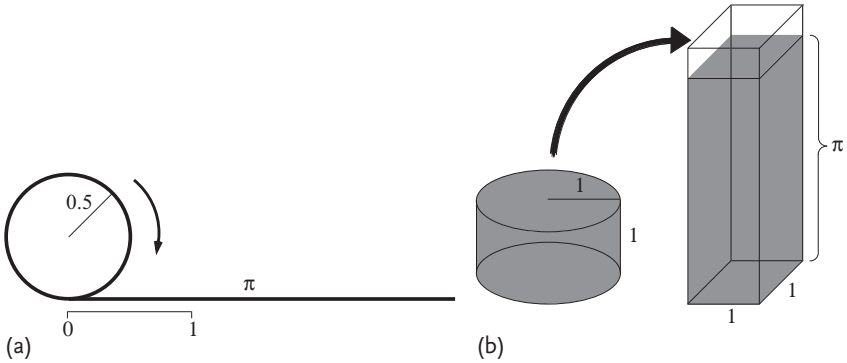


Figure 1.4 (a) "Calculating" π by measuring the length of a string originally wrapped around a cylinder with a radius equal to 0.5; (b) "calculating" π by pouring over water from a cylinder to a vessel whose base is a square 1×1 and measuring the height of the water level.

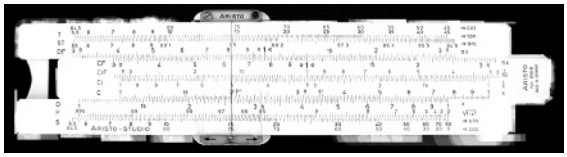


Figure 1.5 A slide rule, essentially being an analog computer, is much more efficient than its digital competitor abacus.

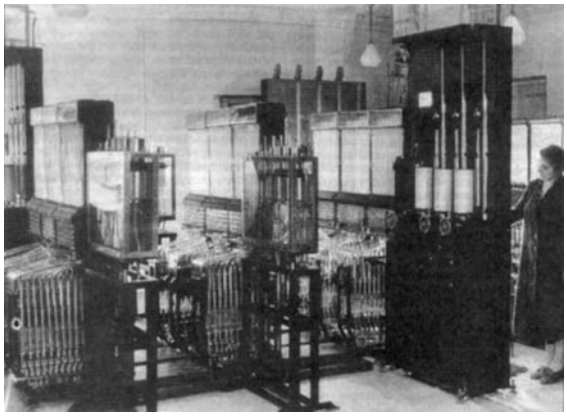


Figure 1.6 A Russian water analog computer built in 1936 by Vladimir Lukyanov. It was capable of solving nonhomogeneous differential equations. Image courtesy of the Polytechnic Museum, Moscow.

Op-amp has a resistor with a very high resistance between $-$ and $+$ terminal so that the current across them is practically zero. Thus, we have $I_- = I_+$ and therefore $V_g = V_+$. Since in our circuit we have $V_g = 0 = V_+$, we must also have $V_- = 0$. Also, $V_{in} - V_- = V_{in} = I_{in} R_{in}$ and $V_- - V_{out} = -V_{out} = I_f R_f$. Then, from $I_{in} = I_f + I_-$, we get $V_{out} = -R_f V_{in}/R_{in}$. By setting R_f to one, we can divide V_{in} by R_{in} in one step. We see that for each division, we have to change V_{in} and R_{in}. When we want to integrate a function, we have to use different elements and for integrations, yet other ones.

There were no such problems with universal digital computers that were advancing rapidly and the *Moore law* finally proclaimed the dead sentence to analog and parallel computers. Moore's law is an *Intel Corporation* self-imposed[4] longterm production road map. After an obviously too ambitious formulation by Gordon Moore, the "law" was recalibrated in 1975 so as to receive the following formulation as announced by David House, an Intel executive at the time [17, 271, 308, 329]:

> Moore's Law. *CPU clock speed and the number of transistors on an integrated circuit double every 18 months.*

However, it was obvious from the very begging that both the CPU speedup and its miniaturization as well as miniaturization of memory units would one day hit the quantum wall. Miniaturization has to stop when the bit carriers come down to one electron, when logic gates and memory units come down to one atom and when the conductors between them come down to monolayers. Actually, in the very same year, when the Moore's law received its definitive wording – in 1975 – Robert Dennard's group at IBM predicted that the power leakage which would switch a transistor out of its "off" state should happen by 2001 – shown in the left image of Figure 1.8. They also formulated their – *Dennard's scaling law* – which specified how to simultaneously reduce gate length, gate insulator thickness, and other feature dimensions to improve switching speed, power consumption, and transistor density and ultimately postpone the leakage. However, the fast developing industry

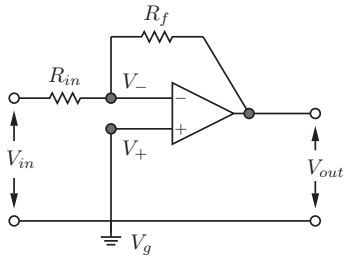


Figure 1.7 Analog computer. Dividing numbers by means of voltages with the help of an operational amplifier.

4) by Gordon Moore, a cofounder of Intel, in the early seventies of the last century

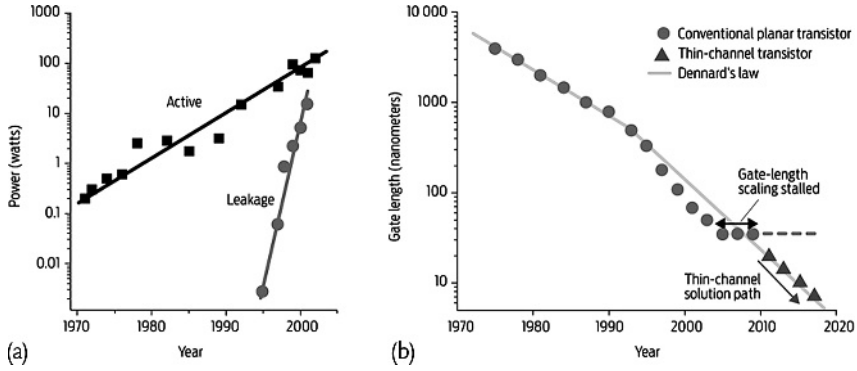


Figure 1.8 Moore's miniaturization will stop by 2020 at the latest. Figure reprinted from [4] with permission from © 2011 IEEE Spectrum magazin.

modified their law and used other technological solutions to pack that 5 billions transistors in a CPU.

However, the quantum wall is inevitable one way or another and now the sizes of gates themselves are approaching the nanometer barrier – an atom has the size of about half a nanometer and, as shown in Figure 1.8b, this will happen by 2020 if the new *thin-channel* solution of designing and connecting transistors proves to be successful [4]. If not, the miniaturization – as Figure 1.8 also shows – has already stopped.

The CPU clock exponential speed-up already hit the wall a few years ago. In 2003, the exponential speed-up turned in a linear one and in 2005 Intel gave up the speed-up completely – see Figure 1.11. In 2008, IBM took over at a pace even slower than linear and dedicated its CPUs to the mainframe usage with a price of over $ 100 000 per CPU. Therefore, after 2005, individuals cannot even dream of speeding up their computations for quite some time to come.

Thus, the researchers started to look for alternatives and turned to parallel computation. For today's market, that meant parallelizing digital computers (we shall come back to this in Section 1.7), but development research turned to quantum and analog computers (again).

For example, the special 2010 issue of *Computers* entitled *Analog Computation* introduces the renewed interest as follows. "Computer scientists worldwide are exploring analog computing under such names as amorphous computing, unconventional computing, computing with bulk matter, nonsilicon computing and other designations. Biologists and computer scientists team up to build "computers" out of neural tissue or slime molds. Physicists design new materials, such as graphenes, whose molecular properties are analogous to the atomic-level quantum behavior. Theoretical computer scientists investigate the complexity of analog computing, and speculate on new complexity classes. All this emerging work has resulted from the limits that physical laws impose on digital computers."

This may enable real computers to solve problems that are inextricable on digital computers. For instance, Hava Siegelmann's neural nets can have noncomputable real weights, making them able to compute nonrecursive languages. Also, the recent development of the massively parallel computer, the so-called *field computer*, indicates that we might be able to solve the so-called NP-problems in a polynomial time.

What that would mean was best explained by Kurt Gödel in a 1956 letter to John von Neumann: "If there actually were a machine with [a polynomial running time] this would have consequences of the greatest magnitude. That is to say, it would clearly indicate that, despite the unsolvability of the Entscheidungsproblem, the mental effort of the mathematician could be completely (apart from the postulation of axioms) replaced by machines."

However, there is a new kind of parallel computers on which we can – in a polynomial time – solve problems which require an exponential time on classical computers. These are *quantum computers* whose physical and parallel computation we are going to analyze in most of the following sections.

1.6 ... vs. Quantum Physical Computation

In this section, we shall show – on a small scale – how a quantum computer works – in principle. What is important here is

- a feature of a quantum system – *photon* – called *superposition* which is a nonclassical property and which enables massive parallelism;
- a physical calculation in a polynomial time of a problem whose solving requires an exponential time on a classical computer.

Let us consider a simple experiment consisting of a photon splitting its path at a 50 : 50 beam splitter (BS; a semitransparent mirror), as shown in Figure 1.9. We denote the two possible incoming paths and also the corresponding states of the photon moving along them by $|0\rangle$ and $|1\rangle$. These are the so-called *ket* vectors belonging to Dirac's *bra-ket* notation which we will formally introduce in Sections 1.8 and 1.11. So, either the photon arrives from above and has the state described by $|0\rangle$ or from below in state $|1\rangle$.

The photon can either go through or be reflected from the beam splitter. Let us take the case of photon $|0\rangle$ coming in. If it passes through, its field vector will remain unchanged. But, because it passes through BS with only 50% probability, we multiply its ket by $1/\sqrt{2}$. On the other hand, a vector field reflected from BS undergoes a phase shift $\pi/2$ with respect to the one which passes through it. (See [78] where you have to assume that the lower incoming beam does not contain a photon.) This phase shift corresponds to multiplying the ket by $e^{i\pi/2} = i$, and therefore the reflected photon will be described by $(1/\sqrt{2})i|1\rangle$. Hence, *before* we detect which outgoing path the photon took – by registering a "click" in either D_0 or D_1 – we

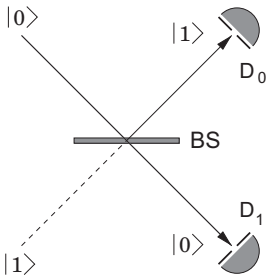


Figure 1.9 Photon at a beam splitter.

describe its state by the following *superposition* (see Section 1.11 for a formal definition) of paths:

$$|\text{out}\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle) \,. \tag{1.5}$$

Such a superposition of states is the crucial ingredient of quantum computation.

Let us now use our photon, our beam splitter, and another beam splitter to make a quantum computer prototype. Such a two beam splitter set through which a photon passes is a device known under the name of a *Mach–Zehnder interferometer* and is shown in Figure 1.10.

The path to the second beam splitter (BS) from above is described by $(i/\sqrt{2})|1\rangle$ and from below by $(1/\sqrt{2})|0\rangle$. Here, we can simply reverse the process we have on the first beam splitter as follows. The two paths superpose at the beam splitter so that upper outgoing path is described by

$$\frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2}}|0\rangle + i\frac{i}{\sqrt{2}}|0\rangle\right) = 0 \,, \tag{1.6}$$

where the second $|0\rangle$ comes from $i|1\rangle$ which was reflected from BS (at the upper side of BS we denote it as $|0\rangle$). The lower path is described by

$$\frac{1}{\sqrt{2}}\left(\frac{i}{\sqrt{2}}|1\rangle + i\frac{1}{\sqrt{2}}|1\rangle\right) = i|1\rangle \,, \tag{1.7}$$

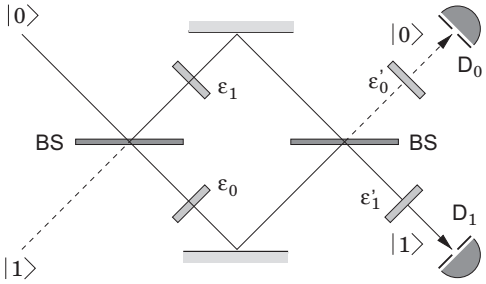


Figure 1.10 Mach–Zehnder interferometer. An incoming $|0\rangle$ ($|1\rangle$) photon will always end up in D_1 (D_0) detector.

where the second $|1\rangle$ comes from $|0\rangle$ which is reflected from BS (on the lower side of BS we denote it as $|1\rangle$) and the first one from a passage of $i|1\rangle$ from above. The phase shifters ϵ_0 and ϵ_1 are set to $\epsilon_0 = 0$ and $\epsilon_1 = 0$ (we tune them off the zero-values to obtain an arbitrary probability of photons exiting through any of the two port). For the zero-values setup, the process at the second beam splitter is just a reversed image of the process at the first one. The probability of detecting the photon by D_0 is 0, and the probability of detecting it by D_1 is $|\langle 1|(-i)i|1\rangle|^2 = 1$.

If we, however, set ϵ_0, ϵ_1, ϵ_0', and ϵ_1' so as to make phase shifts (with respect to the state of the incoming photon) ϕ_0, ϕ_1, ϕ_0', and ϕ_1', respectively, then the probability of detecting a photon by D_1 is no longer 1, but

$$p_1 = \cos^2 \frac{\phi_1 - \phi_0}{2} = \frac{1}{2}(1 + \cos \phi) \,, \tag{1.8}$$

where $\phi = \phi_1 - \phi_0$. Note that the probability would stay the same if we took out the phase shifters ϵ_0' and ϵ_1' which means that the result depends only on the phase difference $\phi_1 - \phi_0$.

Let us see how we can use the result to factor numbers in order to illustrate Shor's algorithm (short of entanglement and the corresponding speed-up, which we are going to address later on), following Johann Summhammer [301].

We obtain the factors of a chosen number, say N, in a "physical" way using the setup shown in Figure 1.10 of the previous section and (1.8). Let us increase the phase shift ϕ in discrete steps $2\pi/n$ so as to have $\phi_j = 2\pi k N/n$, $k = 1, \ldots, n$. If we let n photons through the device: $k = 1, \ldots, n$, the sum of all individual probabilities that the detector D_1 would register a photon – given by (1.8) – will be:

$$I_n = \sum_{k=1}^{n} p_1(k) = \frac{1}{2}\left[n + \sum_{k=1}^{n} \cos\left(\frac{2\pi k N}{n}\right)\right] . \tag{1.9}$$

If n were a factor of N, we would have $p_1(k) = 1$ and $I_n = n$. If not, the cosines would roughly cancel each other and we would get $I_n \approx n/2$. If n were a factor of N then only detector D_1 would react and if n were not a factor of N, then on average we would get half of the clicks in D_1 and half in D_0. So, if we perform n measurements and obtain n clicks in detector D_1 then n is a factor of N.

The numbers we can factor in this way are not big, but the result is very instructive for understanding the problems we face with classical computers and the way we can solve them with quantum ones. For the light with $\lambda = 500\,\text{nm}$, using a continuous wave (CW) laser (for example, Nd:YAG) with which we can have the *coherence length*, Δl – the length over which the phase is fairly constant – of up to 300 km. The corresponding *coherence time* is $\Delta t = \Delta l/c$. The Heisenberg uncertainty relation for energy and time $\Delta E \Delta t \geq \hbar$ and the Planck postulate: $E = h\nu$ give $\Delta\nu\Delta t \approx 1/4\pi$, where $\Delta\nu$ is called the *bandwidth*. From $c = \nu\lambda$ by differentiation we get $\Delta\lambda = -c\Delta/\nu^2 = -\lambda^2\Delta\nu/c$, where $\Delta\lambda$ is called the *linewidth*. Dropping the minus sign which only shows that the changes of $\Delta\nu$ and $\Delta\lambda$ are opposite and using the previous relations we get: $\Delta l \approx \lambda^2/\Delta\lambda$. To keep the linewidth at $\Delta\lambda \approx 10^{-17}$ is feasible since it corresponds to the coherence length $\Delta l \approx 25\,\text{km}$.

In our setup, at each phase step $\Delta\phi = 2\pi/n$ a photon is sent into the interferometer. The phase difference $\Delta\phi$ in our interferometer is proportional to $\Delta o/\lambda$, where Δo is the *optical path difference* [33]. The Δo must be smaller than the coherence length and we can estimate that $n < \lambda/\Delta\lambda$.

Hence, the biggest numbers we could factor are $N \approx 10^{10}$ and any PC can factor a number with 10 digits in a fraction of a second. However, the important property of this example of physical computing is that our "transistor," Mach–Zehnder interferometer, is faster per computing unit (quantum gate) than the standard classical transistor for the same "clock" speed.

The longest factorization test according to (1.9) will take time proportional to nN, because the maximum value of k is n. Since the largest n we have to check is $\sqrt{N}$, the maximum time would be proportional to $N^{3/2}$. The required time is therefore a polynomial function of N.

A direct and the most inefficient algorithm of factoring a number would simply be $\sqrt{N}$ trial divisions. Hence, the number of checks the most inefficient classical factoring algorithm has to carry out is smaller than $N^{3/2}$ we obtained for our "physical calculation" above. Still, given the same clock frequency, a classical computer calculation is slower per computing unit (gate). There are two reasons for this. First, we have to turn numbers into bits, and then we have to carry out binary operations that correspond to division (which is one of the most complicated basic computer operations). The number of used transistors, that is, loops needed for the operations increases exponentially with N and that means that the required time is an exponential function of N. In other words, we obtained an exponential speed-up of factorization of numbers on our optical "quantum analog device" with respect to a binary computer cracking.

As opposed to a computer search-verify procedure, the photon search-verify Mach–Zehnder factorization procedure is instantaneous for each photon. The problem is that we cannot calculate much with only one Mach–Zehnder interferometer. We could parallelize the calculation by putting another Mach–Zehnder interferometer at each output of the first one, then putting another Mach–Zehnder interferometer at each output of the previous one, and so on (see Figure 2.1 in Section 2.2). However, that would mean an exponentially growing number of elements, causing us to lose the advantage we gained. We will show how to get around this later on. But, before we dwell on the solution to this problem, we should first show why do we need to speed-up our calculation at all.

1.7
Complexity Limits: Exponential Time

We have mentioned that the Moore law already hit the clock wall (see Figure 1.11) and that it will soon hit the quantum shrinking barrier – single electron transistor (SET) and monolayer conductors.

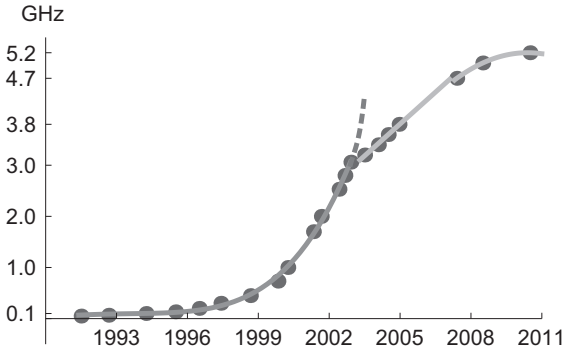


Figure 1.11 CPU's clock frequencies: Intel 486-50 MHz June 1991, DX2-66 August 1992, P(entium)-100 March 1994, P-133 June 1995, P-200 June 1996, PII-300 May 1997, PII-450 August 1998, PIII-733 October 1999, PIII-1.0 GHz March 2000, P4-1.7 April 2001, P4-2.0 August 2001, P4-2.53 May 2002, P4-2.8 August 2002, P4-3.0 April 2003, P4-3.4 April 2004, P4-3.6 June 2004, Intel P4-3.8 November 2004, IBM Risc Power-6 4.7 GHz June 2007, 5 GHz August 2008, IBM zEnterprise 196 (z196) 5.2 GHz August 2010.

Since 2004, when the clock frequency corollary of Moore's law died,[5] parallel processing has been introduced into the very processors: dual cores, quad cores, ... 16 cores. In a way, these processors are just mini clusters. Clusters and interconnected mainframe units have been used for decades to speed-up computation using algorithms that can distribute parts of a task to many CPUs in parallel. But, is that efficient? Actually, for the hardest computing problems we have to solve in various applications, neither classical parallelization nor a speed-up of CPUs are efficient in the sense of obtaining results proportionally faster with higher speed or a higher number of CPUs. Here is why.

Computing problems are categorized according to their complexity in the so-called *complexity classes*. The problems are defined by their models of computation and before they are considered as decision problems that algorithms have to resolve to reach a decision, that is, the final outcome. There are many undecidable problems as, for example, the so-called *halting problem*: "Given a description of a program and a finite input, decide whether the program finishes running or will run forever."

It is often said in the literature that already Alan Turing proved that no Turing machine can solve the halting problem, i.e, that it is undecidable for Turing ma-

5) A widespread rendering of the law: "The number of transistors on a single integrated-circuit chip doubles every 18 months" [28] does not correspond to the historical data which show 26 months [42]. Moore himself commented: "I never said 18 months. I said one year [in 1965], and then two years [in 1975]. One of my Intel colleagues changed it from the complexity of the chips to the performance of computers and decided that not only did you get a benefit from the doubling every two years but we were able to increase the clock frequency, too, so computer performance was actually doubling every 18 months. I guess that's a corollary of Moore's Law. Moore's Law has been the name given to everything that changes exponentially in the industry. I saw, if Al Gore invented the Internet, I invented the exponential." [271, 308, 329]

chines. But, that is only to be expected because a deterministic Turing machine stops after solving a decidable problem by definition. It can say nothing about a theory for which it cannot be defined. So, although there are many other undecidable problems, for us, only those problems that have an algorithm for their solving will be of interest.

We shall be even more specific and will concentrate on the *time complexity* of algorithms, although there is also a space complexity. We shall do so because one of the main advantages of quantum computers is that they are expected to require a polynomial time for solving problems for which classical computers would require an exponential time.

Thus, the class EXPTIME is the set of decision problems that can be solved by some algorithm in an exponential time, the class NP is the set of decision problems that can be solved by a nondeterministic Turing machine in polynomial time, while the class P is the set of decision problems that can be solved by a deterministic Turing machine in a polynomial time.

We stress here that all the problems we shall consider do have some algorithms for their solution. Thus, our main problem with quantum computation will not be to find algorithms for a computation of particular programs in general, but to find algorithms which will be exponentially faster (Shor's algorithm) or at least a few polynomial orders faster (Grover's algorithm) than classical algorithms. Such a speed-up is also possible in the realm of classical computation. Since a classical speed-up can compete with and even outdo quantum ones, some details might be helpful.

Let us consider P and EXPTIME problems for which there exist algorithms described by means of functions $f(n) = a_i n^i$, $i = 1, 2, 3$, $g(n) = b2^n$ and $h(n) = c3^n$, where a_i, b, and c are constants. We shall say that the algorithm is *of order* $O(n)$, $O(n^2)$, $O(n^3)$, $O(2^n)$, and $O(3^n)$.

From Table 1.2, we see that when we take a linear problem that we solved within one day on a personal computer (PC), increase its size by a factor of 1000, and put

Table 1.2 Problems of a polynomial complexity, that is, problems from a P class, can really take advantage of a speed-up of classical computers ($100N_3^3 = x^3 \Rightarrow x = \sqrt[3]{100}N_3 = 4.64N_3$). However, for a problem of an exponential time complexity the speed-up is limited to an additive constant. For example, $100 \cdot 2^{N_4} = 2^x \Rightarrow x = N_4 + (\log 100)/(\log 2) = N_4 + 6.64$.

Time complexity	Size of a solvable problem in a time unit		
	In 1991	Today (2013; on 100 times faster CPU)	Today on a cluster with 1000 CPUs (10^5 times faster)
n	N_1	$100N_1$	$100\,000N_1$
n^2	N_2	$10N_2$	$316.2N_2$
n^3	N_3	$4.64N_3$	$46.4N_3$
2^n	N_4	$N_4 + 6.64$	$N_4 + 16.6$
3^n	N_5	$N_5 + 4.19$	$N_5 + 10.5$

it on a cluster with 1000 CPUs, we will obtain a result also within one day. If one does that with a problem from an EXPTIME class, the required time would exceed the age of the Universe even on whatever cluster we have today.

This was the reason why the following definitions have been proposed.

Definition 2 A polynomial algorithm of a problem is called *feasible* [73].

We often simply say that such a problem is feasible.

Definition 3 A problem which does not have a feasible algorithm is called *intractable*.

Definitions 2 and 3 are not always appropriate because

- Constant factors and lower terms in a polynom can make an "intractable" problem feasible and a "feasible" one intractable. For example, an algorithm that would take time $10^{100}n$ cannot be carried out, but is nevertheless called "feasible" because it is in P, while an algorithm that takes time $10^{-1000}2^n$ can easily be carried out for n as large as 1000, but is called "intractable" because it is in EXPTIME;
- The size of the exponent and of the input can have the same effect.

Still, we do not encounter such unfavorable cases often and therefore the definitions are widely accepted. But, we have to keep in mind that "intractable" does not mean that a problem cannot be computed or that we do not have an algorithm for it. It simply means that we have to spend more time or that we do not have enough money to solve the problem.

Let us have a look at a few problems: *Euler tour*, *Traveling salesman*, and *factoring number* ones.[6)]

The first one is the Euler tour problem for a multigraph. An Euler tour is a tour which covers all the edges but none of them more than once. For example, the multigraph shown over Königsberg bridges in Figure 1.12 does not have an Euler tour. It is shown here because Euler formulated his tour problem and found a linear algorithm for it while solving the Königsberg bridges problem.

Definition 4 A *graph* $G = (V, E)$ consists of a set (V) of *vertices* (points) and a set (E) of *edges* (lines), each of which connects two vertices. A *multigraph* is a graph which has multiple edges.

6) To that, we can add the *satiability problem* (SAT) and the *isomorph-free generation of graphs and hypergraphs*. SAT problem consists in verifying whether Boolean expressions like that one shown in Table 1.1 are satisfied, that is, true, that is, equal to 1 in a Boolean algebra. SAT belongs to EXPTIME and that will help us understand why the factoring number problem computed in a digital computer belongs to EXPTIME too. Graphs and hypergraphs can be used to map nonlinear equations into hypergraphs and filter out equations that have solutions. They also belong to EXPTIME. We shall use them later on to generate the so-called Kochen–Specker sets.

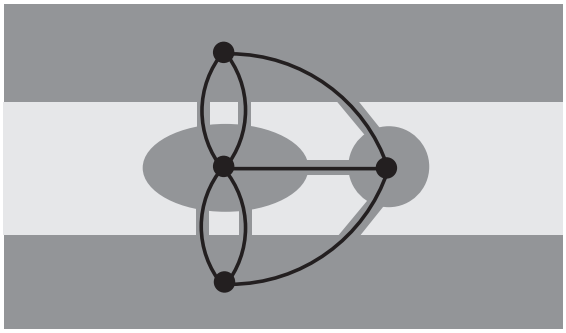


Figure 1.12 Euler tour on the example of Königsberg bridges. Is it possible to take a tour over the bridges, crossing each one only once?

A brute force approach to this problem gives us a search algorithm of complexity order $O(n!)$ (where n is the number of edges) which is even harder to solve than those of order $O(c^n)$. For instance, the number of paths we have to verify for the Königsberg bridges is $(7-1)!$. If a computer needs 1 sec to verify all of them, then the time required to verify paths for twice so many (14) bridges is $13!/6! \approx 100$ days.

A similar problem is the traveling salesman problem (TSP) which consists of finding the cheapest way of visiting all given cities and returning to your starting point. The vertices are cities and edges are routes between any two of them. Each link (edge) connecting two cities (vertices on the edge) is pondered by the cost of going from one of the two cities to the other. Since this is a realistic problem every travel agency would like to have a solution for, we will first try to estimate to what extent they can be of service to their customers if they use a brute-force algorithm which is again of the order $O(n!)$.

Let us assume the agency has a fast machine which provides the cheapest route for 10 cities in 1 s. Should they try to serve a demanding customer who would like to make the cheapest tour through 25 cities? Well, the required time is about 136 billion years or 10 ages of the Universe.

Therefore, a better algorithm for such problems are wanted. But no general approach has been found so far. For instance, it is not known whether the NP set strictly contains the P set or perhaps coincides with it. So, problems are approached individually or according to some features they share.

Euler proved that connected graphs have an Euler tour if every vertex shares an even number of edges and this immediately reduced the time complexity of the problem from EXPTIME to linear P.

S. Lin found a good approximate algorithm of order $O(n^3)$ for the traveling salesman problem. With the help of this algorithm, the agency would be able to serve its customer within 15.6 s, if only approximately.

The next problem of factoring numbers will show us how the complexity of an important application of algorithms we make use of every day depends on a plat-

form we use to solve problems and how we can find faster algorithms on new platforms.

As shown in Section 1.6, the complexity of algorithms for factoring numbers on our photon prototype device is of order $O(n^{3/2})$. The latter algorithms could use the electric analog machine for resetting the device. But, the number of voltage steps an analog computer can tell from each other is also limited. A more sophisticated analog computer would, when compared with a digital one, have two disadvantages: a much lower speed and an inefficient error correction. Therefore, for the time being, a digital computer is the only option for the task.

However, to introduce a natural number N we want to "crack" into a digital computer we have to translate it into a binary string:

$$N_2 = \alpha_{n-1}\alpha_{n-2}\ldots\alpha_1\alpha_0 \tag{1.10}$$

where α_i, $i = 0, \ldots, n-1$ are determined from the following equation:

$$N_2 = \alpha_{n-1}2^{n-1} + \alpha_{n-2}2^{n-2} + \cdots + \alpha_1 2^1 + \alpha_0 2^0 = \sum_{i=0}^{n-1} \alpha_i 2^i \,. \tag{1.11}$$

For instance, to obtain a binary representation of 255, we divide it by 2 until we reach 1. Reminders determine bits. So, 255/2 is 127 with the remainder $\alpha_0 = 1$, and so on, down to $\alpha_7 = 1$ and we get 11111111. In the opposite direction, we have $2^0 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6 + 2^7 = 255$. For 256, we have all the remainders, but the last one equal to zero: $\alpha_0 = \alpha_1 = \ldots = 0$. The last one (of 1/2) is, of course, 1. Thus, we get 100000000 and $2^8 = 256$.

A brute force algorithm for the task consists of trial divisions using basic Boolean operations by means of logic gates shown in Figure 1.2 combined in binary adders as mentioned in Section 1.4 and explained in [239, Section 1.16]. That means that such a search would be of order $O(2^n)$ or higher, where n is the number of bits.

The majority of encryption we use today for bank and Internet transactions are based on composite numbers consisting of two huge prime numbers. They are called RSA after Ron Rivest, Adi Shamir, and Leonard Adleman who invented this encryption method in 1978 [269]. And again, many faster *subexponential*[7] algorithms have been found.

RSA company provides harder and harder challenges every year to stimulate finding better algorithms. On December 12, 2009, such a number with 768 bits and 232 digits was cracked[8] after 2000 CPU years of computation, meaning that it required one month on a cluster with 24 000 CPUs. Also in 2009, a group of enthusiasts factored 512-bit RSA keys for Texas Instruments calculators using software found on the Internet and a distributed computing project. Since 512-bit RSA numbers are the standard for almost all Internet keys, the aforementioned crackings stirred a debate on the RSA keys security.

7) Subexponential or superpolynomial complexity is the one between P and EXPTIME.

8) Number Field Sieve algorithm of subexponential complexity $O\{\exp[c(\log n)^{1/3}(\log\log n)^{2/3}]\}$ was used.

The response of the companies will most probably be to simply switch to a 1024-bit standard, but two issues emerge here. First, already now, all previously illegally intercepted documents encoded by older 256- and 128-bit keys are easily readable. Soon, all intercepted and stored 512-bit ones will be readable. Second, tonight a mathematician somewhere in his attic room can come with an ingenious P algorithm for factoring numbers and crash down the security of the World Internet as of tomorrow morning.

Here, quantum computation and quantum communication can provide a patch.

What the Internet needs is to make connections secure and eavesdropping impossible and that is what quantum cryptography can provide the Internet with already today.

What computation needs is a speed-up, and that is what the hardware of would-be quantum computers together with quantum software of the "Shore kind" can provide us with.

However, before we dwell on these two issues, we first want to consider another important point that will give us a bridge from classical to quantum platforms and from classical to quantum formalism. We have already mentioned that the classical technology already "went parallel" and that means a lot of CPUs, that is, a lot of heat. So, the final issue we have to elaborate on before we go completely quantum is the issue of energy.

1.8
Energy Limits ...

According to the Environmental Protection Agency (EPA) US Congress report in 2007, the energy used by servers and data centers in the US is estimated to be about 61 billion kWh in 2006 (1.5% of total US electricity consumption) for a total electricity cost of about $ 4.5 billion [265]. This estimate includes neither office nor private PCs and it is evaluated to be higher than the electricity consumed by all color televisions in the US.[9)]

EPA also estimated the energy use of servers and data centers in 2006 to be more than double the electricity that was consumed for this purpose in 2000, and that the power and cooling infrastructure that supports IT equipment in data centers also uses significant energy, accounting for 50% of the total consumption of data centers [265]. Taken together, servers and data centers together with their infrastructure in 2006 spent 2.25% of total US electricity consumption. Similar statistics are available for Europe. Intensity of computations constantly going on in Europe

9) The total energy consumption of energy related to computers (in industry, offices, and at home) and Internet (including cooling, personal, maintenance, rooms, and so on) is independently estimated to be between 3 and 9% (in the US), but no detailed study has been carried out in at least 10 years. This is partly because computers are so much a part of production, education, communication, traveling, and everyday life that it is practically impossible to determine the energy spent by them from the total amount of spent energy.

is obvious from the European Particle Physics Real Time Monitor shown in Figure 1.13.

Should we add the new *parallel law*

- *The energy spent by clusters and data centers doubles every five years*

to the dying Moore law.

Apparently, still not. Because the energy spent in the subsequent five year period (2005–2010) did not double [153]. It only increased by about 60% worldwide. There are two main reasons for that. The first is the global crisis occurring in the past few years. The second is the recent virtualization of computational tasks. Recently developed cloud computing installations have higher server utilization levels and infrastructure efficiencies than in-house data centers. But, since the latter filling of the presently existing computational "vacancies" in the existing in-house servers will soon saturate them and since most reports do predict an exponential growth in energy spent by the data centers, the parallel law will continue to hold in its exponential formulation.

On the other hand, the designers of computers and the Internet argue that their efficiency has increased several times over in the last twenty years. Previous NMOS and PMOS transistors dissipated heat through their resistors while today's CMOS gates dispense with resistors; optical fibers substitute copper wires; resistance within conductors is being lowered by reducing the number of electrons within a gate from thousands to one hundred and soon it will be reduced to one in single electron transistors (SET). Can this development outweigh the exponential increase of processed and stored petabytes (PB, 10^{15} byte)? Here, we should mention that in the face of all the mentioned improvements in the efficiency there is a part of the Moore law that has outperformed itself recently and that is about the heat dissipated by the processors since the dissipated heat doubles not every 18 months, but each year or less.

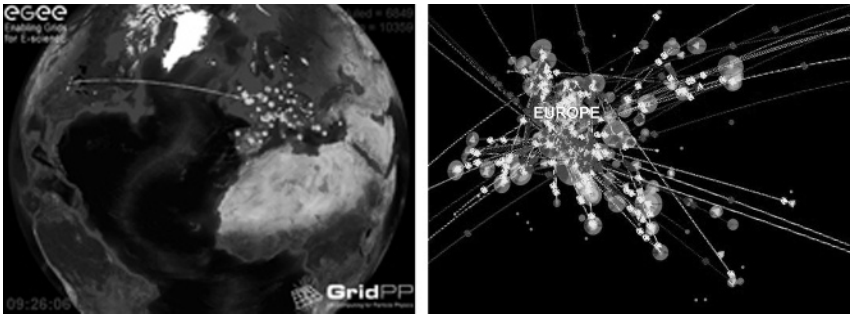


Figure 1.13 Distributed or grid computing consists of sharing computing tasks over multiple computer clusters. Elementary particle physics tasks constantly running on European EGEE and GridPP (distributed over all national European Grids and with links to American (both), Asian, and Australian computing clusters) are shown (57 226 tasks running (dots; bigger dots mean bigger clusters), 21 884 queueing). Reprinted with permission of the UK Grid Operations Support Centre; © GridPP.

The problem is that when we come down to one electron per gate, we are left with pure "information heat" and for so many bytes, it becomes considerable. The information energy that is dissipated in gates just because they compute data or erase data is then significant. For thousands of electrons per transistor erasing data can be compared with erasing data from a book. When we burn two books, one with blank pages and the other with Galileo's *Dialogue on the Two Chief World Systems* printed in it, we would not be able to detect a difference in dissipated heat. But, when we go down to just several electrons we move around or several atoms whose magnetization in a hard disk[10] we changed, then computation and communication become physical. The energy needed for creating or erasing one bit of information directly corresponds to the energy needed to move or change its carrier. Let us calculate this energy.

We shall do so by means of an ideal gas model. We put gas consisting of atoms in a cylinder with a piston as shown in Figure 1.14. Pressure which atoms exert on the piston is $p = F_x/a$ where F is the force with which atoms bounce onto the piston and the walls of the cylinder. So, the work done by the gas is

$$W = \int F_x dx = \int p\, dV\,; \tag{1.12}$$

because $dV = a\,dx$.

We assume that the gas is in a bath at a constant temperature T. The law of ideal gas reads $pV = NkT$, where N is the number of atoms and $k = 1.381 \times 10^{-23}$ J/(molecule K) is the Boltzmann constant. Since the temperature T is constant, the average kinetic energy of the atoms does not change and therefore there is no change of the internal energy. Hence, according to the first law of thermodynamics, work W is equal to the heat Q transfered to a heat reservoir.

Our process is reversible (there is no friction and by returning the heat by means of a reversible process attached to ours, we can restore its initial state) and therefore, using the second law of thermodynamics (the definition of the entropy change for a reversible process is $\Delta S = Q/T$), from (1.12), we obtain

$$\Delta S = \frac{Q}{T} = \frac{1}{T}\int_{V_i}^{V_f} \frac{NkT}{V} dV = Nk \ln \frac{V_f}{V_i}\,. \tag{1.13}$$

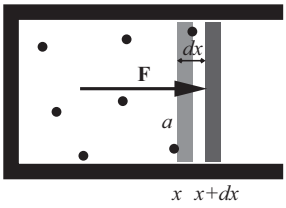


Figure 1.14 Work done by ideal gas during isothermal expansion.

10) Thin layer of magnetic material hard disks are coated with layers 10 nm thick.

Now, let us put just one atom in an empty cylinder – as shown in Figure 1.15. A Maxwell demon watches it and when it is in the left half of the cylinder, he records it ("1") and introduces a piston (adiabatically and reversibly) in the middle of the cylinder (a). In this way, he records one bit of information in cylinder's memory "for free." When he wants to erase this information from its memory, he allows the piston to move freely (without friction) and isothermally. The atom will do work against the piston and push it to the right (b). Our demon now takes out the piston (reversibly and adiabatically) and removes "1"; one bit is erased (c). The cost is given by (1.15).

The entropy increase of the environment caused by erasure of one bit of information is

$$\Delta S = \frac{Q}{T} = k \ln \frac{2V_i}{V_i} = k \ln 2 \,. \tag{1.14}$$

This is known as the *Landauer principle* [165]. The dissipated heat caused by the erasure of one bit is

$$Q = kT \ln 2 \,. \tag{1.15}$$

Instead of dividing the cylinder in two compartments, the demon could have divided it in 4 or 8 or any number w of possible states. Then, we arrive at the famous Boltzmann microscopic entropy

$$\Delta S = k \ln w \tag{1.16}$$

which is as epitaph engraved in Boltzmann's gravestone.

From (1.14), it follows that we cannot discard information in a computer without dissipating heat, no matter how clever we design our circuits. This is a physical law which we cannot go around because we have to assume some work on the part of the computer (atom in Figure 1.15) at least when the calculation is over and the output has to be obtained. This corresponds to "removing of 1" in Figure 1.15b; also, if the demon simply adiabatically removed the piston in (b), then the system would not be in any way connected to its environment and would not provide us with any output. But, we can carry out calculation without discarding information on each step of calculation and that can save us from unnecessary heat dissipation. Let us see how we can do that.

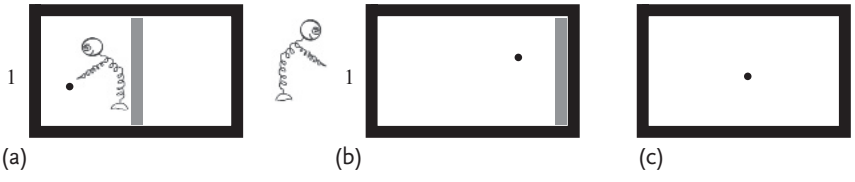


Figure 1.15 Entropy of single atom gas.

1.9
. . . and Reversible Gates

When we, in a decade or two, scale down the transistors to one electron (single electron transistors, SET) and the conductors to monolayers one atom thick, that is, when all Moore's laws die, we will have to take care of "information garbage," that is, the informational heat it produces, given by (1.16).

Since transistors in such atom-level processors will be extremely densely packed – already today their number exceeds 5 billions[11] – we have to think of a way to get rid of the huge amount of heat per volume unit the discarded bits would develop.

And, the best way to get rid of the heat the gates (transistors) would produce is to make gates that do not produce heat. That was the idea (in the early eighties) of reversible computers that would be able to calculate running both "forwards" and "backwards" – like a pendulum – without either dissipating or taking in new energy while calculating.

However, can the binary Boolean algebra and its gates support such swinging reversible circuits?

Let us have a look at Figure 1.16 (compare with Figure 1.2). By looking at the output of a NOT gate, we immediately know what the input was. So, it is reversible. If we keep track of any of the two inputs of an XOR gate, we can reconstruct the other input by looking at it outputs. However, to be able to reconstruct the inputs of an AND gate, we have to keep track of both of them because by knowing that both the output and one of the inputs were 0, we still cannot know whether the other input was zero or one. So, the answer is in the negative. Standard logic gates cannot be implemented in a reversible circuit.

But, if we collect input and output data of a gate, that would suffice to make any operation reversible. Such three-level gates are called the gates of *logic width 3*. (The standard binary logic gates have therefore logic width 1.) Bits at the first two incoming ports of reversible ports are often called the *control* bits or *source* bits, the input bits *target* or *argument* bits, the output ones *result* bits and the one that are not used in further calculations *sink* bits or *garbage*. The terminology will often depend on the kind of gate we will use. With the Fredkin gate [98] (Table 1.3), we obtain different operations at different ports of the gate. With the Toffoli gate, we obtain

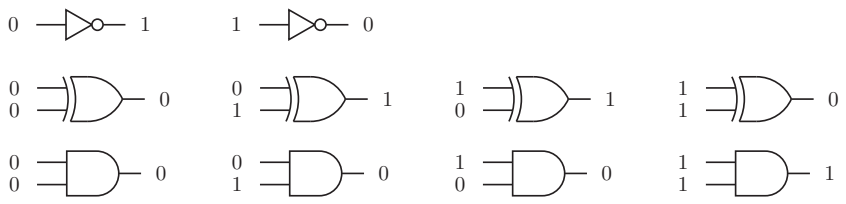


Figure 1.16 Gate NOT is reversible. For the XOR gate to be reversible, at least one of the inputs has to be kept in memory. For AND, both inputs should be kept in memory.

11) Intel 62-core Xeon Phi.

Table 1.3 Truth table of the Fredkin gate – a universal reverse gate that can be used to implement any other gate. Two examples are given. Encircled are the values of the *control* (0) and *result* bits for *controlled* AND gate. Boxed are *control* (1) and *result* values for *controlled* OR gate. Arrows show another way of implementing the latter gate.

Fredkin gate					
Input–output ports			Output–input ports		
Port 1	Port 2	Port 3	Port 1	Port 2	Port 3
0	0	(0)	0	(0)	0
0	0	[1]	0	1	[0]
0	→ 1	(0)	0	→(0)	1
0	→ 1	[1]	0	→ 1	[1]
1	0	(0)	1	(0)	0
1	0	[1]	1	0	[1]
1	→ 1	(0)	1	→(1)	0
1	→ 1	[1]	1	→ 1	[1]

a result mostly at the last output port. We can also use different ports as control ones. For instance, the Fredkin gate originally used input port 2 for control bits and output 2 to obtain operation OR (indicated by arrows in Table 1.3).

The truth values of the Fredkin gate show that we can run it backwards as well. That prompted Fredkin and Tofolli (in 1982 [98]) to propose another way of representing the Fredkin gate which could be integrated in a circuit and enable experimental and industrial implementation. It is shown in Figure 1.17.

In 1985, Richard Feynman [95] recognized that the ability of reversible gates to run backwards as well as forward is just the main feature of the unitary evolution of any quantum system. Thus, he proposed a concept of *quantum mechanical computers* which would essentially use the gates and circuits proposed for reversible computers only applied to *quantum bits*: photons, electrons, and atoms.

Feynman recognized that the *Toffoli gate* and circuits proposed by Tommaso Toffoli in 1980 [303] are better suited for a would-be quantum application and that the Toffoli gate is but one gate in a series of scalable gates which he called NOT, CNOT (CONTROLLED NOT), CCNOT (CONTROLLED CONTROLLED NOT), ... The Toffoli gate is a CCNOT gate. Feynman–Toffoli circuit notation enables an easy handling of gates and is widely accepted in both fields – reversible and quantum computer research.

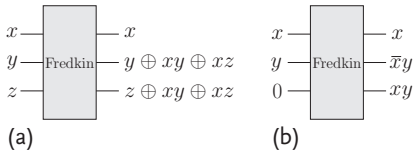


Figure 1.17 (a) General schematic of the Fredkin gate; (b) Schematic of the implementation of an AND gate by means of the Fredkin gate; We can easily check that it is a special case of (a) (see the caption of Figure 1.2). In that way, we can write down any reversible Boolean gate by means of the universal Fredkin gate.

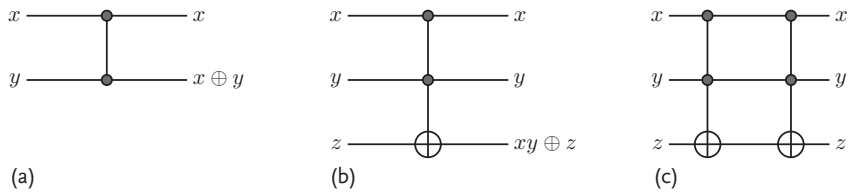


Figure 1.18 (a) General schematic of a CNOT gate; $x \oplus y = x\overline{y} + \overline{x}y$; For $x = 1$, we obtain $1 \oplus y = \overline{y}$; (b) General schematic of a CCNOT gate ($xy \oplus z = \overline{xy}z + xy\overline{z}$); For $x = y = 1$, we obtain $1 \oplus z = \overline{z}$; (c) Reversibility of the CCNOT shown by two concatenated CCNOT gates. This is equivalent to first running CCNOT forward as in (b) and then backward.

Circuit formalism for CONTROLLED-... NOT gates is shown in Figure 1.18. If we concatenate two CCNOT gates, then the 3rd port takes the output of the first gate as its input and the 3rd port of the second gate gives us

$$\begin{aligned} xy \oplus (xy \oplus z) &= \overline{xy}(\overline{xy}z + xy\overline{z}) + xy\overline{(\overline{xy}z + xy\overline{z})} \\ &= \overline{xy}z + xy(x\overline{y} + \overline{z})(\overline{xy} + z) \\ &= (\overline{xy} + xy)z = z \ . \end{aligned} \tag{1.17}$$

This result is graphically presented in Figure 1.18c.

We can see that we obtain the input z we started with and therefore the gate is reversible, actually self-reversible. We can also see that in CC... NOT gates, the control bits and target-result bits are separated and that the control bit inputs are identical with control output ones and are therefore conveniently designed for scaling circuits containing them.

Since the values of the control bits stay the same and the target bit involves symmetric Boolean operation NOT, it is easy to describe the action of the gate on the input state by a matrix. The matrix representation of the three-level CCNOT gate – shown in (1.18) – consists of an "operator-matrix" that just takes care of swapping values of the target bit and "state matrices" that are just columns of the CCNOT truth table as shown in Table 1.4. Actually, this matrix representation seems to have been adopted from the quantum formalism in the classical reversible computation literature. We nevertheless write it here to point to some differences between properties of classical reversible gates and circles and their formalism, on the one side, and quantum ones, on the other.

$$\begin{bmatrix} 1 & & & & & & & \\ & 1 & & & & 0 & & \\ & & 1 & & & & & \\ & & & 1 & & & & \\ & & & & 1 & & & \\ & & & & & 1 & & \\ & & 0 & & & & 0 & 1 \\ & & & & & & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix} \tag{1.18}$$

Table 1.4 Truth tables of CNOT (controlled NOT) and CCNOT (controlled controlled NOT) gates. The latter gate is also called the *Toffoli gate*. C stands for control and T for target bits.

CNOT gate			
In–out		Out–in	
C	T	C	T
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

CCNOT (Toffoli) gate					
Input–output ports			Output–input ports		
C1	C2	T	C1	C2	T
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

We see that apart from the target bit values, all the off-diagonal elements in the matrix are equal to zero. The matrix is equal to itself transposed and multiplied by itself transposed it gives a unit matrix. Since it is a real matrix, it is therefore a unitary matrix as a matrix of a quantum operator. Therefore, its action can clearly be reversed. We can obtain this result by multiplying (1.18) by the matrix from the left. The matrix multiplied by itself is equal to $\mathbb{1}$ and we obtain (1.18) with the reversed positions of "state matrices."

However, an almost diagonal form of the matrix means that the gate exerts only a limited action on the "state matrices." If we wanted to implement other operations, we would have to tamper it with the latter matrices and use both control and target bits as shown in Figure 1.19. As we can see in Figure 1.19c, we use not only the target level, but also the control levels to introduce parameters for obtaining the results. This makes building up circuits more demanding than in a standard binary computer so far as the number of gates is concerned. For example, a comparison of a reversible parallel adder with a standard binary shows that about 40% more gates is needed. This is quite acceptable, though, because both implementations have the complexity $O(n)$. The power consumption, on the other hand, is reduced to 10% of those in the standard chips [76].

On the other hand, we have some restrictions on the circuits that we do not have for the binary circuits. For example, real hardware fan-outs (copies of gate outputs) are not allowed because such copying is irreversible – number of input signals is one and there should be two or more output signals and this is not possi-

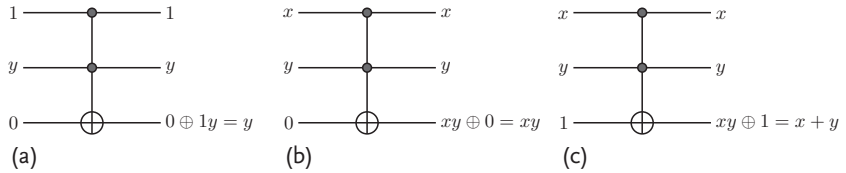


Figure 1.19 (a) Fan-out (two copies of y) simulation (it has to be used for copying gate outputs because a hardware fan-out is not allowed in a reversible circuit); (b) AND implementation; (c) OR implementation which requires four additional CCNOT gates at the control ports.

ble since we cannot generate energy from nowhere (remember that in a reversible circuit electrons/energy "swing"). We can simulate fan-out though, as shown in Figure 1.19a. Classical reversible circuits share the impossibility of having fan-outs with the quantum circuits. The reason why we cannot have a fan-out in a quantum circuit (not even simulated) is the so-called *no-cloning principle*, that is, that a quantum bit cannot be copied. (We shall come back to this principle later on in the book.) Similarly, in reversible circuit, feedbacks (loops) are also not allowed because that would disturb the regularity of "swinging."

In order to implement an OR gate, we have to use either four additional CCNOTs as indicated in Figure 1.19c or a combination of NAND (input target is 1) and NOT (both controls are 1). This is not a problem because CCNOT is universal in a reversible circuit. But, since it is universal neither in the standard binary sense nor in the sense of a quantum universal gate, we shall define the reversible universal gate here [77].

Definition 5 A reversible gate is *r*-universal if and only if any Boolean function $f(x_1, x_2, \ldots, x_n)$ can be synthesized by a loop-free and fan-out-free combinatorial network built from a finite number of such gates, using each input $x_1, x_2, \ldots, x_n$ at most once and using an arbitrary finite number of times the constant inputs 0 and 1.

Both Fredkin and Toffoli (CCNOT) gates are *r*-universal and are necessary and sufficient for a reversible implementation of arbitrary Boolean function of a finite number of logical variables. Now, in the standard classical circuits fan-outs are allowed and the smallest universal gates (NAND and NOR)[12] are of width 1 and are linear; In quantum circuits fan-outs are not allowed and the smallest universal gates are of width 2 and are linear. What is the smallest logic width of *r*-universal gates. Can we correlate a hardware "no fan-outs" restriction with a software condition? The answer is given by the following theorem.

Theorem 6

A reversible gate is *r*-universal if and only if it is not linear [77].

12) Not only can we express all other operations by means of NAN and NOR, but we can also compress all the conditions of the Boolean algebra in a single axiom [193] and [327, pp. 807,1174]. We can even express all the conditions with the help of a universal operation so that they keep an identical form when we substitute NAND, OR, and so on, for that universal operation [198].

A truth table of a logic gate of width w consists of 2^w lines. A gate is reversible if and only if all 2^w output values form a permutation of all 2^w input values. That makes $(2^w)!$ different reversible gates. Two reversible gates of width 1 and all 24 $[(2^2)! = 24]$ of width 2 are linear. Therefore, the smallest r-universal gate are of width 3. There are 1344 linear of all $(2^3)! = 40\,320$ reversal gates of width 3 that makes 38 976 r-universal gates of width 3. Quantum gates do not have truth tables, and so we will look for another explanation of their properties in the next sections.

This also gives us an answer as to why it is relevant to go into the details of gate algebras. They tell us a great deal about hardware; for binary, reversible, and quantum gates alike.

To sum up, the idea of reversible computers emerged from an energy consideration of the scaling down electronic elements to atomic level in the future. At the same time, the idea of quantum computers emerged from a consideration of how to speed-up computation once the CPU clock hits its quantum limits. The development of both ideas are being developed, although the quantum computers are better funded for an obvious reason – no matter how well we solve the heat dissipation, the classical computer can never have a transistor that would work on less than one electron and can never have conductors thinner than one atom while quantum superposition mimics just that.

There are properties that reversible and quantum gates share. These are the reversibility itself, gate and circuit formalism, unitarity of matrices, universality of gates at a particular level, absence of fan-outs, and functionality at the atom level.[13] There are properties that they do not share. For example, there are neither truth values nor truth tables for quantum gates, only for classical reversible ones. Then, at least for the time being, the reversible circuits are much slower than the standard binary ones, while the quantum ones are exponentially faster than the standard binary ones, at least for particular algorithms.

1.10
Ultimate Efficiency: Quantum Computers and Qubits

In Section 1.7, we have seen that the classical solution to a collapse of the expected exponential increase of the CPU speed is a massive parallelism in both individual PCs and supercomputers. Intel has put forward a new slogan: "Parallelism Full Steam Ahead!" in its new journal *The Parallel Universe* [266].

On the other hand, photons – being quantum systems – inherently possess massive parallelism based on their ability to superpose their states and, as we have

13) The main feature of reversible circuits is their power efficiency which stems from the Landauer principle given by (1.14). However, the Landauer heat given by (1.15) is significant only for single electrons (any technology that relies on many electrons supporting a single bit dissipates the heat that is altogether much higher than the sum of information heats of individual electrons). The need for reversible gates will increase as we continue with the miniaturization of transistors and eventually arrive at single electron ones. Quantum circuits, on the other hand, are bound to an atomic level from the very start since they compute by means of photons, electrons, and atoms.

seen in Section 1.6, we can use this superposition of quantum states for a "physical" quantum computation. The prototype we have described there cannot cope with massive calculations due to the requirement that the optical path difference be smaller than the coherence lengths of the laser. Yet, it *was* a quantum computer with one quantum bit only – serving as a quantum CPU with a single quantum transistor, that is, a single quantum gate – that was capable of factoring numbers with up to 10^{10} digits.

This means that 50 quantum bits, each having two states $|1\rangle$ and $|0\rangle$, would give us a superposition of $2^{50} \approx 10^{15}$ states. Any gate operation on these 50 quantum bits amounts to an interaction with all 2^{50} states in parallel since they are all in collective phonon modes. These quantum bits build a composite Hilbert space $\mathcal{H} = \mathcal{H}^2 \otimes \cdots \otimes \mathcal{H}^2$. The computational basis, that is, the basis of this space, consists of the following 2^{50} vectors: $|00\cdots00\rangle, |00\cdots01\rangle, \ldots, |11\cdots11\rangle$. To compute a function f of each of these states means to let the states evolve according to the time evolution unitary operator U (Schrödinger equation):

$$|i_1 i_2 \ldots i_{50}\rangle \longmapsto U|i_1 i_2 \ldots i_{50}\rangle = |f(i_1, \ldots . i_{50})\rangle \; . \tag{1.19}$$

In a classical computer, we would carry out such a computation in a one-state-at-a-time sequence. In a quantum computer, we first put all the states on the left-hand side of (1.19) in a superposition of all 2^{50} basis states and then let them evolve together and in one step:

$$\sum_{i_1 i_2 \ldots i_{50}=0}^{1} \alpha_{i_1 i_2 \ldots i_{50}} |i_1 i_2 \ldots i_{50}\rangle \overset{f}{\longmapsto} \sum_{i_1, i_2, \ldots, i_{50}=0}^{1} \alpha_{i_1 i_2 \ldots i_{50}} |f(i_1 i_2 \ldots i_{50})\rangle \; . \tag{1.20}$$

After that, we let the obtained (evolved) superposition collapse to a particular state that we read as a result. Of course, since such a collapse of the wave packet is intrinsically statistical, we have to repeat it a number of times, but this procedure is of a polynomial complexity provided that we find a proper function f for a problem we want to calculate.

The difference between this quantum and a binary classical computer consists of the fact that 2^{50} states are formed by only 50 quantum transistors (quantum bits), while in a classical computer, we need a new transistor for each new state, that is, 2^{50} or about one million billion transistors or about half a million of today's most advanced CPUs. More realistic and detailed estimations give about 10^6 quantum bits for such computational power, though [264].

Of course, to be able to use this parallelism we must – as for classical parallel systems – find appropriate quantum hardware and software solutions. Quantum computing power would depend on how well we could correct errors and faults in computation, on how well we could interconnect qubits, and on how efficient the algorithms that we would find for them are. To arrive at each aspect of quantum bits, we have to first learn of their most basic properties and how we can handle them. We shall do that starting from their abstract definition in the Hilbert space, but in a pedestrian approach.

Definition 7 A *qubit* (*qu*antum *bit*) is a two-state quantum system. The two states form a basis in a two-dimensional Hilbert space $\mathcal{H}^2$ and are denoted $|0\rangle$ and $|1\rangle$. They are vectors in $\mathcal{H}^2$, form a basis in $\mathcal{H}^2$, and span $\mathcal{H}^2$. In the matrix representation, they read:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{1.21}$$

In the spin and polarization representation, we have $|\uparrow\rangle = |0\rangle$, $|H\rangle = |0\rangle$ (*spin up, horiziontal polarization*) and $|\downarrow\rangle = |1\rangle$, $|V\rangle = |1\rangle$ (*spin down, vertical polarization*).

Definition 8 Any vector from $\mathcal{H}^n$ denoted as $|\Psi\rangle$ is called a *ket*.

Definition 9 Hilbert space $\mathcal{H} = \mathcal{H}^2 \otimes \ldots \otimes \mathcal{H}^2$ is called a *composite qubit space*. The basis of this space, consisting of vectors $|00\cdots00\rangle, |00\cdots01\rangle, \ldots, |11\cdots11\rangle$ is called the *computational basis*.

Qubits might be linear or circular polarization states of photons, two levels of an atom or ion, spin-1/2 nuclear states in a magnetic field (nuclear magnetic resonance, NMR), electron and nuclear states in a silicon, electron states in an electron dot, then charge, flux, phase, and charge-flux states in superconducting devices, and so on.

We shall often measure a state of a qubit so as to let it pass a filter which lets a qubit in a particular state through. In that case the qubit will not be distorted by passing through the filter. Such a filter is called a *perfect filter* and an *ideal measurement*. In quantum computation, filtering (perfect and imperfect) is often used for preparing and handling states of a qubit and we can consider the qubit that exited a specified port of a filter to be measured, although it has not been destroyed by – we say, *collapsed* in – a measurement device.

When we do not deal with a perfect filtering, then the state is distorted and changed by filtering. This can be statistically described and the outcome can be statistically predicted, but an outcome of an individual measurement will in the latter case remain random and unpredictable.

This inherent randomness in triggering a measuring device or passing through a filter whenever their setups do not match the state in which the qubit was prepared is one of the main features of a qubit. For example, if a qubit is prepared in a spin-up state oriented along the $+Oz$-axis as shown in Figure 1.20, then it will always pass through an s_z filter, but it will pass through $+Ox$ and $+Oy$ only every second time. We can verify similar behaviors with polarized photons. A photon prepared by a polarizer will pass through another polarizer oriented in the same direction.

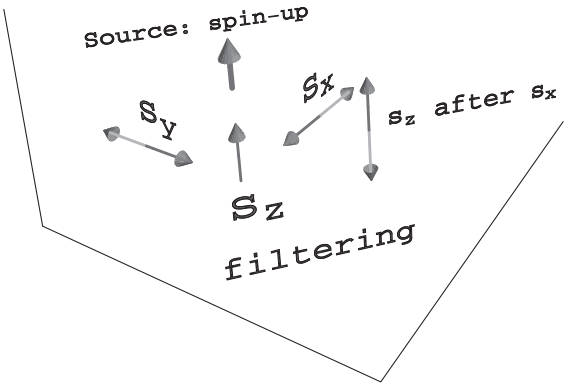


Figure 1.20 A qubit prepared in a spin-up state along $+Oz$ will always pass a $+s_z$ filter, but will pass (at random) $+s_x$ and $+s_y$ only in 50% of verifications (it will pass $-s_x$ and $-s_y$ in the other 50% of verifications). After passing s_x filters, the qubit will pass a new $+s_z$ filter only in half of the cases.

However, a polarizer rotated by 45° will only let every second photon through.

The probability of a photon passing the second polarizer oriented in the same direction as the first one is 1. The probability of a photon passing through the second polarizer rotated at an angle ϕ with respect to the first one is

$$P_\phi = \cos^2 \phi \ . \tag{1.22}$$

This is the so-called *Malus law*.[14] It gives meaning to the "statistical description" we mentioned above. For a particular photon, there is no way of predicting whether it will pass the polarizer or not, but the probability of passing it is $\cos^2 \phi$ for every one of them.

The randomness will make the algorithms for quantum computation rather demanding, but on the other hand, it will make messages coded by a quantum cryptography protocol unbreakable and eavesdropping impossible.

1.11
Combining and Measuring Qubits: Quantum Superposition – Qubit Primer

Superposition of qubits enables exponential speed-up of would-be quantum computation, as we explained in the previous section. It also determines the way in which we describe interaction of qubits, their manipulation, and their measurement. In Section 1.6, we introduced a superposition of photons in a Mach–Zehnder interferometer. In a quantum computer, we, however, expect to deal with qubits that are parts of atoms or ions and therefore we shall introduce a superposition by considering atom levels of a rubidium atom ^{87}Rb in a cavity as shown in Figure 1.21.

14) Of, say, 100 photons, $100 \cos^2 \varphi$ will pass the filter and $100 \sin^2 \varphi$ will not.

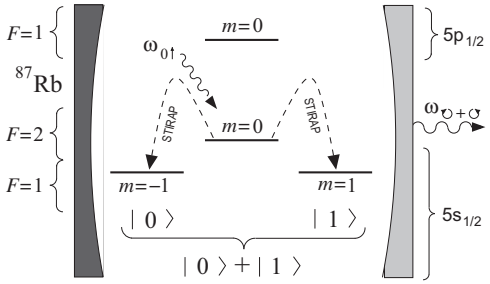


Figure 1.21 Superposition of atomic (^{87}Rb) states. *F* denotes hyperfine levels and *m* are magnetic Zeeman sublevels. (See Section 3.1.3.)

For the time being, we shall only consider essential features of such qubits and their superposition in order to better understand which physical systems can be chosen for qubits, which processes they can undergo, and how we can superpose them. We do so here because we want to stress the complexity of setting our qubits in superposed states. (See Section 3.1.3.)

The rubidium atom ^{87}Rb has four closed inner shells and one electron in the 5s shell. This means that we do not have to take electrons in the inner shells into account. The atom behaves like having just one electron. Both $5s_{1/2}$ and $5p_{1/2}$ levels are split by nuclear-electron interaction (*hyperfine* structure) into $F = 1$ and $F = 2$ (nuclear angular momentum of ^{87}Rb is 3/2 and therefore $F = 3/2 \pm 1/2 = 2, 1$). An external magnetic field $\boldsymbol{B}$ additionally splits the levels into magnetic Zeeman sublevels: $m = -F, -F + 1, \ldots, F$. We start with the electron in the state $F = 2, m = 0$ and apply a linear polarized laser beam of frequencies ω_0 to the atom. The atom absorbs the photon of energy $\hbar\omega_0$ and if it were not in a cavity, it would be excited to the level $5p_{1/2}$, $F = 1, m = 0$ (electron would be raised to the level)[15] (see Figure 1.22).

After that, it would spontaneously emit right and left polarized photons and would deexcite to levels $m = 1$ and $m = -1$, respectively. Deexcitation to $F = 1$, $m = 0$ is forbidden by the selection rules.[16] However, the cavity whose resonance

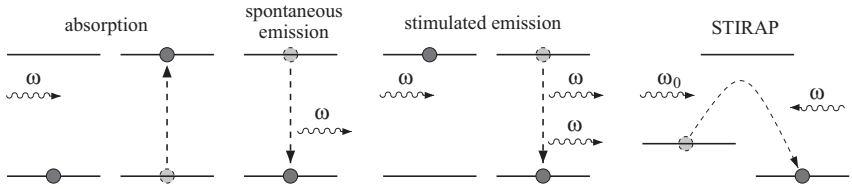


Figure 1.22 Absorption, spontaneous and stimulated emissions, and STIRAP.

15) We speak of an atom absorbing a photon and of an atom being excited because an isolated electron cannot absorb a photon. To see that, let us consider their energies in their center of mass frame. We should have $\hbar + m_e c^2 = m_{e0} c^2$, but that is impossible since the mass of an electron in motion cannot be smaller than its rest mass.

16) There are some more details here, like the detuning of the laser beam, and so on, that we will not go into.

frequency is set to ω stimulates "emission" before the absorption is complete, that is, before the electron reached level $5p_{1/2}$ and the atom adiabatically goes to two other ground levels. This adiabatic process is called STIRAP, which we will come back to in Section 3.1.3. So, the $5p_{1/2}$ level is not populated, but we nevertheless obtain a photon of frequency ω in the state $|\circlearrowright\rangle + |\circlearrowleft\rangle$ that would eventually leak from the cavity. So, here, as in Figure 1.10, we have indistinguishable states claimed by the same qubit that amounts to its being in a superposition of both of them.

A cavity with properly chosen parameters and its resonance frequency can either enhance or suppress emission [119]. In our case, it would quasi-enhance emission of ω photon (although the upper level remains unpopulated) and that results in STIRAPs to both ground levels. Thus, the electron shares both levels or the atom is in both states. Since our atom is our qubit, we say that the qubit is in a *superposition* of its states $|0\rangle$ ($m = -1$) and $|1\rangle$ ($m = 1$).

Definition 10 A *superposition* of basis states $|0\rangle$ and $|1\rangle$ is their linear combination

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle\,; \tag{1.23}$$

which is also a vector from the space spanned by $|0\rangle$ and $|1\rangle$. So, $|\psi\rangle$ is a vector sum of vectors $|0\rangle$ and $|1\rangle$, where α, β are arbitrary complex numbers. Its matrix representation is

$$\alpha \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \beta \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}. \tag{1.24}$$

Since qubits might interact and receive a representation in a bigger Hilbert space, we introduce the following definition.

Definition 11 A linear combination of basis states $|\psi_i\rangle$, $i = 1, \ldots, n$ is a vector

$$|\psi\rangle = \sum_{i=1}^{n} c_i |\psi_i\rangle\,; \tag{1.25}$$

of ket space where c_i are arbitrary complex numbers. Vector $|\psi\rangle$ is a *superposition* of vectors $|\psi_i\rangle$.

Now, any linear function ϕ of ket vectors $|\psi\rangle$, that is, $\phi(|\psi\rangle)$, possesses the superposition property characteristic of ket vectors (see (1.25)). For instance, from (1.23), we get

$$\phi(\alpha|0\rangle + \beta|1\rangle) = \alpha\phi(|0\rangle) + \beta\phi(|1\rangle)\,. \tag{1.26}$$

Hence, if ϕ_1 and ϕ_2 have this property, then $\gamma\phi_1 + \delta\phi_2$ has it too.

Definition 12 A *bra* vector $\langle\phi|$ is a vector associated with a linear function $\phi(|\psi\rangle)$ of an arbitrary ket vector $|\psi\rangle$ from the ket space. Bra vectors belong to a vector space for which we say is *dual* to the ket space.

Definition 13 The value of $\phi(|\psi\rangle)$ for a ket $|\psi\rangle$ is a complex number which we call a *bracket* and denote by the symbol $\langle\phi|\psi\rangle$. We also call it a *scalar product* or an *inner product*.

Definition 14 A ket $|\psi\rangle$ and a bra $\langle\phi|$ (or simply, vectors ψ and ϕ) are *orthogonal* if $\langle\phi|\psi\rangle = 0$.

When we multiply (1.25) by $\langle\psi_k|$, we obtain $|\psi_i\rangle$, $i = 1, \ldots, n$ is a vector

$$c_k = \langle\psi_k|\psi\rangle\,; \tag{1.27}$$

since the following holds

$$\langle\psi_j|\psi_k\rangle = \delta_{jk} \stackrel{\text{def}}{=} \begin{cases} 1 & \text{if } j = k \\ 0 & \text{if } j \neq k\,. \end{cases} \tag{1.28}$$

Therefore, (1.25) can be written as

$$|\psi\rangle = \sum_{i=1}^{n} \langle\psi_i|\psi\rangle|\psi_i\rangle = \sum_{i=1}^{n} |\psi_i\rangle\langle\psi_i|\psi\rangle = \begin{pmatrix} \langle\psi_1|\psi\rangle \\ \langle\psi_2|\psi\rangle \\ \vdots \\ \langle\psi_n|\psi\rangle \end{pmatrix}. \tag{1.29}$$

For bra vectors, we have

$$\langle\phi| = \sum_{i=1}^{n} \langle\phi|\psi_i\rangle\langle\psi_i| = (\langle\phi|\psi_1\rangle\langle\phi|\psi_2\rangle \ldots \langle\phi|\psi_n\rangle)\ , \tag{1.30}$$

where the last expression is a 1-row matrix.

An inner (scalar) product can be written as (see (1.34))

$$\langle\phi|\psi\rangle = \sum_{i=1}^{n} \langle\phi|\psi_i\rangle\langle\psi_i|\psi\rangle\,. \tag{1.31}$$

Hypothesis 15

We assume that the properties of wave functions and their scalar products that hold for solutions of Schrödinger equations also hold for bra and ket vectors and their scalar (inner) products.

1. There exists a one-to-one *antilinear* correspondence between ket and bra vectors. We say that the bra $\langle\phi| = \alpha^*\langle 0| + \beta^*\langle 1|$ is a *conjugate* of the ket $|\phi\rangle = \alpha|0\rangle + \beta|1\rangle$.
2. $\langle\psi|\phi\rangle = \langle\phi|\psi\rangle^*$

3. For any $|\psi\rangle$, $\langle\psi|\psi\rangle \geq 0$ holds. $||\psi|| \equiv \sqrt{\langle\psi|\psi\rangle}$ is called a *norm* of vector ψ. $||\psi|| = 0$ iff (if and only if) $|\psi\rangle = 0$.
4. The space of ket vectors admits an *orthonormal basis* – a set of mutually orthogonal vectors of magnitude 1. We say that the space is *separable*. We call the space the *ket space*.
5. If a series $\sum_{k=0}^{\infty} \psi_k$ converges absolutely ($\sum_{k=0}^{\infty} ||\psi_k|| \leq \infty$), then the series converges in the ket space in the sense that the partial sums converge to an element of the space. A finite dimensional space is always complete.

The reader who is not familiar with the Schrödinger wave mechanics can simply take the conditions of Hypothesis 15 as postulates of the ket space.

Theorem 16

The ket space is a *Hilbert space*.

Proof: Properties 1–5 of Hypothesis 15 amount to a definition of a Hilbert space [203]. □

Definition 17 If to each $|\psi\rangle$ from a Hilbert space corresponded a certain $|\phi\rangle$ and if the correspondence were linear, then we would say that a *linear operator* A acts on $|\psi\rangle$ so as to yield $|\phi\rangle$:

$$|\phi\rangle = A|\psi\rangle \,. \tag{1.32}$$

Lemma 18

(a) $A = 0$ iff $\langle\psi|A|\psi\rangle = 0$; (b) $A = B$ iff $\langle\psi|A|\psi\rangle = \langle\psi|B|\psi\rangle$ for any $|\psi\rangle$.

Proof: For a given bra $\langle\phi|$, the scalar product $\langle\phi|(A|\psi\rangle)$ is a linear function of $|\psi\rangle$ since A is linear. Let us call this function $\langle\chi|$. A $\langle\chi|$ corresponds to $\langle\phi|$. The correspondence is linear since the scalar product is linear. So, we can say that $\langle\chi| = \langle\phi|A$. Thus, we obtain

$$\langle\phi|A|\psi\rangle = \langle\phi|(A|\psi\rangle) = (\langle\phi|A)|\psi\rangle \,. \tag{1.33}$$

(a) If $A = 0$, then by (1.33) $\langle\psi|A = 0$ for any $|\psi\rangle$. $\langle\psi|A$ is a function of $|\psi\rangle$; therefore $\langle\psi|A = 0$ yields $\langle\phi|A|\psi\rangle = 0$. If $\langle\psi|A|\psi\rangle = 0$, then function $\langle\psi|A$ vanishes by definition. However, since $\langle\psi|A| = 0$ for any $\langle\psi|$, we obtain $A = 0$; (b) Follows similarly. □

Definition 19 Using Definition 17 and (1.33), we define the following operations:

$$\begin{aligned}&\text{multiplication:} \quad (cA)|\psi\rangle = c(A|\psi\rangle); \quad \langle\phi|(cA) = c(\langle\phi|A)\\ &\text{sum:} \quad (A+B)|\psi\rangle = A|\psi\rangle + B|\psi\rangle; \quad \langle\phi|(A+B) = \langle\phi|A + \langle\phi|B\\ &\text{product:} \quad (AB)|\psi\rangle = A(B|\psi\rangle); \quad \langle\phi|(AB) = (\langle\phi|A)B.\end{aligned}$$

Definition 20 We define *identity operator* I as

$$I = \sum_{i=1}^{n} |\psi_i\rangle\langle\psi_i| \,. \tag{1.34}$$

We call $|\psi\rangle\langle\psi|$ a *dyad*.

Definition 21 We define the *commutator* ($[A, B]$) of two linear operators A and B as follows (a product of two linear operator is *associative*, but it is not *commutative*)

$$[A, B] = AB - BA \,. \tag{1.35}$$

Matrix representation of operators in $\mathcal{H}^2$ follows straightforwardly from (1.32) and (1.21), that is,

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \alpha A_{11} + \beta A_{12} \\ \alpha A_{21} a + \beta A_{22} b \end{pmatrix} . \tag{1.36}$$

For $\mathcal{H}^n$ we obtain a matrix representation analogously. It is closely connected with the measurements as we shall see below.

When we deal with many qubits, we represent them in a product of $\mathcal{H}^2$ spaces. Vectors and operators of such a system consisting of subsystems are then represented by a *tensor product* $A \otimes B$ between matrices that represent them. In a finite dimensional spaces and for finite dimensional matrices, the tensor product (also called a *direct product*) coincides with the so-called Kronecker product whenever it is base independent. In this book, we only deal with finite dimensional spaces, but in particular implementations (e.g., of the CNOT operators) the products will be base dependent, so we shall keep to the term "tensor product." However, with paying attention to how the coordinates change, we can use all practical algorithms and software recently developed for the Kronecker product, for example, in Wolfram's *Mathematica* and this is what we actually do whenever calculating any chosen tensor product.

Definition 22 Let

$$A = \begin{pmatrix} A_{11} & \dots & A_{1n} \\ \dots & \dots & \dots \\ A_{m1} & \dots & A_{mn} \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} B_{11} & \dots & B_{1q} \\ \dots & \dots & \dots \\ B_{p1} & \dots & B_{pq} \end{pmatrix} \tag{1.37}$$

be matrices that represent either operators or vectors. *Tensor product* $A \otimes B$ is then

$$A \otimes B = \begin{pmatrix} A_{11}B & \dots & A_{1n}B \\ \dots & \dots & \dots \\ A_{m1}B & \dots & A_{mn}B \end{pmatrix}$$
$$= \begin{pmatrix} A_{11}B_{11} & A_{11}B_{12} & \dots & A_{1n}B_{1(q-1)} & A_{1n}B_{1q} \\ A_{11}B_{21} & A_{11}B_{22} & \dots & A_{1n}B_{2(q-1)} & A_{1n}B_{2q} \\ \dots & \dots & \dots & \dots & \dots \\ A_{m1}B_{(p-1)1} & A_{m1}B_{(p-1)2} & \dots & A_{mn}B_{(p-1)(q-1)} & A_{mn}B_{(p-1)q} \\ A_{m1}B_{p1} & A_{m1}B_{p2} & \dots & A_{mn}B_{p(q-1)} & A_{mn}B_{pq} \end{pmatrix}, \tag{1.38}$$

where A_{11} in the first column appears in the first p rows and A_{11} in the first row appears in the first q columns, and so on.

Tensor products are in general not commutative.

In dealing with qubits, we shall only come across tensor products of vectors (states) and of operators represented by square matrices. However, these products are important for us and since they are likely to cause some intuitive misreading, we give a few typical examples here.

When we combine two qubits, their states are represented in a 4-dim space $\mathcal{H}^4 = \mathcal{H}^2 \otimes \mathcal{H}^2$ as

$$|0\rangle \otimes |1\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1\begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ 0\begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}. \tag{1.39}$$

In a similar fashion, we obtain

$$|00\rangle = |0\rangle \otimes |0\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}; \quad |10\rangle = |1\rangle \otimes |0\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix};$$
$$|11\rangle = |1\rangle \otimes |1\rangle = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \tag{1.40}$$

Note that the bra of, for example, $|10\rangle$, is $\langle 10| = (0010)$, that is, a transposed matrix.

As for operators, let us consider the so-called *Pauli matrices* and the *unit matrix*. They describe spin-projection filtering and measuring of spin-1/2 particles.

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}; \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}; \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}; \quad I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{1.41}$$

For instance, we recognize σ_x as a NOT gate (see Figure 1.1 and Section 1.4). It flips $|0\rangle$ to $|1\rangle$ and vice versa. Here is how it filters a superposition:

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \beta \\ \alpha \end{pmatrix} = (|0\rangle\langle 1| + |1\rangle\langle 0|)(\alpha|0\rangle + \beta|1\rangle) = (\beta|0\rangle + \alpha|1\rangle)\,. \tag{1.42}$$

The other two Pauli matrices and I in this representation – we call it the *ket-bra representation* – read

$$\sigma_y = -i|0\rangle\langle 1| + i|1\rangle\langle 0|\,; \quad \sigma_z = |0\rangle\langle 0| - |1\rangle\langle 1|\,; \quad I = |0\rangle\langle 0| + |1\rangle\langle 1|\,. \tag{1.43}$$

Some of their products are

$$\sigma_z \otimes I = \begin{pmatrix} 1\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & 0\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\ 0\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & -1\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}\,; \tag{1.44}$$

$$I \otimes \sigma_x = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}; \quad \sigma_y \otimes \sigma_z = \begin{pmatrix} 0 & 0 & -i & 0 \\ 0 & 0 & 0 & i \\ i & 0 & 0 & 0 \\ 0 & -i & 0 & 0 \end{pmatrix}\,. \tag{1.45}$$

$\sigma_z \otimes I$ means that we apply σ_z filter (gate) to the first qubit and just pass (do nothing to) the second one. With $I \otimes \sigma_z$, we pass the first qubit and σ_z filters the second. $\sigma_y \otimes \sigma_z$ means applying the σ_y filter to the first qubit and σ_z to the second.

When we look at the matrices (1.41) and (1.45), we see that they are all symmetric with respect to its diagonals, provided we applied the complex conjugation operation to their elements.

Example 23

To better understand what "do nothing" above means, let us have a look at the following example in some detail. We want to have σ_x (in the four-dimensional Hilbert space) which would act on the first qubit – let us denote it as $\sigma_x^{(1)}$ – and σ_x which would act on the second qubit – let us denote it as $\sigma_x^{(2)}$. Let us make them act on the qubits $|00\rangle = |0\rangle_1|0\rangle_2$

$$\sigma_x^{(1)}|00\rangle = (\sigma_{x1} \otimes I_2)|0\rangle_1|0\rangle_2 = \sigma_x|0\rangle_1 \otimes I|0\rangle_2 = |1\rangle_1 \otimes |0\rangle_2 = |1\rangle_1|0\rangle_2 = |10\rangle$$

$$= \begin{pmatrix} 0\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & 1\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\ 1\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & 0\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \end{pmatrix} \begin{pmatrix} 1\begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ 0\begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = |10\rangle \tag{1.46}$$

$$\sigma_x^{(2)}|00\rangle = (I_1 \otimes \sigma_{x2})|0\rangle_1|0\rangle_2 = I|0\rangle_1 \otimes \sigma_x|0\rangle_2 = |0\rangle_1 \otimes |1\rangle_2 = |0\rangle_1|1\rangle_2 = |01\rangle$$
$$= \begin{pmatrix} 1\begin{pmatrix}0&1\\1&0\end{pmatrix} & 0\begin{pmatrix}0&1\\1&0\end{pmatrix} \\ 0\begin{pmatrix}0&1\\1&0\end{pmatrix} & 1\begin{pmatrix}0&1\\1&0\end{pmatrix} \end{pmatrix} \begin{pmatrix}1\\0\\0\\0\end{pmatrix} = \begin{pmatrix}0&1&0&0\\1&0&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix} \begin{pmatrix}1\\0\\0\\0\end{pmatrix} = \begin{pmatrix}0\\1\\0\\0\end{pmatrix} = |01\rangle. \tag{1.47}$$

Definition 24 A square matrix $A = (A_{ij})$ which is equal to its own conjugate transpose (also called adjoint) matrix – $A^\dagger = (A^*_{ji})$ – is called a *Hermitian matrix* (or a *self-adjoint matrix*). That is, a Hermitian matrix is a matrix for which the following holds.

$$A = A^\dagger . \tag{1.48}$$

Definition 25 Let $|\phi\rangle$ be a conjugate to $\langle\psi|A$, where A is a linear operator. $|\phi\rangle$ depends antilinearly on $\langle\psi|$ and is therefore its linear function which defines a linear operator we call a *Hermitian conjugate* (or *adjoint*) operator $A^\dagger$ of A:

$$|\phi\rangle = A^\dagger|\psi\rangle . \tag{1.49}$$

We can easily prove the following properties of Hermitian conjugation:

$$\begin{aligned} \langle\phi|A^\dagger|\psi\rangle &= \langle\psi|A|\phi\rangle^* \\ (A^\dagger)^\dagger &= A \\ (cA)^\dagger &= c^* A^\dagger \\ (A+B)^\dagger &= A^\dagger + B^\dagger \\ (AB)^\dagger &= B^\dagger A^\dagger . \end{aligned} \tag{1.50}$$

The bra of the ket $|\psi\rangle$ is its Hermitian conjugate: $(|\psi\rangle)^\dagger$

Definition 26 A linear operator H is *Hermitian* if it is its own Hermitian conjugate operator, that is, its own adjoint

$$H = H^\dagger . \tag{1.51}$$

An operator G is *anti-Hermitian* if:

$$G = -G^\dagger . \tag{1.52}$$

Definition 27 An operator U is *unitary* if it is inverse of its own adjoint (conjugate transpose)

$$\boxed{UU^\dagger = U^\dagger U = I} . \tag{1.53}$$

Unitary operators play a crucial role in quantum computation, but on the other hand, all operators of interest in quantum mechanics are unitary. Let us see what the roles of unitary and Hermitian operators in quantum manipulations and measurements are.

Definition 28 A square matrix U is *unitary* if the following condition is satisfied

$$\boxed{UU^\dagger = U^\dagger U = I}\,, \tag{1.54}$$

where $U^\dagger$ is the conjugate transpose of U.

Definition 29 The complex number α is called an *eigenvalue* and the ket (vector) $|\psi\rangle$ an eigenket (*eigenvector*) if

$$A|\psi\rangle = \alpha|\psi\rangle\,. \tag{1.55}$$

The bra $\langle\phi|$ is an eigenbra with an eigenvalue β if

$$\langle\phi|A = \beta\langle\phi|\,. \tag{1.56}$$

It is obvious that all eigenvalues of Hermitian operators are real and that makes them measurable.

Definition 30 If an arbitrary vector from space $\mathcal{H}$ can be expressed by means of eigenvectors of a Hermitian operator, we say that the eigenvectors span $\mathcal{H}$ and form a complete set and therefore a Hermitian operator is called an *observable*.

Observables are the only operators that we can directly measure in the sense of obtaining "clicks" from a measuring device which statistics then correspond to eigenvalues of the measured observable (for instance, momentum and position). However, as we shall see later on, we can measure unitary operators indirectly in the sense of detecting its action on a ket (state) which are not its eigenvectors. An important example of observables that are directly measurable are projectors.

Definition 31 A set of vectors orthogonal to subspace $\mathcal{E} \subseteq \mathcal{H}$ (i.e., orthogonal to all vectors from $\mathcal{E}$) form a subspace $\mathcal{E}^\perp \subseteq \mathcal{H}$. We say that $\mathcal{E}^\perp$ is *orthogonal* to $\mathcal{E}$ and call it the *complementary subspace* of $\mathcal{E}$. Every vector can be written as

$$|\psi\rangle = |\psi_E\rangle + |\psi_E^\perp\rangle \tag{1.57}$$

where $|\psi_E\rangle \in \mathcal{E}$ and $|\psi_E^\perp\rangle \in \mathcal{E}^\perp$.

Theorem 32

Operator P_E

$$P_E|\psi\rangle = |\psi_E\rangle \tag{1.58}$$

is a *projector* and an observable. It satisfies the relation

$$P_E^2 = P_E \,. \tag{1.59}$$

Proof: Each $|\psi\rangle$ (see (1.57)) has a projection in $\mathcal{E}$ and a projection in $\mathcal{E}^\perp$. That means that there is a unique $|\psi_E\rangle$ for each $|\psi\rangle$. This correspondence is linear and can be described by means of a linear projection operator (*projector*) P_E:

$$P_E|\psi\rangle = |\psi_E\rangle \,. \tag{1.60}$$

We also have

$$\langle\psi|P_E|\phi\rangle = \langle\psi|\phi_E\rangle = \langle\psi_E|\phi_E\rangle = \langle\psi_E|\phi\rangle \Rightarrow \langle\psi|P_E = \langle\psi_E| \,. \tag{1.61}$$

Hence, P_E is a Hermitian operator. P_E has two eigenkets $|\psi_E\rangle$ and $|\psi_E^\perp\rangle$ and two eigenvalues 1 and 0, respectively. Therefore, P_E is an observable.

$P_E^2 = P_E$ follows from

$$P_E^2|\psi\rangle = P_E(P_E|\psi\rangle) = P_E|\psi_E\rangle = |\psi_E\rangle = P_E|\psi\rangle \,. \tag{1.62}$$

□

Definition 33 A *pure state* is any nonzero ket $\alpha|\psi\rangle$ (where α is complex number) from a subspace $\mathcal{H}_P$ of a Hilbert space.

The set $\{\alpha|\psi\rangle\}$ is also called a *ray*. The projector onto $\mathcal{H}_P$ is

$$P_E = \frac{|\psi\rangle\langle\psi|}{\langle\psi|\psi\rangle} \,. \tag{1.63}$$

Theorem 34

Let $\mathcal{H}_A$ be a vector space formed by linear superposition of eigenkets $|\psi_i\rangle_i$, $i = 1, \ldots, n$ of a Hermitian operator A from Hilbert space $\mathcal{H}$. The eigenvalues of A are a_i, $i = 1, \ldots, n$. If A is an observable and its spectrum (set of eigenvalues) is discrete, then $\mathcal{H}_A = \mathcal{H}$ and we have

$$\sum_{i=1}^{n} P_i = \sum_{i=1}^{n} |\psi_i\rangle\langle\psi_i| = I \tag{1.64}$$

where P_i are projectors to the eigenkets of A. Equation (1.64) is called the *decomposition of unity* with respect to eigenvalues a_i. Observable A can be written as

$$A = \sum_{i=1}^{n} a_i P_i = \sum_{i=1}^{n} |\psi_i\rangle a_i \langle\psi_i| \,. \tag{1.65}$$

If P_i are degenerate, we have

$$A = \sum_{j=1}^{m}\sum_{i=1}^{n} a_i P_{ij} = \sum_{j=1}^{m}\sum_{i=1}^{n} |\psi_{ij}\rangle a_i \langle\psi_{ij}| \,. \tag{1.66}$$

Postulate 35

The *mean value* of a function $f(A)$ of an observable A is

$$\boxed{\langle f(A)\rangle = \frac{\langle\psi|f(A)|\psi\rangle}{\langle\psi|\psi\rangle}\,.} \tag{1.67}$$

We can see that vectors $\alpha|\psi\rangle$ and $|\psi\rangle$ represent the same dynamical state, whatever the operator $f(A)$ is. For instance, $e^{i\tau}|\psi\rangle$ and $|\psi\rangle$ represent the same dynamical state, that is, a dynamical state is defined by a vector up to a phase vector.

Postulate 36

Measurable values of a quantity are eigenvalues of an observable A associated with the quantity.

A measurable value must be real. This is assured by the fact that every observable is a Hermitian operator and every eigenvalue of a Hermitian operator is real.

Postulate 37

Let us consider eigenvalues of observable A in a domain D. The corresponding eigenkets span subspace $\mathcal{H}_D$. Then, the probability $P(D)$ that a measurement of A will give result D is

$$\boxed{P(D) = \langle P_D\rangle = \frac{\langle\psi_D|\psi_D\rangle}{\langle\psi|\psi\rangle}\,.} \tag{1.68}$$

Of course, "$P(D)$" is *not* a projector while "P_D" is. Which "P" is a probability and which a projector will be obvious from the context.

When a state of a quantum system is not completely known (e.g., when it is not a pure), the system has probabilities p_i, $i = 1, 2, \ldots, n$ of being in states $|\psi_i\rangle$, respectively. When we measure a quantity represented by operator A, the probability that we shall obtain the result

$$\langle A\rangle_i = \frac{\langle\psi_i|A|\psi_i\rangle}{\langle\psi_i|\psi_i\rangle}\,; \tag{1.69}$$

is equal to p_i. Thus, the mean value of an arbitrary function $f(A)$ is

$$\langle f(A)\rangle = \sum_{i=1}^{n} p_i \frac{\langle\psi_i|f(A)|\psi_i\rangle}{\langle\psi_i|\psi_i\rangle}\,. \tag{1.70}$$

Equation (1.70) describes a *statistical mixture* of states which means that it cannot be represented as a superposition of states, that is, as a vector $|\psi\rangle = \sum_{i=1}^{n} c_i|\psi_i\rangle$

given by (1.25) even when $c_i = p_i$. Therefore, states from a statistical mixture cannot interfere. A preparation of a statistical mixture is incomplete as opposed to a complete preparation when the initial state of the system is exactly known as for a pure state or a superposition of states.

To obtain a more operational version of (1.70) which will help us to handle statistical mixtures effectively, we introduce notions of the *density operator* and *trace*.

Definition 38 The operator

$$\rho = \sum_{i=1}^{n} |\psi_i\rangle p_i \langle\psi_i| \tag{1.71}$$

is called a *density operator* or a *statistical operator*, provided the following two conditions hold:

$$p_k \geq 0 \,, \quad \sum_{i=1}^{n} p_i = 1 \,. \tag{1.72}$$

Definition 39 A *trace* of an operator A is the sum of its diagonal matrix elements:

$$\mathrm{Tr}(A) = \sum_{i=1}^{n} \langle\psi_i|A|\psi_i\rangle \,. \tag{1.73}$$

A *trace* of a dyad is

$$\mathrm{Tr}(|\phi\rangle\langle\psi|) = \sum_{i=1}^{n} \langle\psi_i|\phi\rangle\langle\psi|\psi_i\rangle \,. \tag{1.74}$$

The following obvious relations hold:

$$\mathrm{Tr}(AB) = \mathrm{Tr}(BA) \,, \quad \mathrm{Tr}(ABC) = \mathrm{Tr}(BCA) = \mathrm{Tr}(CAB) \,, \quad \text{etc.} \tag{1.75}$$

$$[\mathrm{Tr}(A)]^* = \mathrm{Tr}(A^\dagger) \,, \quad [\mathrm{Tr}(AB)]^* = \mathrm{Tr}(B^\dagger A^\dagger) \,,$$
$$[\mathrm{Tr}(ABC)]^* = \mathrm{Tr}(C^\dagger B^\dagger A^\dagger) \,, \quad \text{etc.} \tag{1.76}$$

Lemma 40

$$\mathrm{Tr}(|\phi\rangle\langle\psi|) = \langle\psi|\phi\rangle \,. \tag{1.77}$$

Proof: Follows from (1.31). □

Theorem 41

$$\boxed{\langle f(A)\rangle = \mathrm{Tr}[\rho f(A)]}. \tag{1.78}$$

Proof: We first show $\langle A\rangle = \mathrm{Tr}(\rho A)$ assuming $|\psi_i\rangle$ are normalized to unity. Let us start with $\langle\psi_i|f(A)|\psi_i\rangle$ from (1.70). Applying (1.34), (1.31), (1.75), and (1.59), we obtain

$$\begin{aligned}\langle\psi_i|A|\psi_i\rangle &= \langle\psi_i|A|\psi_i\rangle\mathrm{Tr}(|\psi_i\rangle\langle\psi_i|) = \mathrm{Tr}(|\psi_i\rangle\langle\psi_i|A|\psi_i\rangle\langle\psi_i|)\\ &= \mathrm{Tr}[(|\psi_i\rangle\langle\psi_i|)^2 A] = \mathrm{Tr}(|\psi_i\rangle\langle\psi_i|A).\end{aligned} \tag{1.79}$$

Hence,

$$\mathrm{Tr}(\rho A) = \sum_{i=1}^{n}\mathrm{Tr}(|\psi_i\rangle\langle\psi_i|A) = \sum_{i=1}^{n}\langle\psi_i|A|\psi_i\rangle\ . \tag{1.80}$$

The proof goes through for any function of A and for nonnormalized basis vectors, and thus we obtain (1.70) which proves (1.78). □

The theorem gives the normalization condition $\mathrm{Tr}(\rho) = 1$ for $A = I$.

Corollary 42: The probability $P(D)$ that the result of a measurement is from D is

$$\boxed{P(D) = \mathrm{Tr}(\rho P_D)}. \tag{1.81}$$

The probability of a system being in a state described by ket $|\psi\rangle$ is

$$\boxed{P(\psi) = \frac{\langle\psi|\rho|\psi\rangle}{\langle\psi|\psi\rangle}}. \tag{1.82}$$

Proof: Equation (1.81) follows from Postulate 37 and Theorem 41.
Equation (1.82) follows from (1.71). □

Example 43 CNOT operator, gate, and matrix.

To better understand how operators and states they act upon are related, we shall consider the CNOT gate and the CNOT operator. They will play a central role in building a quantum computer. The CNOT gates are universal in the sense that we can build up any circuit using only them. The CNOT operator and matrix are unitary and reversible.

We shall make use of CNOT in Section 1.18 (see (1.281)), in Section 2.2.3 (see (2.27)), in Section 2.2.4 (see (2.33)), in Section 2.3 (see (2.92)), in Section 2.4 (see Figure 2.26), in Section 2.5 (see (2.130)), in Section 2.6.5 (see (2.149)), in Section 2.7.7 (see (2.219)), and in Section 3.2 (see (3.60)). Here, we only show how we can construct a general CNOT operator.

A CNOT gate leaves the target qubit unchanged whenever the control qubit is $|0\rangle$ and flips it whenever the control qubit is $|1\rangle$.

$$\widehat{\text{CNOT}}|00\rangle = |00\rangle\ , \quad \widehat{\text{CNOT}}|01\rangle = |01\rangle\ ,$$
$$\widehat{\text{CNOT}}|10\rangle = |11\rangle\ , \quad \widehat{\text{CNOT}}|11\rangle = |10\rangle\ . \tag{1.83}$$

This can be written as (see (1.39) and (1.40))[17], where, e. g. the bra of $|01\rangle$ is $\langle 01|$ (steps: $|01\rangle^\dagger = (|0\rangle_1|1\rangle_2)^\dagger = (|0\rangle_1 \otimes |1\rangle_2)^\dagger = {}_1\langle 0| \otimes {}_2\langle 1| = {}_1\langle 0|_2\langle 1| = \langle 01|$):

$$\begin{aligned}\widehat{\text{CNOT}} &= |00\rangle\langle 00| + |01\rangle\langle 01| + |11\rangle\langle 10| + |10\rangle\langle 11| \\ &= \begin{pmatrix}1\\0\\0\\0\end{pmatrix}(1000) + \begin{pmatrix}0\\1\\0\\0\end{pmatrix}(0100) + \begin{pmatrix}0\\0\\0\\1\end{pmatrix}(0010) + \begin{pmatrix}0\\0\\1\\0\end{pmatrix}(0001) \\ &= \begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix}\ .\end{aligned} \tag{1.84}$$

The form the CNOT matrix has in (1.84) therefore depends on the basis in which it is defined ($\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$). So, if we change the basis (e.g., by rotating the axes), our CNOT will not apply to them in the sense of its definition given by (1.83). Thus, if wanted to keep the functionality of the CNOT gate according to its definition in a new basis, we should rotate it too, that is, apply appropriate additional gate manipulations depending on the aforementioned realistic implementation of the circuit within which we implement our CNOT gate.

1.12
Generating Qubits: Sources of Photons – Polarization Primer

In a quantum computer, every qubit supports a vast amount of states that are used to carry out a computation. It proves essential to correlate states of different qubits – we say to *entangle qubits*.

17) In the rest of the book, we shall drop the "hat" over the CNOT operator since it will always be clear from the context whether it appears as an operator or as a gate in a particular realistic implementation within a circuit.

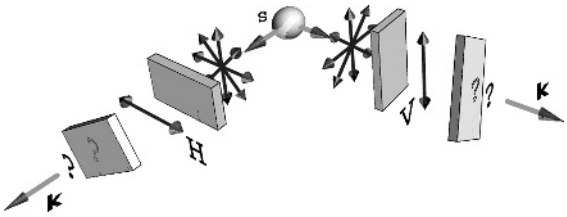


Figure 1.23 Linear polarization. S is a source of unpolarized photons. Filters (polarizers) let through horizontal (H) and vertical (V) linearly polarized photons. The second polarizers (?) let a portion of H and V photons through according the Malus law, but whether a particular photon would pass it is impossible to predict.

In effect, entanglement boils down to picking out particular correlated states from a tensor product of qubit states. To arrive at this result, we should introduce the notion of photon polarization and specify sources we shall use for generation of polarized photons. There is no difference in description of a single free photon and a beam of photons. So, we shall start with electrical vectors from Faraday's electromagnetism. The difference will show up when we entangle photon qubits. Let us start with the linear polarization shown in Figure 1.23

Photons that pass through horizontal (H) and vertical (V) polarizers are horizontally and vertically polarized, respectively, but that does not mean that of all shown vectors in front of the polarizers only H and V are allowed through. Actually, a percentage of all orientations apart from strictly vertical might pass through H and a percentage of all orientations apart from strictly horizontal through V. (Polarized glasses reduce the intensity of unpolarized light by half.) This can be illustrated by the behavior of photons that pass through the second polarizers (denoted by "?" in the figure) rotated at an angle with respect to the first ones. The photons will pass the second polarizer or not in a ratio determined by the Malus law but whether a particular photon sent from H or V to "?" will pass "?" or not is according to the principles of quantum mechanics impossible to predict.

Still, if we combine such unpolarized photons in space (Hanbury Brown–Twiss effect [228])[18] or at a beam splitter, any two of them will become deterministically correlated in polarization, meaning that a passage of one of them through a polarizer will determine that the second photon will pass through an equally oriented polarizer as well. We will say that the two photons are *entangled* in polarization. We shall elaborate on entanglement and precisely define it in Section 1.13. Entanglement will then serve us to achieve teleportation of polarization and actually of any qubit state in Section 1.16.

To formalize polarization, let us consider two perpendicular harmonic electric fields. (Light is electromagnetic radiation and every photon can be described with

18) The effect discovered with photons by Robert Hanbury Brown and Richard Quentin Twiss more than half a century ago [115] has since then been confirmed for various boson and fermion particles [133].

the help of electric field vectors.)

$$\begin{aligned} \boldsymbol{E}_x(\boldsymbol{r},t) &= \boldsymbol{E}_{0x}\, e^{i(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t)} \\ &= \boldsymbol{E}_{0x}[\cos(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t) + i\sin(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t)]\,, \\ \boldsymbol{E}_y(\boldsymbol{r},t) &= \boldsymbol{E}_{0y}\, e^{i(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t+\epsilon)} \\ &= \boldsymbol{E}_{0y}[\cos(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t+\epsilon) + i\sin(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t+\epsilon)]\,, \end{aligned} \tag{1.85}$$

where $\boldsymbol{k}$ ($|\boldsymbol{k}| = 2\pi/\lambda$) is the *wave vector*, also called *propagation vector* (where λ is the wave length); ε is a relative phase; ω is the angular frequency, $\omega = 2\pi\nu$; ν is the frequency of the wave. The wave moves in $+z$ direction.

Vector $\boldsymbol{E}_x$ oscillates in the xz plane and can be considered as a horizontally (H) linearly polarized wave and $\boldsymbol{E}_y$ oscillates in the yz plane and can be considered as a vertically (V) polarized wave. When they propagate along the same propagation vector $\boldsymbol{k}$, they superpose and we obtain

$$\boldsymbol{E}(\boldsymbol{r},t) = \boldsymbol{E}_x(\boldsymbol{r},t) + \boldsymbol{E}_y(\boldsymbol{r},t) = \boldsymbol{E}_{0x}\, e^{i(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t)} + \boldsymbol{E}_{0y}\, e^{i(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t+\epsilon)}\,, \tag{1.86}$$

$$\mathrm{Re}[\boldsymbol{E}(\boldsymbol{r},t)] = \boldsymbol{E}_{0x}\cos(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t) + \boldsymbol{E}_{0y}\cos(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t+\epsilon)\,. \tag{1.87}$$

For $\epsilon = n\pi$, $n = 0, \pm1, \pm2, \ldots$, we obtain diagonally linearly polarized waves ($\mathrm{D}^{\pm}$). For an even n or $n = 0$, we get D^{+} which oscillates in the plane which we get by rotating the plane xz along $\boldsymbol{k}$ by $+45°$,

$$\mathrm{Re}[\boldsymbol{E}_{+45°}(\boldsymbol{r},t)] = (\boldsymbol{E}_{0x} + \boldsymbol{E}_{0y})\cos(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t) \tag{1.88}$$

For an odd n, we get D^{-} which oscillates in the plane which we get by rotating the plane xz along $\boldsymbol{k}$ by $-45°$.

$$\mathrm{Re}[\boldsymbol{E}_{-45°}(\boldsymbol{r},t)] = (\boldsymbol{E}_{0x} - \boldsymbol{E}_{0y})\cos(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t) \tag{1.89}$$

For $\epsilon = \pm(2n-1/2)\pi$, $n = 0, \pm1, \pm2, \ldots$ and $|\boldsymbol{E}_{0x}| = |\boldsymbol{E}_{0y}|$, we obtain *right* (R) and *left* (L) *circularly polarized* waves. For $\epsilon = (2n-1/2)\pi$, we get R wave whose vector $\boldsymbol{E}$ rotates clockwise around the propagation vector $\boldsymbol{k}$ viewed from the direction towards which the wave is approaching at a *fixed position* z – as shown in Figure 1.24. The cosine term containing ε in (1.87) turns into a sine term with the same argument as the other cosine term.

$$\mathrm{Re}[\boldsymbol{E}_{\mathrm{R}}(\boldsymbol{r},t)] = \boldsymbol{E}_{0x}\cos(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t) + \boldsymbol{E}_{0y}\sin(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t) \tag{1.90}$$

For $\epsilon = -(2n-1/2)\pi$, we get L wave whose vector $\boldsymbol{E}$ rotates counterclockwise around $\boldsymbol{k}$ viewed from the direction towards which the wave is approaching at a

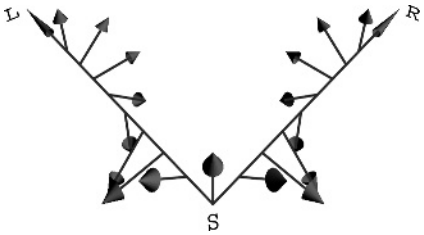


Figure 1.24 Circular polarization. When we view the vectors passing through a fixed location on the right propagating vector from its top we see that each new vector $\boldsymbol{E}_{\mathrm{R}}$ that comes to that location is rotated clockwise with respect to a previous one in time.

fixed position z – as shown in Figure 1.24.[19] The cosine term containing ε in (1.87) turns into a negative sine term with the same argument as the other cosine term.

$$\mathrm{Re}[\boldsymbol{E}_{\mathrm{L}}(\boldsymbol{r},t)] = \boldsymbol{E}_{0x}\cos(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t) - \boldsymbol{E}_{0y}\sin(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t)\,. \tag{1.91}$$

We see that for $|\boldsymbol{E}_{0x}| \neq |\boldsymbol{E}_{0y}|$, (1.86) gives a complete characterization of a *monochromatic plane wave* of circular frequency ω. It corresponds to an elliptically polarized wave and is called the *Jones Vector*. Its matrix representation reads

$$\boldsymbol{E}(\boldsymbol{r},t) = \begin{pmatrix} E_{0x}\,e^{i(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t)} \\ E_{0y}\,e^{i(\boldsymbol{k}\cdot\boldsymbol{r}-\omega t+\epsilon)} \end{pmatrix}. \tag{1.92}$$

The intensity of the wave is

$$I = \frac{1}{2\eta}\left(E_{0x}^2 + E_{0y}^2\right) \tag{1.93}$$

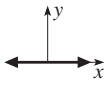
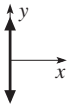
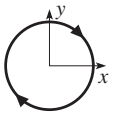
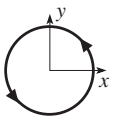
where η is the impedance of the medium.[20]

The electric field vectors, that is, Jones vectors characterize both, a beam of photons and a single photon, in the same way. Therefore, the Jones vectors of the afore-given polarizations, with the intensity normalized to 1, can be considered as basis states.

Definition 44 Polarization ket vectors are defined by the Jones vectors with the intensity normalized to 1 as explained above and summarized below.

19) This is the convention adopted in most physical literature. In the engineering literature, it is the other way around – they view the rotating vector from the source. (Recall the physical electron flow vs. engineering current flow – against electron flow.) So, according to the latter convention, we should exchange R and L indices in (1.90) and (1.91). Also, the Jones matrices in Definition 44 should be exchanged accordingly.

20) If we let a wave through a horizontal filter, it will come out as $\boldsymbol{E}_x$. Its intensity is $I_0 = E_{0x}^2$. Now, let the wave pass through a second filter rotated at an angle α with respect to the first one. It will come out as $\boldsymbol{E}_\alpha$ whose amplitude is the projection $E_{0x}\cos\alpha$ and the intensity is $I_\alpha = I_0\cos^2\alpha$. This is called the *Malus law*. For a single photon which passes the first filter we have $I_0 = 1$ and therefore its probability of passing the second one is $\cos^2\alpha$.

Horizontally polarized	$\|\mathrm{H}\rangle = \|\leftrightarrow\rangle = \|0\rangle = \begin{pmatrix}1\\0\end{pmatrix}$	
Vertically polarized	$\|\mathrm{V}\rangle = \|\updownarrow\rangle = \|1\rangle = \begin{pmatrix}0\\1\end{pmatrix}$	
Diagonally polarized, $+45°$	$\|\mathrm{D}^+\rangle = \|\nearrow\rangle = \|0\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix}$	
Anti-diagonally polarized, $-45°$	$\|\mathrm{D}^-\rangle = \|\searrow\rangle = \|1\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}-1\\1\end{pmatrix}$	
Right circularly polarized	$\|\mathrm{R}\rangle = \|\circlearrowright\rangle = \|0\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\i\end{pmatrix}$	
Left circularly polarized	$\|\mathrm{L}\rangle = \|\circlearrowleft\rangle = \|1\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\-i\end{pmatrix}$	

Any pair of polarization kets (the Jones vectors above) forms a basis which can be used to express any other such vector. For instance,

$$|\mathrm{V}\rangle = \frac{i}{\sqrt{2}}(|\mathrm{L}\rangle - |\mathrm{R}\rangle)\,, \quad |\mathrm{L}\rangle = \frac{1}{\sqrt{2}}(|\mathrm{H}\rangle - i|\mathrm{V}\rangle)\,. \tag{1.94}$$

We can turn linearly polarized photons into circularly polarized ones and back by means of a *quarter wave plate* (QWP). A quarter wave plate is a phase retarder which introduces a phase shift between the vertical and horizontal components of the field. Uniaxial birefringent crystals, such as quartz, serve as phase retarders. They have one crystal axis that is different from the other two crystal axes. The former one is called the *extraordinary* or the *optic axis*. The optic axis of a quartz crystal is a *slow axis*, that is, the axis with the highest refractive index. It is called a slow axis because the phase velocity of light is lower along this axis than along the other two (which are called *fast axes*). The Jones matrix for a quarter plate reads

$$\begin{aligned}\mathrm{QWP}(\theta) &= \begin{pmatrix}\cos^2\theta - i\sin^2\theta & (1+i)\cos\theta\sin\theta\\ (1+i)\cos\theta\sin\theta & -i\cos^2\theta + \sin^2\theta\end{pmatrix}\\ &= \frac{1}{2}\left[(1-i)\,I + (1+i)(\sin 2\theta\,\sigma_x + \cos 2\theta\,\sigma_z)\right]\,,\end{aligned} \tag{1.95}$$

where θ is the angle between the optical axis and the horizontal polarization direction, and σ_x, σ_z Pauli matrices are given by (1.41).

For $\theta = \pi/4$, we get $\mathrm{QWP}(\pi/4) = \sigma_x$ and for $\theta = 0$, we have

$$\mathrm{QWP}(0)|\mathrm{D}^+\rangle = \begin{pmatrix} 1 & 0 \\ 0 & -i \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -i \end{pmatrix} = |\mathrm{L}\rangle$$
$$\mathrm{QWP}(0)|\mathrm{D}^-\rangle = |\mathrm{R}\rangle\,, \quad \mathrm{QWP}(0)|\mathrm{L}\rangle = |\mathrm{D}^-\rangle\,, \quad \mathrm{QWP}(0)|\mathrm{R}\rangle = |\mathrm{D}^+\rangle\,. \tag{1.96}$$

Let us now see how we can get a pair of photons we can entangle. What we need are unpolarized photons. The oldest known source is an atomic spontaneous emission. A distribution of such photons is usually isotropic and we cannot well control them. However, the biggest problem is that we have to bring two of them together so that they can combine and correlate in particular properties, that is, to assure their coherence.

Definition 45 The *coherence* of photon fields radiated from n sources S_i amounts to the stability of phases of their fields in the following sense. Their total field amplitude E at point P is a superposition of partial amplitudes E_i and phases ϕ_i, $i = 1, \ldots, n$ at time t:

$$E(P,t) = \sum_{i=1}^{n} E_i(P,t) e^{i\phi_i(P,t)} = \sum_{i=1}^{n} \frac{E_i(0,t)}{r_i^2} e^{i\phi_{0i} + i\omega t + 2\pi r_n/\lambda} \tag{1.97}$$

where r_i are radius vectors from S_i to P and ϕ_{0i} is the phase of the partial wave at S_i. We say that the field is *temporally coherent* if $\phi_1(P,t_1) - \phi_1(P,t_2) \approx \ldots \approx \phi_n(P,t_1) - \phi_n(P,t_2) = \Delta\phi_n$ within the time period $\Delta t = t_2 - t_1$ which is called the *coherence time*; the coherence time can also be expressed as $\Delta t = 1/\Delta\nu$. It is the time within which the superposition (1.97) can result in interference phenomena. The length $\Delta s_c = c\Delta t$ traveled by the wave during the coherence time Δt is called the *coherence length*. The phases include ω, so the frequency must be stable too). If the phases do not change in time for a considered period of time and $\phi(P_i,t) \neq \phi(P_j,t)$ for two different ($i \neq j$) points P_i and P_j, we say that the sources are *spatially coherent*. The latter points which have nearly the same optical path difference from the source form the *coherence volume*. The interference phenomena can be observed only within the coherence volume.

A low temporal coherence is usually caused by the nonmonochromaticity of photons and a low spatial coherence and coherence volume is related to the geometry of modes.

Thus, to have interference with two photons from an atomic source, we should have well-defined positions of atoms, frequency of photons, and above all, a narrow time window.

To explain the entanglement, we need unpolarized photons and photons that we can obtain by spontaneous or stimulated emission are polarized. Photons that we can get in an atomic cascade emission shown in Figure 1.25 are unpolarized and are already entangled. Besides, they are of different frequencies. We could combine

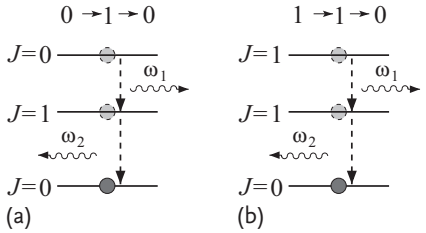


Figure 1.25 Cascade photon emissions. (a) A cascade with one initial and one final state, for example, ^{40}Ca: $4p^2\,{}^1S_0 \rightarrow 4s4p\,{}^1P_1 \rightarrow 4p^2\,{}^1S_0$. The photons are entangled: $|\psi\rangle = |H\rangle_1|H\rangle_2 + |V\rangle_1|V\rangle_2$ and therefore unpolarized; (b) A cascade with a superposition of three states as its initial state, for example, ^{200}Hg: $7^3S^1 \rightarrow 6^3P^1 \rightarrow 6^1S_0$. The photons are entangled: $|\psi\rangle = |H\rangle_1|V\rangle_2 + |V\rangle_1|H\rangle_2$ and therefore unpolarized.

two cascades, but then we should use ultrafast detectors in a very narrow time window. (We still need the cascade sources for some other purpose.)

This is why today we mostly entangle photons by means of *parametric down-conversion* (the inverse of parametric generation) which is a quantum effect in nonlinear optics. In a nonlinear medium – a particular crystal – an intense electric field (laser pump beam) of one frequency (ω_0) generates two photons of other frequencies (ω_1 and ω_2) called *signal* and *idler*.[21] Energy ($E = h\nu$) conservation yields $\omega_0 = \omega_1 + \omega_2$, while momentum conservation implies the phase-matching condition $\boldsymbol{p}_0 = \boldsymbol{p}_1 + \boldsymbol{p}_2 \Rightarrow \boldsymbol{k}_0 = \boldsymbol{k}_1 + \boldsymbol{k}_2$.

We distinguish two major types of down-conversion:

- *Type-I down-conversion* (Figure 1.26) (KDP crystals, for example, $AgGaSe_2$) and
- *Type-II down-conversion* (Figure 1.27) (BBO crystals; beta-barium-borate, β-BaB_2O_4).

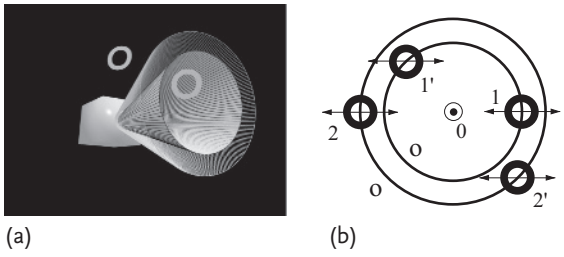


Figure 1.26 Type-I down-conversion. Two cones in (a) contain signal and idlers of two different frequencies, but with the same linear polarization (both photons are *ordinarily* (o) polarized). The cones are inside each other, except when the frequencies are the same – then the cones coincide, but signals and idlers are always at the opposite sides of the optical axis, as shown in (b) with horizontal polarization.

21) "Signal" and "idler" are just two down-converted photons. We never say which is which.

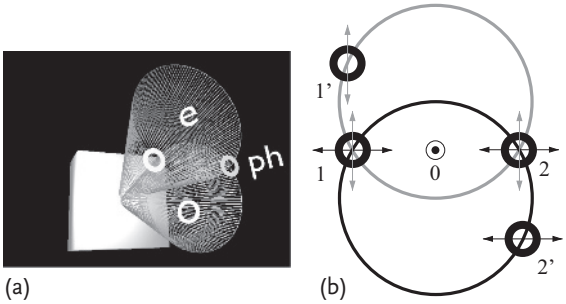


Figure 1.27 Type-II down-conversion. Two cones in (a) contain signals and idlers of two different frequencies with different linear polarization (*extraordinary* (e) and *ordinary* (o) photons). The cones intersect each other. When the frequencies are the same, then the opening angle of the cones are equal. The signals and idlers are always at the opposite sides of the optical axis (O) as shown in (b). The photons 1 and 2 that pass through pinholes ph are entangled $|\psi\rangle = |\mathrm{H}\rangle_1|\mathrm{V}\rangle_2 + e^{i\phi}|\mathrm{V}\rangle_1|\mathrm{H}\rangle_2$ and unpolarized.

We see that we cannot obtain the unpolarized photons by a type-I down-conversion as is. However, we can put two type-I crystals atop of each other and rotate one with respect to the other at 90° [160]. With a pump beam polarized at 45° with respect to both (perpendicular) planes defined by the pump beam and optical axis of each crystal, we might get a down-conversion from either of the crystals. When the crystals of its origin are indistinguishable from each other, we shall have the entanglement $|\psi\rangle = |\mathrm{H}\rangle_1|\mathrm{H}\rangle_2 + e^{i\phi}|\mathrm{V}\rangle_1|\mathrm{V}\rangle_2$ and therefore unpolarized individual photons.

In type-II crystals, photons 1 and 2 coming out through the pinholes *ph* positioned at the intersection of two cones as shown in Figure 1.27 *are* are unpolarized and of the same frequency. They too are already entangled, but if we wanted to have unpolarized photons that are not entangled, we could take unpolarized photons from two separately down-converted pairs – controlling such two pairs is experimentally feasible today. The same applies to the aforementioned glued type-I crystals.

1.13
Correlating Unpolarized Qubits: Quantum Entanglement

So, we now know how to obtain unpolarized photons and we can explain how to entangle them in a controlled manner that we would need for quantum computation. Many essential properties and devices of quantum computation and communication such as circuits, gates, teleportation, repeater, error correction, and so on, are enabled by a particular kind of correlation between the qubit states, that is, by *entanglement* of qubits. Entanglement enhances properties of qubits by selecting states to be used for computation. To understand that, let us have a look at the setup shown in Figure 1.28.

We shall first assume that photons 1 and 2 are polarized in some particular planes and only after we carry out our calculations, shall we randomize the orientations of the planes so as to obtain the results for unpolarized photons.

The state of polarized photons immediately after leaving the sources is described by the product of two prepared linear-polarization states:

$$|\Psi\rangle = (\cos\theta_{1'}|0\rangle_1 + \sin\theta_{1'}|1\rangle_1) \otimes (\cos\theta_{2'}|0\rangle_2 + \sin\theta_{2'}|1\rangle_2) \ , \tag{1.98}$$

where $|0\rangle$ and $|1\rangle$ denote the mutually orthogonal photon states. So, for example, $|0\rangle_1$ means the state of a photon leaving the upper source polarized in direction x. If the beam splitter were removed, it would cause a "*click*" at the detector D_1 and no "*click*" at the detector $D_1^{\perp}$, provided the lover polarizing beam splitter (PBS) is oriented along x. Here, $D_1^{\perp}$ means a detector counting photons coming out at the other exit of the PBS. Angles $\theta_{1'}$, $\theta_{2'}$ are the angles along which incident photons are polarized with respect to a fixed direction.

For unpolarized photons, the density matrix is proportional to the unit matrix and this means that we only need products $|0\rangle_1|0\rangle_2 = |00\rangle$, $|0\rangle_1|1\rangle_2 = |01\rangle$, $|1\rangle_1|0\rangle_2 = |10\rangle$, and $|1\rangle_1|1\rangle_2 = |11\rangle$ to form partial probabilities which then sum up to the total correlation probability as used below.

To describe the interaction of photons with the beam splitter, polarizers and detectors, we use the quantized electric field operators often employed in quantum optical analysis [217, 220, 221],

$$\hat{E}_j(\boldsymbol{r}_j, t) = \hat{a}(\omega_j)e^{i\boldsymbol{k}_j\cdot\boldsymbol{r}_j - i\omega_j t} \ , \tag{1.99}$$

where $\hat{a}$ is an annihilation operator and where $j = 1, 2$. We tacitly assume that photons arrive at the beam splitter practically simultaneously, that is, with appropriate short delays. The annihilation operators like $\hat{a}$ in (1.99) describe joint actions of polarizers, beam splitter, and detectors. They act on the states as follows: $\hat{a}_{1x}|0\rangle_1 = |\varnothing\rangle_1$, $\hat{a}^{\dagger}_{1x}|\varnothing\rangle_1 = |0\rangle_1$, $\hat{a}_{1x}|\varnothing\rangle_1 = 0$, and so on, where $|\varnothing\rangle$ is the vacuum state which we take here simply as a detection of qubits. The annihilation operators and the vacuum state terms belong to the so-called Fock state formalism. We will not define the Fock space itself because all we need from it is the simple algebra

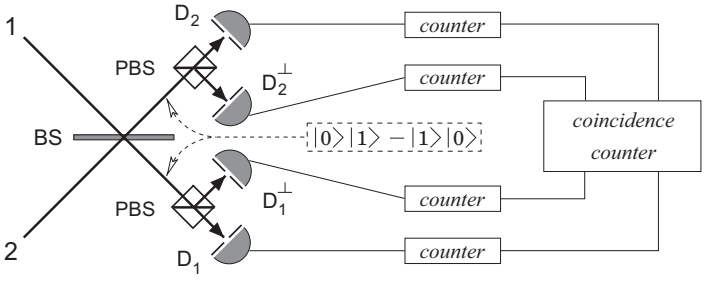


Figure 1.28 Two photon interference at a beam splitter. BS is a beam splitter. PBS are polarizing beam splitters. They let horizontally polarized photons through and into detectors D and reflect vertically polarized ones to detectors $D^{\perp}$.

of the aforementioned examples. For example, the example $\hat{a}_{1x}|0\rangle_1 = |\varnothing\rangle_1$ simply means that we detected state $|0\rangle$ and obtained a "click" corresponding to state $|\varnothing\rangle$ in our detector. The algebra is used to deal with products of states. For instance, if we act with $\hat{a}_{1x}\hat{a}_{2x}$ on $|0\rangle_1|0\rangle_2$, we will get $\hat{a}_{1x}\hat{a}_{2x}|0\rangle_1|0\rangle_2 = |\varnothing\rangle_1|\varnothing\rangle_2$ which means a detection of $|0\rangle_1|0\rangle_2$. However, if we act on this state with $\hat{a}_{1x}\hat{a}_{1x}$, we will get $\hat{a}_{1x}\hat{a}_{1x}|0\rangle_1|0\rangle_2 = 0$. That means that all the terms that contain $\hat{a}_{1x}\hat{a}_{1x}$ will cancel out when applied to the above state. We shall use this algebra to calculate our expressions below.

To take the linear polarization along orthogonal directions into account, we shall consider two perpendicular axes $0x$ and $0y$ to which the polarizers are rotated by an angle θ. The action of the beam splitter we describe by the input annihilation operators $\hat{a}_{1\text{in}}$ and $\hat{a}_{2\text{in}}$ and the following output ones:

$$\begin{aligned} \hat{a}_{1x-\text{out}} &= t_x\hat{a}_{1x-\text{in}} + ir_x\hat{a}_{2x-\text{in}}\,, \qquad \hat{a}_{1y-\text{out}} = t_y\hat{a}_{1y-in} + ir_y\hat{a}_{2y-\text{in}}\,, \\ \hat{a}_{2x-\text{out}} &= ir_x\hat{a}_{1x-\text{in}} + t_x\hat{a}_{2x-\text{in}}\,, \qquad \hat{a}_{2y-\text{out}} = ir_y\hat{a}_{1y-\text{in}} + t_y\hat{a}_{2y-\text{in}}\,, \end{aligned} \tag{1.100}$$

where $t = |\sqrt{T}|$ and $r = |\sqrt{R}|$, where T and R denote transmittance and reflectance, respectively. An elaboration of a general case is given in [234].

The action of the polarizers P_1, P_2 and detectors D_1, D_2 can then be expressed as

$$\hat{a}_{j-\text{out}} = \hat{a}_{jx-\text{out}}\cos\theta_j + \hat{a}_{jy-\text{out}}\sin\theta_j\,, \tag{1.101}$$

where $j = 1, 2$.

Projections corresponding to the other choices of polarizers and detectors are obtained by using appropriate transformations instead of the ones given by (1.103). For example, we obtain the action of the polarizer $P_2^{\perp}$ (orthogonal outgoing port of the lower PBS) and the corresponding detector $D_2^{\perp}$ if we substitute

$$\hat{a}_{2-\text{out}} = -\hat{a}_{2x-\text{out}}\sin\theta_2 + \hat{a}_{2y-\text{out}}\cos\theta_2 \tag{1.102}$$

for (1.101) with $j = 2$.

Below, we drop subscripts "in" and "out" (to ease the notation) since the meaning of the annihilation operators is clear from the content.

Hence, the appropriate outgoing electric field operators are

$$\begin{aligned} \hat{E}_1 &= (\hat{a}_{1x}t_x\cos\theta_1 + \hat{a}_{1y}t_y\sin\theta_1)\eta_{11} + i(\hat{a}_{2x}r_x\cos\theta_1 + \hat{a}_{2y}r_y\sin\theta_1)\eta_{12}\,, \\ \hat{E}_2 &= i(\hat{a}_{1x}r_x\cos\theta_2 + \hat{a}_{1y}r_y\sin\theta_2)\eta_{21} + (\hat{a}_{2x}t_x\cos\theta_2 + \hat{a}_{2y}t_y\sin\theta_2)\eta_{22}\,, \end{aligned} \tag{1.103}$$

where

$$\begin{aligned} \eta_{11} &= e^{i\boldsymbol{k}_1\cdot\boldsymbol{r}_1 - i\omega_1(t-\tau_1)}\,, \quad \eta_{12} = e^{i\tilde{\boldsymbol{k}}_2\cdot\boldsymbol{r}_1 - i\omega_2(t-\tau_2)}\,, \\ \eta_{21} &= e^{i\tilde{\boldsymbol{k}}_1\cdot\boldsymbol{r}_2 - i\omega_1(t-\tau_1)}\,, \quad \eta_{22} = e^{i\boldsymbol{k}_2\cdot\boldsymbol{r}_2 - i\omega_2(t-\tau_2)}\,, \end{aligned} \tag{1.104}$$

where τ_j is the time delay after which the photon reaches detector D, ω_j is the frequency of photon j, and c is the velocity of light. The detectors and the crystal are assumed to be positioned symmetrically with regard to the beam splitter, as shown in Figure 1.29, so that two time delays suffice. The wave vectors $\tilde{\boldsymbol{k}}$, $\boldsymbol{k}$, $k = \omega/c$, and radius vectors $\boldsymbol{r}$ (r is the path length from the source to the detectors) are also shown in Figure 1.29.

The joint interaction of both photons with the beam splitter, polarizers P_1, P_2, and detectors D_1, D_2 is given by the following projection of our wave function onto the detection vacuum:

$$\begin{aligned}\hat{E}_1\hat{E}_2|\Psi\rangle = \Big[&\left(t_x^2\varepsilon_{12} - r_x^2\tilde{\varepsilon}_{12}\right)\cos\theta_{1'}\cos\theta_{2'}\cos\theta_1\cos\theta_2 \\ &+ \left(t_x t_y\varepsilon_{12}\sin\theta_1\cos\theta_2 - r_x r_y\tilde{\varepsilon}_{12}\cos\theta_1\sin\theta_2\right)\sin\theta_{1'}\cos\theta_{2'} \\ &+ \left(t_x t_y\varepsilon_{12}\cos\theta_1\sin\theta_2 - r_x r_y\tilde{\varepsilon}_{12}\sin\theta_1\cos\theta_2\right)\cos\theta_{1'}\sin\theta_{2'} \\ &+ \left(t_y^2\varepsilon_{12} - r_y^2\tilde{\varepsilon}_{12}\right)\sin\theta_{1'}\sin\theta_{2'}\sin\theta_1\sin\theta_2\Big]\,\varepsilon|\varnothing\rangle\ , \end{aligned} \tag{1.105}$$

where

$$\begin{aligned}\varepsilon &= \exp[-i\omega_1(t-\tau_1) - \omega_2(t-\tau_2)]\ , \\ \varepsilon_{12} &= \eta_{11}\eta_{22}\varepsilon^{-1} = \exp[i(\boldsymbol{k}_1\cdot\boldsymbol{r}_1 + \boldsymbol{k}_2\cdot\boldsymbol{r}_2)]\ , \\ \tilde{\varepsilon}_{12} &= \eta_{12}\eta_{21}\varepsilon^{-1} = \exp[i(\tilde{\boldsymbol{k}}_1\cdot\boldsymbol{r}_2 + \tilde{\boldsymbol{k}}_2\cdot\boldsymbol{r}_1)]\ . \end{aligned} \tag{1.106}$$

The corresponding probability of detecting the photons by detectors D_1, D_2 is thus

$$P(\theta_{1'},\theta_{2'},\theta_1,\theta_2) = \langle\hat{E}_2^\dagger\hat{E}_1^\dagger\hat{E}_1\hat{E}_2\rangle = A^2 + B^2 - 2AB\cos\phi\ , \tag{1.107}$$

where

$$\begin{aligned}A &= t_x^2\cos\theta_{1'}\cos\theta_{2'}\cos\theta_1\cos\theta_2 + t_y^2\sin\theta_{1'}\sin\theta_{2'}\sin\theta_1\sin\theta_2 \\ &\quad + t_x t_y\left(\cos\theta_{1'}\sin\theta_{2'}\cos\theta_1\sin\theta_2 + \sin\theta_{1'}\cos\theta_{2'}\sin\theta_1\cos\theta_2\right)\ , \\ B &= r_x^2\cos\theta_{1'}\cos\theta_{2'}\cos\theta_1\cos\theta_2 + r_y^2\sin\theta_{1'}\sin\theta_{2'}\sin\theta_1\sin\theta_2 \\ &\quad + r_x r_y\left(\cos\theta_{1'}\sin\theta_{2'}\sin\theta_1\cos\theta_2 + \sin\theta_{1'}\cos\theta_{2'}\cos\theta_1\sin\theta_2\right)\ , \end{aligned} \tag{1.108}$$

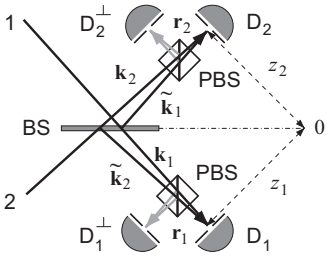


Figure 1.29 Measuring two photon interference at a beam splitter. For symmetric directions of outcoming photons, that is, for a symmetric positioning of detectors, we obtain a maximal entanglement for a 50 : 50 beam splitter.

$$\phi = i(\arg[\varepsilon_{12}] - \arg[\tilde{\varepsilon}_{12}]) = (\tilde{\boldsymbol{k}}_2 - \boldsymbol{k}_1)\cdot\boldsymbol{r}_1 + (\tilde{\boldsymbol{k}}_1 - \boldsymbol{k}_2)\cdot\boldsymbol{r}_2 = 2\pi\frac{z_2 - z_1}{L}\ , \quad (1.109)$$

where L is the spacing of the interference fringes as shown in Figure 1.29 [215]. Phase ϕ can be changed by moving the detectors transversely to the incident beams; arg[·] simply means the argument of an expression – in our case $\arg[\varepsilon_{12}] = \arg[e^{i(\boldsymbol{k}_1\cdot\boldsymbol{r}_1 + \boldsymbol{k}_2\cdot\boldsymbol{r}_2)}] = i(\boldsymbol{k}_1\cdot\boldsymbol{r}_1 + \boldsymbol{k}_2\cdot\boldsymbol{r}_2)$, and so on.

In the following, we shall only consider symmetric locations of the detectors so as to have $\cos\phi = 1$. This corresponds to the same angles that both outcoming photons make with respect to the beam splitter plane. We shall only consider a 50 : 50 beam splitter, that is, $R = T = 1/2$. The probability now reads

$$P(\theta_{1'}, \theta_{2'}, \theta_1, \theta_2) = (A - B)^2 = \frac{1}{4}\sin^2(\theta_{2'} - \theta_{1'})\sin^2(\theta_2 - \theta_1)\ . \quad (1.110)$$

We see that the probability unexpectedly factorizes left to right and not up to down as one would be tempted to conjecture from the initial up–down independence expressed by the product of the "upper" and "lower" function in (1.98). We get the maximal correlation and maximal probability of detecting both photons for a mutually perpendicular orientation of both, polarization of incoming photons and polarizers that filter outcoming photons. The left–right factorization of (1.110) has been verified experimentally as shown in Figure 1.30.

This gives us the idea that maximal correlation of detections given by the condition $\theta_2 - \theta_1 = \pi/2$ might not depend on the polarizations of the incoming photons at all. To verify the conjecture, we proceed as follows.

To obtain the general probability for unpolarized light, $\hat{E}_1$, $\hat{E}_2$ given by (1.103) should be applied to $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$ so as to give four probabilities which

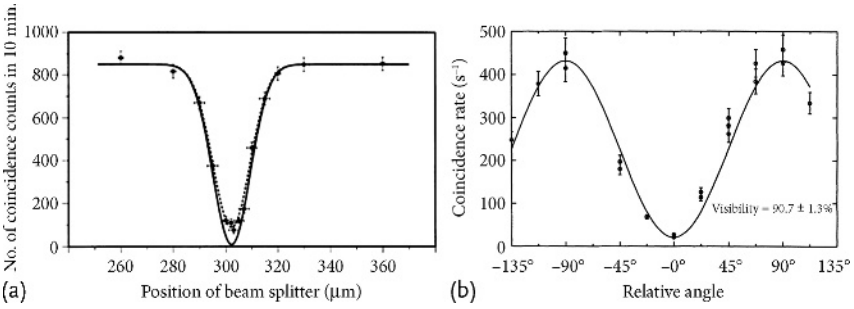


Figure 1.30 Hong–Ou–Mandel dip. A historical figure (a) is reprinted from [125] with permission from the authors; © 1987, American Physical Society; It shows an experimental verification of (1.110). The dip is obtained for the coincidence rate for $\theta_{2'} = \theta_{1'}$; no polarizers were put in front of detectors D_1 and D_2. (b) shows a plot of coincidence rate vs. relative angle of polarizers P_1 and P_2 obtained for $\theta_{2'} - \theta_{1'} = \pi/2$ reprinted from [159] with permission from the authors, © 1992, American Physical Society; see also [216].

then sum up to the following correlation probability:

$$P(\infty,\infty,\theta_1,\theta_2) = \frac{1}{4}\left(t_x^2\cos^2\theta_1 + t_y^2\sin^2\theta_1\right)\left(t_x^2\cos^2\theta_2 + t_y^2\sin^2\theta_2\right) + \frac{1}{4}\left(r_x^2\cos^2\theta_1 + r_y^2\sin^2\theta_1\right)\left(r_x^2\cos^2\theta_2 + r_y^2\sin^2\theta_2\right) - \frac{1}{2}\left(t_x^2 r_x^2\cos\theta_1\cos\theta_2 + t_y^2 r_y^2\sin\theta_1\sin\theta_2\right)^2\cos\phi \ . \tag{1.111}$$

For a 50 : 50 beam splitter and for $\phi = 0$, we get

$$P(\infty,\infty,\theta_1,\theta_2) = P_\infty(\theta_1,\theta_2) = \frac{1}{8}\sin^2(\theta_2-\theta_1) \ , \tag{1.112}$$

amounting to the following conclusion.

Unpolarized photons incident on opposites sides of a beam splitter appear correlated in polarization so that their probability of passing the polarizers P_1 *and* P_2 *is given by* (1.112) *whenever they emerge from the opposite sides of the beam splitter.*

Instead of looking at detections and projections onto the vacuum state of the detectors, we can look at the states of the outcoming photons. We do that by applying electric field operators (given by (1.103)) "in reverse" (*after* the beam splitter) to vacuum states, using the creation operators $a_{1x}^\dagger$, $a_{1y}^\dagger$, $a_{2x}^\dagger$, and $a_{2y}^\dagger$ (that work like this: $a_{1x}^\dagger|\varnothing\rangle = |0\rangle_1$, and so on), taking $t_x = t_y = r_x = r_y = 1/\sqrt{2}$, $\tilde{\varepsilon}_{12} = \varepsilon_{12}$ (i.e., $\phi = 0$), and dropping the overall coefficients $\varepsilon_{12}, \varepsilon$.

$$\hat{E}_{10}^\dagger\hat{E}_{20}^\dagger|\varnothing\rangle = \frac{1}{2}\sin(\theta_2-\theta_1)(|01\rangle - |10\rangle) \ . \tag{1.113}$$

Term $\sin(\theta_2-\theta_1)$ is here for possible orientation of polarizers. In order to most efficiently detect $|\Psi^-\rangle$ (see (1.114)), the detectors must be mutually orthogonal and therefore we must have $\theta_2 - \theta_1 = \pi/2$. The term also shows that the coincidence detection of the photons in the obtained state would remain unaffected by a rotation of polarizers, if we kept their mutual orientation orthogonal. So, after normalization, the state reads:

$$|\Psi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle) \ . \tag{1.114}$$

The state given by (1.114) is called a *singlet* state in analogy to the singlet state of two electrons in a chemical bond. Both states, of pairs of electrons and photons, are formally described by a wave function which is antisymmetrical with respect to exchange of particles within their respective pairs.

Unpolarized photons incident on opposites sides of a beam splitter appear in a singlet state whenever they emerge from the opposite sides of the beam splitter – provided the beam splitter preserves polarization as, for example, a metallic one (see Property 46 in Section 1.14).

The calculation that led us to (1.113) included a cancellation of coefficients of $|00\rangle$ and $|11\rangle$ states. This was a result of a particular form of electric field operators that described only photons emerging from the opposite sides of the beam splitter. To obtain these states, we have to use the electric field operators that describe the photons that emerge from the same sides of the beam splitter.

We employ four detectors in each arm: $D_{2\omega_1}$–$D^{\perp}_{2\omega_2}$ and $D_{1\omega_1}$–$D^{\perp}_{1\omega_2}$, in the upper and lower arm, respectively. Let us do the calculation for the upper arm. Instead of $\hat{E}_1$ from (1.103), we must use

$$\hat{E}'_2 = i\left(\hat{a}_{1x}\, r_x \cos\theta_1 + \hat{a}_{1y}\, r_y \sin\theta_1\right)\eta'_{21} + \left(\hat{a}_{2x}\, t_x \cos\theta_1 + \hat{a}_{2y}\, t_y \sin\theta_1\right)\eta'_{22}\,, \tag{1.115}$$

where

$$\eta'_{21} = e^{i\tilde{\boldsymbol{k}}'_1\cdot\boldsymbol{r}'_2 - i\omega_1(t-\tau_1)}\,, \quad \eta'_{22} = e^{i\boldsymbol{k}'_2\cdot\boldsymbol{r}'_2 - i\omega_2(t-\tau_2)}\,, \tag{1.116}$$

so as to obtain the following analogue of (1.105)

$$\begin{aligned}\hat{E}'_2\hat{E}_2|\Psi\rangle = \big[&t_x r_y\left(\varepsilon_{22}\cos\theta_1\sin\theta_2 + \varepsilon'_{22}\sin\theta_1\cos\theta_2\right)\sin\theta_{1'}\cos\theta_{2'}\\ &+\left(\varepsilon_{22}+\varepsilon'_{22}\right)\left(t_x r_x\cos\theta_{1'}\cos\theta_{2'}\cos\theta_1\cos\theta_2\right.\\ &+\left. t_y r_y\,\sin\theta_{1'}\sin\theta_{2'}\sin\theta_1\sin\theta_2\right)\\ &+ t_y r_x\left(\varepsilon_{22}\sin\theta_1\cos\theta_2 + \varepsilon'_{22}\cos\theta_1\sin\theta_2\right)\\ &\times\ \cos\theta_{1'}\sin\theta_{2'}\big]\,\varepsilon|0\rangle\,,\end{aligned} \tag{1.117}$$

where

$$\begin{aligned}\varepsilon &= \exp\{-i[\omega_1(t-\tau_1)+\omega_2(t-\tau_2)]\}\,,\\ \varepsilon_{22} &= \exp\left[i\left(\boldsymbol{k}_2\cdot\boldsymbol{r}_2 + \tilde{\boldsymbol{k}}'_1\cdot\boldsymbol{r}'_2\right)\right]\,,\\ \varepsilon'_{22} &= \exp\left[i\left(\boldsymbol{k}'_2\cdot\boldsymbol{r}'_2 + \tilde{\boldsymbol{k}}_1\cdot\boldsymbol{r}_2\right)\right]\,.\end{aligned} \tag{1.118}$$

Similarly to (1.109), we have

$$\psi = \left(\tilde{\boldsymbol{k}}_1 - \boldsymbol{k}_2\right)\cdot\boldsymbol{r}_2 + \left(\boldsymbol{k}'_2 - \tilde{\boldsymbol{k}}'_1\right)\cdot\boldsymbol{r}'_2 = 2\pi\frac{Z_2 - Z'_2}{L}\,. \tag{1.119}$$

A general solution is given in [234]. For a 50 : 50 beam splitter and $\psi = 0$, the probability that both photons would emerge from the same side of the beam

splitter and be detected by detectors D reads

$$P(\theta_{1'}, \theta_{2'}, \theta_1 \times \theta_2) = \frac{1}{8}[\cos(\theta_{1'} - \theta_2)\cos(\theta_{2'} - \theta_1) + \cos(\theta_{1'} - \theta_1)\cos(\theta_{2'} - \theta_2)]^2 \,. \tag{1.120}$$

The overall probability that unpolarized incident photons would emerge from the same side of the beam splitter and be detected by detectors D is

$$P(\infty, \infty, \theta_1 \times \theta_2) = P_\infty(\theta_1 \times \theta_2) = \frac{1}{8}\left[1 + \cos^2(\theta_1 - \theta_2)\right] \,. \tag{1.121}$$

This probability and $P_\infty(\theta_1, \theta_2)$, given by (1.112), add up to 1/4.

On the other hand, a probability that we shall have clicks in all four detectors for photons emerging from the opposite and from the same sides of the beam splitters are

$$\begin{aligned} P_\infty\left(\begin{array}{c}\nearrow\\ \searrow\end{array}\right) &= P_\infty(\theta_1, \theta_2) + P_\infty\left(\theta_1^\perp, \theta_2\right) + P_\infty\left(\theta_1, \theta_2^\perp\right) + P_\infty\left(\theta_1^\perp, \theta_2^\perp\right) \\ &= \frac{2}{8}\left[\cos^2(\theta_1 - \theta_2) + \sin^2(\theta_1 - \theta_2)\right] = \frac{1}{4} \end{aligned} \tag{1.122}$$

and

$$\begin{aligned} P_\infty\left(\begin{array}{c}\rightarrow\\ \rightarrow\end{array}, \begin{array}{c}\nearrow\\ \nearrow\end{array}\right) &= P_\infty(\theta_1 \times \theta_2) + P_\infty\left(\theta_1^\perp \times \theta_2\right) + P_\infty\left(\theta_1 \times \theta_2^\perp\right) \\ &\quad + P_\infty\left(\theta_1^\perp \times \theta_2^\perp\right) \\ &= \frac{2}{8}\left[1 + \cos^2(\theta_1 - \theta_2) + 1 + \sin^2(\theta_1 - \theta_2)\right] = \frac{3}{4} \,, \end{aligned} \tag{1.123}$$

respectively.

We see that the probability of photons emerging from the opposite sides of a beam splitter is 25% and from the same side of it is 75%. When the incident photons are polarized, we obtain the distributions shown in Figure 1.31.

Apart from the singlet state given by (1.114), we can engineer three symmetric triplet states. See (1.191)–(1.193) in Section 1.15.

$$|\Psi^+\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) \,, \quad |\Phi^\pm\rangle = \frac{1}{\sqrt{2}}(|00\rangle \pm |11\rangle) \,. \tag{1.124}$$

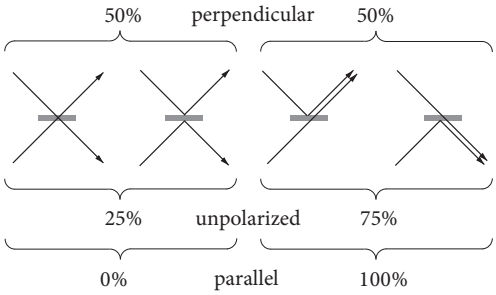


Figure 1.31 Beam splitter output photon distribution for unpolarized, parallelly polarized, and perpendicularly polarized incident photons. See (1.110), (1.112), (1.120), (1.122), and (1.123).

They are named *triplet* states again in analogy to the triplet state of two electrons in a chemical bond. Both, electrons and photons, are formally described by a wave function which is symmetrical with respect to the exchange of particles within a pair.

Singlet state $|\Psi^-\rangle$ and triplet states $|\Psi^+\rangle$ and $|\Phi^\pm\rangle$ together are called the *Bell states*. They form an orthonormal basis in a Hilbert space of two qubits; that means that every pure two-qubit state can be written as their superposition. For instance,

$$|00\rangle = \frac{1}{\sqrt{2}}(|\Phi^+\rangle + |\Phi^-\rangle)\,, \quad |01\rangle = \frac{1}{\sqrt{2}}(|\Psi^+\rangle + |\Psi^-\rangle)\,,$$
$$|10\rangle = \frac{1}{\sqrt{2}}(|\Psi^+\rangle - |\Psi^-\rangle)\,, \quad |11\rangle = \frac{1}{\sqrt{2}}(|\Phi^+\rangle - |\Phi^-\rangle)\,. \tag{1.125}$$

When we measure Bell states with the help of linear optical devices (analyzers, beam splitters, half wave plates, single-photon detectors, and so on), only the product states $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$ can be detected. This requires a certain level of sophistication. For example, we cannot differentiate Ψ^- from Ψ^+ only by means of two detectors behind two analyzers because the projectors to the states $|01\rangle$ and $|10\rangle$, that is, $P_{01} = |01\rangle\langle 10|$ and $P_{10} = |10\rangle\langle 01|$ give the following equal probabilities for both of them for the same experimental arrangement. From (1.68), we obtain

$$P(01) = P_{01} = \langle\Psi^-|01\rangle\langle 10|\Psi^-\rangle = \langle\Psi^+|01\rangle\langle 10|\Psi^+\rangle = \frac{1}{2}$$
$$P(10) = P_{10} = \langle\Psi^-|10\rangle\langle 01|\Psi^-\rangle = \langle\Psi^+|10\rangle\langle 01|\Psi^+\rangle = \frac{1}{2}\,, \tag{1.126}$$

where we made use of $\langle 01|10\rangle = 1$, $\langle 01|01\rangle = 0$, and so on. How we can distinguish the Bell states from each other shall be explained in the next section.

But, from the form of $|\Psi^-\rangle$ we can see that, for example, a horizontal polarization measurement on one of its photons (qubits) (say the first one; the left one in Figure 1.32) along an arbitrary angle

$${}_1\langle 0|\Psi^-\rangle = \frac{1}{\sqrt{2}}{}_1\langle 0|(|0\rangle_1|1\rangle_2 + |1\rangle_1|0\rangle_2) = \frac{1}{\sqrt{2}}|1\rangle_2\,; \tag{1.127}$$

will leave the other photon vertically polarized. And, no matter which angle we chose, a measurement of the other photon will always confirm that it is in the state which is perpendicular to the one we found at the first photon.

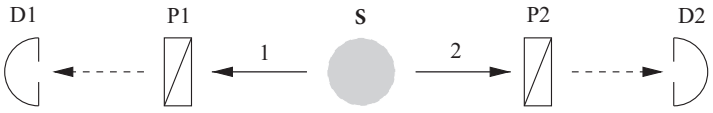


Figure 1.32 Photons entangled in state $|\Psi^-\rangle$. Measurement carried out on photon 1 with the polarizer oriented along any angle θ predetermines with certainty the outcome of a measurement carried out on photon 2 with the polarizer oriented along $\theta \pm \pi/2$.

We also cannot separate the Bell states, that is, we cannot represent them as a product of individual qubit states. To see this, let us assume the opposite for $|\Psi^-\rangle$

$$|\Psi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle) = (\alpha|0\rangle_1 + \beta|1\rangle_1) \otimes (\gamma|0\rangle_2 + \delta|1\rangle_2)$$
$$= \alpha\gamma|00\rangle + \alpha\delta|01\rangle + \beta\gamma|10\rangle + \beta\delta|11\rangle \,, \tag{1.128}$$

wherefrom we get $\alpha\gamma = \beta\delta = 0$ and $\alpha\delta = -\beta\gamma = \sqrt{2}/2$ which is a contradiction.

The last two properties of the Bell states are summarized as follows for a singlet state.

1. Whenever we let one of the two qubits (a photon) from a singlet through an analyzer oriented along an arbitrary angle θ and detect it, then we immediately know the other photon will pass an analyzer oriented along the angle $\theta \pm \pi/2$ with certainty.
2. The singlet state cannot be separated, that is, it cannot be represented by a product of single qubit states, as shown by (1.128). Hence, if two qubits are in the singlet state, neither of them can be in a definite pure state.

The above two properties hold for all three other (triplet) Bell states too and can be straightforwardly generalized to more than two qubits. The state that satisfies the above two conditions is called an *entangled state*.

Two qubits are in an entangled state if the state is inseparable (cannot be expressed as a product of individual qubit states) and if a measurement of an observable carried out on one of the qubits instantly predetermines the outcome of the same observable carried out on the other.

We shall make use the entanglement to obtain *superdense coding* in Section 1.15 and *teleportation* in Section 1.16.

An important property of entangled qubit states which follows from (1.112) is that each of them is undefined by itself.

Independent measurements of the states of each of two entangled qubits give random and unpredictable values.

This means that each photon from a pair of entangled photons is unpolarized. Polarization measurements of one photon with respect to another are correlated, though, as shown by (1.112). We shall make use of these properties for formulating the so-called "ping–pong" protocol of quantum cryptography in Section 1.22.1.

1.14 Separating and Transforming Entanglements: Bell States at a Beam Splitter

To be able to move an entanglement around in a quantum computer, to teleport it from one part of a quantum processor to another and from one computer to another (see Section 1.16), as well as to make use of it in quantum cryptography, we have to learn how to separate and transform entanglements. We shall do that on an example of a two-qubit entangled photon pair at a beam splitter. At the same time, we shall introduce a simplified symmetric second quantization formalism for such a manipulation.

We shall consider photon pairs in the Bell states coming at a beam splitter from its opposite sides. The state of incoming photons will therefore not be the product state given by (1.98), but one of the Bell states given by (1.114) and (1.124), that is, a superposition of two product states (because in, for example, $|\Psi^-\rangle$ the H-photon can come from one side and the V-one from the other but also the other way around).

We start with $|\Psi\rangle = |\Psi^-\rangle$ and form $\hat{E}_1\hat{E}_2|\Psi\rangle$ as in (1.105).

$$\begin{aligned}\hat{E}_1\hat{E}_2|\Psi^-\rangle = &\left[\left(\hat{a}_{1x}t_x\cos\theta_1 + \hat{a}_{1y}t_y\sin\theta_1\right)\eta_{11}\right.\\ &\left.+\, i\left(\hat{a}_{2x}r_x\cos\theta_1 + \hat{a}_{2y}r_y\sin\theta_1\right)\eta_{12}\right]\\ &\times\left[i\left(\hat{a}_{1x}r_x\cos\theta_2 + \hat{a}_{1y}r_y\sin\theta_2\right)\eta_{21}\right.\\ &\left.+\left(\hat{a}_{2x}t_x\cos\theta_2 + \hat{a}_{2y}t_y\sin\theta_2\right)\eta_{22}\right]\\ &\times\frac{1}{\sqrt{2}}(|0\rangle_1|1\rangle_2 - |1\rangle_1|0\rangle_2)\\ = &\sin(\theta_2-\theta_1)\left(t_xt_y\varepsilon_{12} + r_xr_y\tilde{\varepsilon}_{12}\right)\varepsilon|\varnothing\rangle\,, \end{aligned} \tag{1.129}$$

where η's and ε's are given by (1.104) and (1.106), respectively. The procedure is straightforward: we only keep those terms for which the second quantization annihilation operators after acting on kets $|0\rangle_1$, $|0\rangle_2$, $|1\rangle_1$, and $|1\rangle_2$ give $|\varnothing\rangle$ and not 0. For instance, we keep $\hat{a}_{1x}\hat{a}_{2y}|0\rangle_1|1\rangle_2 = |\varnothing\rangle$, but discard terms with $\hat{a}_{1x}\hat{a}_{1y}$ because $\hat{a}_{1x}\hat{a}_{1y}|0\rangle_1|1\rangle_2 = 0$ and $\hat{a}_{1x}\hat{a}_{1y}|1\rangle_1|2\rangle_2 = 0$, and so on. We also have $\hat{a}_{1x}|1\rangle_1 = 0$, $\hat{a}_{1x}|0\rangle_2 = 0$, $\hat{a}_{1x}|1\rangle_2 = 0$, and $\hat{a}_{1x}|0\rangle_1 = |\varnothing\rangle$, and so on.

Similarly, for $|\Psi^+\rangle$ and $|\Phi^\pm\rangle$, we obtain

$$\hat{E}_1\hat{E}_2|\Psi^+\rangle = \frac{\varepsilon}{\sqrt{2}}\left(t_xt_y\varepsilon_{12} - r_xr_y\tilde{\varepsilon_{12}}\right)\varepsilon\sin(\theta_1+\theta_2)|\varnothing\rangle\,, \tag{1.130}$$

$$\begin{aligned}\hat{E}_1\hat{E}_2|\Phi^\mp\rangle = \frac{\varepsilon}{\sqrt{2}}&\left[\left(t_x^2\varepsilon_{12} - r_x^2\varepsilon_{12}\right)\cos\theta_1\cos\theta_2\right.\\ &\left.\mp\left(t_y^2\varepsilon_{12} - r_y^2\varepsilon_{12}\right)\sin\theta_1\sin\theta_2\right]|\varnothing\rangle\,. \end{aligned} \tag{1.131}$$

For a symmetric beam splitter, $t_x = t_y = r_x = t_y = 1/\sqrt{2}$, this yields

$$\begin{aligned}\hat{E}_1\hat{E}_2|\Psi^\mp\rangle &= \frac{\varepsilon(\varepsilon_{12}\pm\tilde{\varepsilon}_{12})}{2\sqrt{2}}\sin(\theta_2\mp\theta_1)|\varnothing\rangle\,,\\ \hat{E}_1\hat{E}_2|\Phi^\mp\rangle &= \frac{\varepsilon(\varepsilon_{12}-\tilde{\varepsilon}_{12})}{2\sqrt{2}}\cos(\theta_2\pm\theta_1)|\varnothing\rangle \end{aligned} \tag{1.132}$$

and the probability of detecting photons coming out from the opposite sides of the beam splitter, that is, being simultaneously detected by detectors, is for Ψ^-

$$P_{\Psi^-}(\theta_1,\theta_2) = \langle\Psi^-|\hat{E}_2^\dagger\hat{E}_1^\dagger\hat{E}_1\hat{E}_2|\Psi^-\rangle = \langle\hat{E}_2^\dagger\hat{E}_1^\dagger\hat{E}_1\hat{E}_2\rangle$$
$$= \frac{1}{8}(e^{-i\alpha}+e^{-i\tilde{\alpha}})(e^{i\alpha}+e^{i\tilde{\alpha}})\sin^2(\theta_2-\theta_1) = \frac{1+\cos\phi}{4}\sin^2(\theta_2-\theta_1) \tag{1.133}$$

where $\alpha = -i\arg[\varepsilon_{12}] = \boldsymbol{k}_1\cdot\boldsymbol{r}_1+\boldsymbol{k}_2\cdot\boldsymbol{r}_2$, $\tilde{\alpha} = -i\arg[\tilde{\varepsilon}_{12}] = \tilde{\boldsymbol{k}}_1\cdot\boldsymbol{r}_2+\tilde{\boldsymbol{k}}_2\cdot\boldsymbol{r}_1$, and ϕ is given by (1.109).

The probabilities for Ψ^+ and $\Phi^\mp$ are

$$P_{\Psi^+}(\theta_1,\theta_2) = \frac{1-\cos\phi}{4}\sin^2(\theta_2+\theta_1)\,,$$
$$P_{\Phi^\mp}(\theta_1,\theta_2) = \frac{1-\cos\phi}{4}\cos^2(\theta_2\pm\theta_1)\,. \tag{1.134}$$

For a symmetric positioning of detectors (see Figure 1.29), we have $\phi = 0$ and for this case – which is going to be of primary concern in the next two sections – (1.133) and (1.134) yield

$$P_{\Psi^-}(\theta_1,\theta_2) = \frac{1}{2}\sin^2(\theta_2-\theta_1)\,,\quad P_{\Psi^+}(\theta_1,\theta_2) = 0\,,$$
$$P_{\Phi^\mp}(\theta_1,\theta_2) = 0\,. \tag{1.135}$$

For mutually perpendicularly oriented detectors (e.g., D and $\mathrm{D}^\perp$ in Figure 1.29), we obtain

$$P_{\Psi^-}\left(\theta_1,\theta_2^\perp\right) = \frac{1}{2}\cos^2(\theta_2-\theta_1)\,,\quad P_{\Psi^+}\left(\theta_1,\theta_2^\perp\right) = 0\,,$$
$$P_{\Phi^\mp}\left(\theta_1,\theta_2^\perp\right) = 0\,. \tag{1.136}$$

That means that only photons in state $|\Psi^-\rangle$ split at a beam splitter (emerge from its opposite sides; *anti-bunching* effect) with probability $P_{\Psi^-}(\theta_1,\theta_2) + P_{\Psi^-}(\theta_1,\theta_2^\perp) = 1$, while photons in states $|\Psi^+\rangle$ and $|\Phi^\mp\rangle$ do not split at all.

To see which states the split photons are in, $|\Psi^-\rangle$ or $|\Psi^+\rangle$, we proceed as in (1.113) and act with $\hat{E}_{10}^\dagger\hat{E}_{20}^\dagger$ to the vacuum state $|\varnothing\rangle$. In doing so, we recover the photon states that we can measure after their passing through the beam splitter, that is, we go back to the terms that we obtain after multiplication of the expressions in two squared brackets from (1.129); those terms that correspond to the photons which passed through the beam splitter will give sign + to the created $|01\rangle$ part and those that correspond to the photons which were reflected from the beam splitter will give sign $i\cdot i = -1$ to the $|10\rangle$ part. Therefore, the photons that emerge from opposite sides of the beam splitter are in state $|\Psi^-\rangle$ as we also found for unpolarized photons in Section 1.13 (see (1.114)).

Let us now see in which states the photons are in when they emerge from the same sides of the beam splitter, that is, when they *bunch* (*stick*) together. To calculate

their probabilities of being detected by photon number resolving detectors after passing through a polarizing beam splitter (PBS), we shall use (1.115) and (1.117).

$$\begin{aligned}\hat{E}_2'\hat{E}_2|\Psi^{\mp}\rangle &= \big[\big(\hat{a}_{2x}t_x\cos\theta_2+\hat{a}_{2y}t_y\sin\theta_2\big)\,\eta_{22}\\ &\quad+i\,\big(\hat{a}_{1x}r_x\cos\theta_2+\hat{a}_{1y}r_y\sin\theta_2\big)\,\eta_{21}\big]\\ &\quad\times\big[i\,\big(\hat{a}_{1x}r_x\cos\theta_1+\hat{a}_{1y}r_y\sin\theta_1\big)\,\eta_{21}'\\ &\quad+\big(\hat{a}_{2x}t_x\cos\theta_1+\hat{a}_{2y}t_y\sin\theta_1\big)\,\eta_{22}'\big]\\ &\quad\times\frac{1}{\sqrt{2}}(|0\rangle_1|1\rangle_2\mp|1\rangle_1|0\rangle_2)\\ &=\frac{i\varepsilon}{\sqrt{2}}\big[t_yr_x(\varepsilon_{22}+\varepsilon_{22}')\mp t_xr_y(\varepsilon_{22}+\varepsilon_{22}')\big]\sin(\theta_2\mp\theta_1)|\varnothing\rangle\,,\end{aligned}\tag{1.137}$$

where η_{21} and η_{22} are given by (1.104), η_{21}' and η_{22}' by (1.116), and ε_{22} and ε_{22}' by (1.118), that is,

$$\begin{aligned}\hat{E}_2'\hat{E}_2|\Phi^{\mp}\rangle &= \big[\big(\hat{a}_{2x}t_x\cos\theta_2+\hat{a}_{2y}t_y\sin\theta_2\big)\,\eta_{22}\\ &\quad+i\,\big(\hat{a}_{1x}r_x\cos\theta_2+\hat{a}_{1y}r_y\sin\theta_2\big)\,\eta_{21}\big]\\ &\quad\times\big[i\,\big(\hat{a}_{1x}r_x\cos\theta_1+\hat{a}_{1y}r_y\sin\theta_1\big)\,\eta_{21}'\\ &\quad+\big(\hat{a}_{2x}t_x\cos\theta_1+\hat{a}_{2y}t_y\sin\theta_1\big)\,\eta_{22}'\big]\\ &\quad\times\frac{1}{\sqrt{2}}(|0\rangle_1|0\rangle_2\mp|1\rangle_1|1\rangle_2)\\ &=\frac{i\varepsilon}{\sqrt{2}}\big[t_xr_x(\varepsilon_{22}+\varepsilon_{22}')\cos\theta_1\cos\theta_2\\ &\quad\mp t_yr_y(\varepsilon_{22}+\varepsilon_{22}')\sin\theta_1\sin\theta_2\big]\,|\varnothing\rangle\,.\end{aligned}\tag{1.138}$$

For a symmetric beam splitter, $t_x=t_y=r_x=t_y=1/\sqrt{2}$, we obtain

$$\begin{aligned}&\hat{E}_2\hat{E}_2'|\Psi^-\rangle=0\,,\quad \hat{E}_2\hat{E}_2'|\Psi^+\rangle=\frac{i\varepsilon(\varepsilon_{22}+\varepsilon_{12}')}{2\sqrt{2}}\sin(\theta_2+\theta_1)|\varnothing\rangle\,,\\ &\hat{E}_2\hat{E}_2'|\Phi^{\mp}\rangle=\frac{i\varepsilon(\varepsilon_{22}+\varepsilon_{12}')}{2\sqrt{2}}\cos(\theta_2\pm\theta_1)|\varnothing\rangle\end{aligned}\tag{1.139}$$

and the probability of detecting photons coming out from the same side of the beam splitter, that is, being bunched together, for a symmetric positioning of detectors (see Figure 1.29), we have $\phi=0$. For $|\Phi^{\mp}\rangle$, it is just a photon number detector that can tell two photons from one.

$$\begin{aligned}&P_{\Psi^-}(\theta_1,\theta_2)=0\,,\quad P_{\Psi^+}(\theta_1,\theta_2)=\frac{1}{2}\sin^2(\theta_1+\theta_2)\,,\\ &P_{\Phi^{\mp}}(\theta_1,\theta_2)=\frac{1}{2}\cos^2(\theta_1\pm\theta_2)\,.\end{aligned}\tag{1.140}$$

To determine the states the obtained probabilities correspond to, we can proceed "backwards" as we did in Section 1.13 to obtain the Bell state $|\Psi^-\rangle$ in (1.114). Here

we get $|\Psi^+\rangle$ and $|\Phi^\mp\rangle$ bunched together as opposed to, for example, those given by (1.191)–(1.193) in Section 1.15. However, the "backwards" procedure is rather complicated and therefore we introduce a simplified procedure to get two-qubit state transformations at a beam splitter below.

To this aim we consider electrical fields that pass through a beam splitter as given by (1.103). In order to describe their behavior with respect to unitarity and energy conservation we start without polarization. The annihilation operator part of (1.103) can then be written in the following simplified matrix form (cf. (1.100)):

$$\hat{a}_{\text{out}} = \begin{pmatrix} \hat{a}_{1-\text{out}} \\ \hat{a}_{2-\text{out}} \end{pmatrix} = \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix} \begin{pmatrix} \hat{a}_{1-\text{in}} \\ \hat{a}_{2-\text{in}} \end{pmatrix} = \boldsymbol{s}\hat{a}_{\text{in}} \,, \tag{1.141}$$

where s_{12}, s_{21} are the complex transmittances and s_{11}, s_{22} the complex reflectances with the implicit meaning given by the indices (e.g., s_{11} means a reflectance at side 1 and s_{12} a transmittance from side 1 to side 2).

To specify s_{ij}, $i, j = 1, 2$ further we consider Ou–Mandel setup [219] – shown in Figure 1.33 – and closely follow their derivation of $\boldsymbol{s}$-matrix relations for a beam splitter from energy balance.

We send a monochromatic plane wave from point A_1 to a beam splitter (BS) at an arbitrary angle. It suffers a phase shift ϕ_1 from A_1 to B_1. Then, it is partly reflected from BS and partly transmitted through it. The reflected part suffers ϕ_3 from B_1 to M_1 and back; then, it is partly reflected from BS and suffers ϕ_1 from B_1 to A_1 and partly transmitted through BS and then it suffers ϕ_2 from B_2 to A_2. The transmitted part suffers ϕ_4 from B_2 to M_2 and back; then, it is partly reflected from BS and suffers ϕ_2 from B_2 to A_2 and partly transmitted through BS and then it suffers ϕ_1 from B_1 to A_1.

We describe the amplitudes of the waves that exit at A_1 and A_2 by

$$A_1 = e^{i\phi_1} s_{11} e^{i\phi_3} s_{11} e^{i\phi_1} + e^{i\phi_1} s_{12} e^{i\phi_4} s_{21} e^{i\phi_1} \tag{1.142}$$

and

$$A_2 = e^{i\phi_1} s_{11} e^{i\phi_3} s_{12} e^{i\phi_2} + e^{i\phi_1} s_{12} e^{i\phi_4} s_{22} e^{i\phi_2} \,, \tag{1.143}$$

respectively.

We now introduce the following way of writing $\boldsymbol{s}$-matrix elements.

$$s_{11} = r_{11} e^{i\sigma_{11}} \,, \quad s_{22} = r_{22} e^{i\sigma_{22}} \,, \quad s_{12} = t_{12} e^{i\sigma_{12}} \,, \quad s_{21} = t_{21} e^{i\sigma_{21}} \,. \tag{1.144}$$

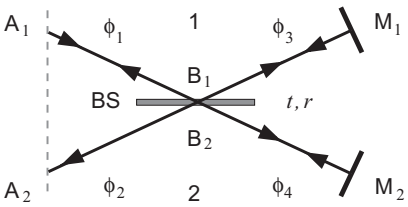


Figure 1.33 Michelson interferometer used for deriving beam splitter properties. ϕ_j, $j = 1, 2, 3, 4$ are phase shifts, M_k, $k = 1, 2$

The energy conservation requires that the sum of outgoing intensities (see (1.142) and (1.143)) be equal to the incoming intensity, that is, to one.

$$\begin{aligned} 1 = A_1^* A_1 + A_2^* A_2 = r_{11}^2 \left(r_{11}^2 + t_{12}^2\right) + 2 r_{11}^2 t_{12} t_{21} \\ \times \cos(\sigma_{12} + \sigma_{21} - 2\sigma_{11} + \phi_4 - \phi_3) + 2 t_{12}^2 r_{11} r_{22} \\ \times \cos(\sigma_{22} - \sigma_{11} + \phi_4 - \phi_3) + t_{12}^2 \left(t_{21}^2 + r_{22}^2\right). \end{aligned} \tag{1.145}$$

The terms containing cosine functions must vanish because they contain arbitrary phases ϕ and this is only possible if [219]

$$\sigma_{12} + \sigma_{21} - \sigma_{11} - \sigma_{22} = \pm\pi \tag{1.146}$$

and

$$r_{11} t_{21} = r_{22} t_{12} \,. \tag{1.147}$$

Then, the remaining terms in (1.145) yield

$$r_{11} = r_{22} = r \,, \quad t_{21} = t_{12} = t \,, \quad \text{and} \quad r^2 + t^2 = 1 \,. \tag{1.148}$$

From (1.144), (1.147), and (1.148), we get

$$r t e^{i\sigma_{12} - i\sigma_{11}} + r t e^{i\sigma_{21} - i\sigma_{22}} = 0 \Longrightarrow s_{11}^* s_{21} + s_{12}^* s_{22} = 0 \,. \tag{1.149}$$

That means that $\boldsymbol{s}$ matrix (1.141) is a unitary matrix, as follows from (1.54), (1.148), and (1.149):

$$\begin{aligned} \boldsymbol{s}\boldsymbol{s}^\dagger &= \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix} \begin{pmatrix} s_{11}^* & s_{21}^* \\ s_{12}^* & s_{22}^* \end{pmatrix} = \begin{pmatrix} s_{11} s_{11}^* + s_{12} s_{12}^* & s_{11} s_{21}^* + s_{12} s_{22}^* \\ s_{11}^* s_{21} + s_{12}^* s_{22} & s_{21} s_{21}^* + s_{22} s_{22}^* \end{pmatrix} \\ &= \begin{pmatrix} r^2 + t^2 & 0 \\ 0 & t^2 + r^2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = I. \end{aligned} \tag{1.150}$$

The unitarity of $\boldsymbol{s}$ *matrix* (1.141) *for a beam splitter follows from the input–output photon energy conservation at a beam splitter.*

The $\boldsymbol{s}$ matrix is a kind of scattering matrix and according to Wigner's theorem, any scattering matrix must be either unitary or antiunitary.

We choose a particular form of the $\boldsymbol{s}$ matrix so as to suit an application we want to use it for and so as to satisfy the conditions (1.147)–(1.149). For example, to get the electric field operators in (1.103), we use $\sigma_{11} = \pi/2$, $\sigma_{12} = \sigma_{21} = 0$. From (1.146) we then get $\sigma_{22} = -3\pi/2$ and $\sigma_{22} = \pi/2$, both of which give $e^{i\sigma_{22}} = i$ and yield $\boldsymbol{s}_{i,i}$ in (1.151). Another possibility, which is also frequently employed in the literature, is to take $\sigma_{11} = \sigma_{12} = \sigma_{21} = 0$. From (1.146), we get $\sigma_{22} = \pm\pi$ and $e^{\sigma_{22}} = -1$, so as to obtain $\boldsymbol{s}_{1,-1}$.

$$\boldsymbol{s}_{i,i} = \begin{pmatrix} ir & t \\ t & ir \end{pmatrix} \;; \quad \boldsymbol{s}_{1,-1} = \begin{pmatrix} r & t \\ t & -r \end{pmatrix} \,. \tag{1.151}$$

Two other possibilities will be of importance below. One is $\sigma_{11} = -\pi/2$, $\sigma_{12} = \sigma_{21} = 0$. From (1.146), we then get $\sigma_{22} = 3\pi/2$ and $\sigma_{22} = -\pi/2$, both of which give $e^{\sigma_{22}} = -i$, so as to obtain $\boldsymbol{s}_{-i,-i}$. The other is to take $\sigma_{12} = \sigma_{21} = \sigma_{22} = 0$. From (1.146), we get $\sigma_{11} = \pm\pi$ and $e^{\sigma_{11}} = \boldsymbol{s}_{-1,1}$,

$$\boldsymbol{s}_{-i,-i} = \begin{pmatrix} -ir & t \\ t & -ir \end{pmatrix} ; \quad \boldsymbol{s}_{-1,1} = \begin{pmatrix} -r & t \\ t & r \end{pmatrix} . \tag{1.152}$$

The following procedure of finding how photons combine and how photon states transform at a beam splitter is by and large limited to symmetric beam splitters.

Property 46

When we make setups and experiments with polarization, then we usually have the following two options

1. polarization-preserving beam splitter; for example, a *metallic* beam splitter;
2. polarization-non-preserving beam splitter, that is, a beam splitter at which vertical polarization (perpendicular to the incident plane) suffers an extra π phase shift; for example, a *dielectric* beam splitter.

For the first kind of beam splitters, (1.141) gives the following equations for annihilation operators for symmetric beam splitters ($t_x = t_y = r_x = r_y = 1/\sqrt{2}$) when we choose $\boldsymbol{s}_{i,i}$ from (1.151) (see Figure 1.34).

$$\begin{aligned} \hat{a}_{1x-\text{out}} &= \frac{1}{\sqrt{2}}\left(\hat{a}_{1x-\text{in}} + i\hat{a}_{2x-\text{in}}\right) , \quad \hat{a}_{1y-\text{out}} = \frac{1}{\sqrt{2}}\left(\hat{a}_{1y-\text{in}} + i\hat{a}_{2y-\text{in}}\right) , \\ \hat{a}_{2x-\text{out}} &= \frac{1}{\sqrt{2}}\left(\hat{a}_{2x-\text{in}} + i\hat{a}_{1x-\text{in}}\right) , \quad \hat{a}_{2y-\text{out}} = \frac{1}{\sqrt{2}}\left(\hat{a}_{2y-\text{in}} + i\hat{a}_{1y-\text{in}}\right) . \end{aligned} \tag{1.153}$$

Equivalently, we can take any $\boldsymbol{s}$ matrix from (1.151) and (1.152).

The input state can be a product of states, superposition of states, or their entanglement – see (1.129), (1.131), (1.137), (1.138), and so on. This selects a particular

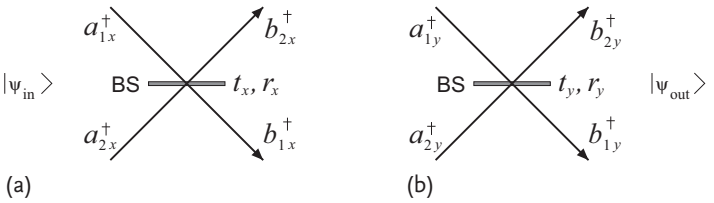


Figure 1.34 Creation operator approach to a beam splitter; (i) horizontal polarization; (ii) vertical polarization.

combination of $\hat{a}_{\text{out}}$ operators. Thus, we can look at the latter combinations as at input-determined creation operators which will give a final output state when arranged according to the input state when applied to the Fock vacuum state $|\varnothing\rangle$. For the $\boldsymbol{s}_{1,-1}$ choice, we might write

$$\hat{b}_{1x}^{\dagger} = \frac{1}{\sqrt{2}}\left(\hat{a}_{1x}^{\dagger} + \hat{a}_{2x}^{\dagger}\right), \qquad \hat{b}_{1y}^{\dagger} = \frac{1}{\sqrt{2}}\left(\hat{a}_{1y}^{\dagger} + \hat{a}_{2y}^{\dagger}\right),$$
$$\hat{b}_{2x}^{\dagger} = \frac{1}{\sqrt{2}}\left(\hat{a}_{2x}^{\dagger} - \hat{a}_{1x}^{\dagger}\right), \qquad \hat{b}_{2y}^{\dagger} = \frac{1}{\sqrt{2}}\left(\hat{a}_{2y}^{\dagger} - \hat{a}_{1y}^{\dagger}\right). \tag{1.154}$$

To obtain the outcoming state of the photons incoming to a beam splitter in the Bell states $|\Psi^{\mp}\rangle$, $|\Phi^{\mp}\rangle$, we follow the following procedure. We first switch from the state vector representation to the annihilation operator representation, that is,

$$|\Psi_{\text{in}}^{\mp}\rangle = \frac{1}{\sqrt{2}}(|01\rangle \mp |10\rangle) = \frac{1}{\sqrt{2}}\left(a_{1x}^{\dagger}a_{2y}^{\dagger} \mp a_{1y}^{\dagger}a_{2x}^{\dagger}\right)|\varnothing\rangle,$$
$$|\Phi_{\text{in}}^{\mp}\rangle = \frac{1}{\sqrt{2}}(|00\rangle \mp |11\rangle) = \frac{1}{\sqrt{2}}\left(a_{1x}^{\dagger}a_{2x}^{\dagger} \mp a_{1y}^{\dagger}a_{2y}^{\dagger}\right)|\varnothing\rangle, \tag{1.155}$$

where the order of creation operators is made with respect to the sides of the beam splitter – the indices "1" and "2" denote the sides of the beam splitter from which the photons come; they do not refer to the first or the second photon.

Then, we find transformations that are inverse to those given by (1.154). To do so, we first write them down in a matrix form:

$$\begin{pmatrix}\hat{b}_{1x}^{\dagger}\\ \hat{b}_{2x}^{\dagger}\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ -1 & 1\end{pmatrix}\begin{pmatrix}\hat{a}_{1x}^{\dagger}\\ \hat{a}_{2x}^{\dagger}\end{pmatrix} = \begin{pmatrix}\hat{a}_{1x}^{\dagger} + \hat{a}_{2x}^{\dagger}\\ \hat{a}_{2x}^{\dagger} - \hat{a}_{1x}^{\dagger}\end{pmatrix}, \tag{1.156}$$
$$\begin{pmatrix}\hat{b}_{1y}^{\dagger}\\ \hat{b}_{2y}^{\dagger}\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ -1 & 1\end{pmatrix}\begin{pmatrix}\hat{a}_{1y}^{\dagger}\\ \hat{a}_{2y}^{\dagger}\end{pmatrix} = \begin{pmatrix}\hat{a}_{1y}^{\dagger} + \hat{a}_{2y}^{\dagger}\\ \hat{a}_{2y}^{\dagger} - \hat{a}_{1y}^{\dagger}\end{pmatrix}. \tag{1.157}$$

According to Definition 27, the operator inverse to the one in (1.156) and (1.156) is its conjugate transpose and since it is real, only transpose. Thus, the inverse transformations in the matrix form read

$$\begin{pmatrix}\hat{a}_{1x}^{\dagger}\\ \hat{a}_{2x}^{\dagger}\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & -1\\ 1 & 1\end{pmatrix}\begin{pmatrix}\hat{b}_{1x}^{\dagger}\\ \hat{b}_{2x}^{\dagger}\end{pmatrix} = \begin{pmatrix}\hat{b}_{1x}^{\dagger} - \hat{b}_{2x}^{\dagger}\\ \hat{b}_{1x}^{\dagger} + \hat{b}_{2x}^{\dagger}\end{pmatrix}, \tag{1.158}$$
$$\begin{pmatrix}\hat{a}_{1y}^{\dagger}\\ \hat{a}_{2y}^{\dagger}\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & -1\\ 1 & 1\end{pmatrix}\begin{pmatrix}\hat{b}_{1y}^{\dagger}\\ \hat{b}_{2y}^{\dagger}\end{pmatrix} = \begin{pmatrix}\hat{b}_{1y}^{\dagger} - \hat{b}_{2y}^{\dagger}\\ \hat{b}_{1y}^{\dagger} + \hat{b}_{2y}^{\dagger}\end{pmatrix} \tag{1.159}$$

and in the standard annihilation operator form:

$$\hat{a}_{1x}^{\dagger} = \frac{1}{\sqrt{2}}\left(\hat{b}_{1x}^{\dagger} - \hat{b}_{2x}^{\dagger}\right), \quad \hat{a}_{1y}^{\dagger} = \frac{1}{\sqrt{2}}\left(\hat{b}_{1y}^{\dagger} - \hat{b}_{2y}^{\dagger}\right),$$
$$\hat{a}_{2x}^{\dagger} = \frac{1}{\sqrt{2}}\left(\hat{b}_{1x}^{\dagger} + \hat{b}_{2x}^{\dagger}\right), \quad \hat{a}_{2y}^{\dagger} = \frac{1}{\sqrt{2}}\left(\hat{b}_{1y}^{\dagger} + \hat{b}_{2y}^{\dagger}\right). \tag{1.160}$$

By introducing a's from (1.160) into (1.155), we obtain

$$
\begin{aligned}
|\Psi^-\rangle_{12\text{in}} &\to \frac{1}{\sqrt{2}}\left(\hat{b}^\dagger_{1x}\hat{b}^\dagger_{2y} - \hat{b}^\dagger_{1y}\hat{b}^\dagger_{2x}\right)|\varnothing\rangle = |\Psi^-\rangle_{12\text{out}}\,, \\
|\Psi^+\rangle_{12\text{in}} &\to \frac{1}{\sqrt{2}}\left(\hat{b}^\dagger_{1x}\hat{b}^\dagger_{1y} - \hat{b}^\dagger_{2y}\hat{b}^\dagger_{2x}\right)|\varnothing\rangle = \frac{1}{\sqrt{2}}\left(|01\rangle_{11\text{out}} - |10\rangle_{22\text{out}}\right)\,, \\
|\Phi^\mp\rangle_{12\text{in}} &\to \frac{1}{2\sqrt{2}}\left[\hat{b}^\dagger_{1x}\hat{b}^\dagger_{1x} \mp \hat{b}^\dagger_{1y}\hat{b}^\dagger_{1y} - \left(\hat{b}^\dagger_{2x}\hat{b}^\dagger_{2x} \mp \hat{b}^\dagger_{2y}\hat{b}^\dagger_{2y}\right)\right]|\varnothing\rangle \\
&= \frac{1}{2}\left(|\Phi^\mp\rangle_{11\text{out}} - |\Phi^\mp\rangle_{22\text{out}}\right)\,.
\end{aligned}
\tag{1.161}
$$

There is yet another link between the state in which photons are in and the probability that we will detect the state when the analyzers are rotated by an arbitrary angle. The state $|\Psi^-\rangle$ has the probability $P_{\Psi^-}(\theta_1, \theta_2)$ of being detected by detectors behind analyzers oriented along angles θ_1 and θ_2 equal to $1/2\sin^2(\theta_2 - \theta_1)$ (see (1.135)). Both, the probability and the state are obviously invariant under rotation by angle $\pi/2$.

The state $|\Psi^+\rangle$ has the probability of being detected equal to $1/2\sin^2(\theta_2 + \theta_1)$ (see (1.140)). Here, the invariance under rotation is less obvious. We see that the probability is invariant under a change of θ_1 by an arbitrary angle θ_0 and θ_2 by $-\theta_0$. Let us verify that the state is invariant under this rotation as well.

When we rotate the bases of polarization by θ_0 and $-\theta_0$, we obtain

$$
\begin{aligned}
|0\rangle'_1 &= \cos\theta_0|0\rangle_1 + \sin\theta_0|1\rangle_1\,, & |0\rangle'_2 &= \cos\theta_0|0\rangle_2 - \sin\theta_0|1\rangle_2\,, \\
|1\rangle'_1 &= -\sin\theta_0|0\rangle_1 + \cos\theta_0|1\rangle_1\,, & |1\rangle'_2 &= \sin\theta_0|0\rangle_2 + \cos\theta_0|1\rangle_2\,.
\end{aligned}
\tag{1.162}
$$

To get the original basis $|0\rangle$–$|1\rangle$ expressed in the $|0\rangle'$–$|1\rangle'$ basis, we solve (1.162), that is,

$$
\begin{aligned}
|0\rangle_1 &= \cos\theta_0|0\rangle'_1 - \sin\theta_0|1\rangle'_1\,, & |0\rangle_2 &= \cos\theta_0|0\rangle'_2 + \sin\theta_0|1\rangle'_2\,, \\
|1\rangle_1 &= \sin\theta_0|0\rangle'_1 + \cos\theta_0|1\rangle'_1\,, & |1\rangle_2 &= -\sin\theta_0|0\rangle'_2 + \cos\theta_0|1\rangle'_2\,.
\end{aligned}
\tag{1.163}
$$

By introducing the obtained $|0\rangle, |1\rangle$ into

$$
|\Psi^+\rangle = \frac{1}{\sqrt{2}}(|0\rangle_1|1\rangle_2 + |1\rangle_1|0\rangle_2)\,, \tag{1.164}
$$

we obtain

$$
|\Psi^+\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle'_1|1\rangle'_2 + |1\rangle'_1|0\rangle'_2\right)\,, \tag{1.165}
$$

proving that the state $|\Psi^+\rangle$ really is invariant under the aforementioned rotation.

Now, we might recall that $\boldsymbol{s}$ matrix is unitary and ask ourselves whether the bunching of photons in the three Bell states can be reversed. The answer is partly in the negative. When we send the bunched photons to another beam splitter, only

half of them will antibunch (split) and the other half will emerge from it bunched again. For instance, for $|00\rangle$, we obtain from (1.160)

$$\hat{a}^{\dagger}_{2x}\hat{a}^{\dagger}_{2x} = \hat{b}^{\dagger}_{1x}\hat{b}^{\dagger}_{2x} + \frac{1}{2}\left(\hat{b}^{\dagger}_{1x}\hat{b}^{\dagger}_{1x} + \hat{b}^{\dagger}_{2x}\hat{b}^{\dagger}_{2x}\right) . \tag{1.166}$$

The same argument holds for unpolarized photons or any Φ Bell state (bunched photons cannot be in a Ψ Bell state, of course; instead, they are in $|01\rangle$ state). For instance, for $|\Phi^{-}\rangle$, we get

$$|\Phi^{-}_{\text{in}}\rangle_{11} \rightarrow \text{either} \quad |\Phi^{-}_{\text{out}}\rangle_{11} \quad \text{or} \quad |\Phi^{-}_{\text{out}}\rangle_{12} \quad \text{or} \quad |\Phi^{-}_{\text{out}}\rangle_{21} \quad \text{or} \quad |\Phi^{-}_{\text{out}}\rangle_{22} . \tag{1.167}$$

Two identical photons that are incident at the same side of a beam splitter behave completely classically, as we can see from (1.166). More specifically, the statistics shown in Figure 1.35 follow from the classical Bernoulli distribution [61, IV.D]. This has also been verified experimentally [166].

At the second kind of beam splitters – polarization-nonpreserving ones – vertical polarization suffers a π phase shift. And, when we look at how we obtain $\boldsymbol{s}$ matrices in (1.151) and (1.152), we see that a difference between $\boldsymbol{s}_{i,i}$ and $\boldsymbol{s}_{-i,-i}$ is in $\sigma_{22} = -3\pi/2$ and $\sigma_{22} = \pi/2$ for the former one vs. $\sigma_{22} = 3\pi/2$ and $\sigma_{22} = -\pi/2$ for the latter. A π phase shift turns the former matrix into the latter one and vice versa. Similarly, for $\boldsymbol{s}_{1,-1}$ and $\boldsymbol{s}_{-1,1}$, we have $\sigma_{11} = 0$ and $\sigma_{22} = \pm\pi$ for the former matrix and $\sigma_{11} = \pm\pi$ and $\sigma_{22} = 0$ for the latter, and action of the shift is obvious.

Thus, when we utilize a dielectric beam splitter of the second kind we must apply *mixed transformations*. For instance, for $\boldsymbol{s}_{1,-1}$ and $\boldsymbol{s}_{-1,1}$, they read

$$\begin{aligned} \hat{a}^{\dagger}_{1x} &= \frac{1}{\sqrt{2}}\left(\hat{b}^{\dagger}_{1x} + \hat{b}^{\dagger}_{2x}\right) , \qquad \hat{a}^{\dagger}_{1y} = \frac{1}{\sqrt{2}}\left(\hat{b}^{\dagger}_{1y} - \hat{b}^{\dagger}_{2y}\right) , \\ \hat{a}^{\dagger}_{2x} &= \frac{1}{\sqrt{2}}\left(\hat{b}^{\dagger}_{2x} - \hat{b}^{\dagger}_{1x}\right) , \qquad \hat{a}^{\dagger}_{2y} = \frac{1}{\sqrt{2}}\left(\hat{b}^{\dagger}_{2y} + \hat{b}^{\dagger}_{1y}\right) . \end{aligned} \tag{1.168}$$

With these transformations, that is, at a dielectric beam splitter, the Bell states do not transform in the same way as at a metallic beam splitter of the first kind (see Properties 46). In particular, the input Bell states $|\Psi^{\mp}\rangle_{\text{in}}$, $|\Phi^{\mp}\rangle_{\text{in}}$ do not emerge from the other sides of a dielectric beam splitter as output Bell states $|\Psi^{\mp}\rangle_{\text{out}}$, $|\Phi^{\mp}\rangle_{\text{out}}$ as opposed to the results we obtained for a metallic beam splitter in (1.161).

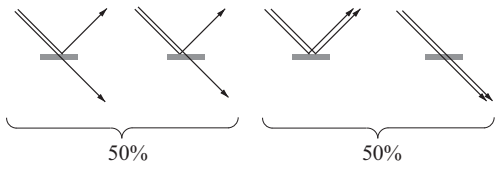


Figure 1.35 Beam splitter classical output photon distribution for photons incident at the same side of a beam splitter of any kind: parallelly polarized, perpendicularly, polarized, or entangled in the Bell states. Compare with Figure 1.31.

The transformations have been theoretically elaborated in [222, Section 4.1] and [215], and referred to in an excellent review on generation, observation, and characterization of entangled photons by Keiichi Edamatsu [89]. It was used for experiments in [218, 280]. The majority of experiments and proposals so far have been carried out by means of metallic beam splitters – so, the details of dielectric ones are not widely known and therefore we shall elaborate on some of them in our approach below.[22)]

We obtain creation operators for a polarization-nonpreserving (dielectric) beam splitter as follows. Vertical polarization suffers a phase shift of π from the Fresnel coefficients; see [222, p. 64, Eqs. (4.2,3)].

$$\hat{b}^\dagger_{1x} = \frac{1}{\sqrt{2}}\left(\hat{a}^\dagger_{1x} + \hat{a}^\dagger_{2x}\right) , \qquad \hat{b}^\dagger_{1y} = \frac{1}{\sqrt{2}}\left(\hat{a}^\dagger_{1y} - \hat{a}^\dagger_{2y}\right) ,$$
$$\hat{b}^\dagger_{2x} = \frac{1}{\sqrt{2}}\left(\hat{a}^\dagger_{2x} - \hat{a}^\dagger_{1x}\right) , \qquad \hat{b}^\dagger_{2y} = \frac{1}{\sqrt{2}}\left(\hat{a}^\dagger_{1y} + \hat{a}^\dagger_{2y}\right) , \tag{1.169}$$

Equation (1.169) can be written as follows:

$$\begin{pmatrix} \hat{b}^\dagger_{1x} \\ \hat{b}^\dagger_{2x} \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} \hat{a}^\dagger_{1x} \\ \hat{a}^\dagger_{2x} \end{pmatrix} = \begin{pmatrix} \hat{a}^\dagger_{1x} + \hat{a}^\dagger_{2x} \\ \hat{a}^\dagger_{2x} - \hat{a}^\dagger_{1x} \end{pmatrix} , \tag{1.170}$$

$$\begin{pmatrix} \hat{b}^\dagger_{1y} \\ \hat{b}^\dagger_{2y} \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \hat{a}^\dagger_{1y} \\ \hat{a}^\dagger_{2y} \end{pmatrix} = \begin{pmatrix} \hat{a}^\dagger_{1y} - \hat{a}^\dagger_{2y} \\ \hat{a}^\dagger_{1y} + \hat{a}^\dagger_{2y} \end{pmatrix} . \tag{1.171}$$

Their (unitary) inverse transformations (cf. [2, p. 141, (4)]) are

$$\begin{pmatrix} \hat{a}^\dagger_{1x} \\ \hat{a}^\dagger_{2x} \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \hat{b}^\dagger_{1x} \\ \hat{b}^\dagger_{2x} \end{pmatrix} = \begin{pmatrix} \hat{b}^\dagger_{1x} - \hat{b}^\dagger_{2x} \\ \hat{b}^\dagger_{1x} + \hat{b}^\dagger_{2x} \end{pmatrix} , \tag{1.172}$$

$$\begin{pmatrix} \hat{a}^\dagger_{1y} \\ \hat{a}^\dagger_{2y} \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} \hat{b}^\dagger_{1y} \\ \hat{b}^\dagger_{2y} \end{pmatrix} = \begin{pmatrix} \hat{b}^\dagger_{1y} + \hat{b}^\dagger_{2y} \\ \hat{b}^\dagger_{2y} - \hat{b}^\dagger_{1y} \end{pmatrix} . \tag{1.173}$$

22) Apart from understanding the experiments referred to in the literature, a presentation of both kinds of beam splitters can help us in resolving some interpretational problems. For instance, the following boson interpretation for obtaining the transformations given by (1.161) at a metallic beam splitter has been put forward [158], [222, Section 4.1.2]. It states that the total wave function (a product of the polarization and spatial ones) should be symmetric because that is required for bosons. The antisymmetric wave function $|\Psi^-\rangle_{12}$ which emerges from a metallic beam splitter describes a pair of photons that emerge from the opposite sides of the beam splitter and therefore have an antisymmetric spatial wave function. The total wave function is symmetric. The three symmetric wave functions that emerge from a symmetric beam splitter $|\Psi^+\rangle_{jj}$, $|\Phi^\mp\rangle_{jj}$, $j = 1, 2$ describe pairs of photons that emerge from the same sides of the beam splitter and therefore have symmetric spatial wave functions. Again, the total wave functions are symmetric. The problem with this interpretation is that it does not work with dielectric beam splitters where we obtain a product of antisymmetrical spatial wave function and a symmetric spin (polarization) function (and vice versa) what makes the total wave function antisymmetric and that is not "required for bosons." Hence, it seems that this boson reasoning cannot be universally applied to Bell state transformations [222, Section 4.1.2].

So, for inverse transformation at a polarization-non-preserving beam splitter, we use

$$\begin{aligned}
\hat{a}_{1x}^{\dagger} &= \frac{1}{\sqrt{2}}\left(\hat{b}_{1x}^{\dagger} - \hat{b}_{2x}^{\dagger}\right), \qquad \hat{a}_{1y}^{\dagger} = \frac{1}{\sqrt{2}}\left(\hat{b}_{1y}^{\dagger} + \hat{b}_{2y}^{\dagger}\right), \\
\hat{a}_{2x}^{\dagger} &= \frac{1}{\sqrt{2}}\left(\hat{b}_{1x}^{\dagger} + \hat{b}_{2x}^{\dagger}\right), \qquad \hat{a}_{2y}^{\dagger} = \frac{1}{\sqrt{2}}\left(\hat{b}_{2y}^{\dagger} - \hat{b}_{1y}^{\dagger}\right).
\end{aligned} \tag{1.174}$$

By introducing *a*s from (1.174) into (1.155), we obtain

$$\begin{aligned}
|\Psi^{-}\rangle_{12\text{in}} &\to -\frac{1}{\sqrt{2}}\left(\hat{b}_{1x}^{\dagger}\hat{b}_{1y}^{\dagger} - \hat{b}_{2y}^{\dagger}\hat{b}_{2x}^{\dagger}\right)\varnothing\rangle = \frac{1}{\sqrt{2}}\left(|01\rangle_{11\text{out}} + |01\rangle_{22\text{out}}\right) \\
|\Psi^{+}\rangle_{12\text{in}} &\to \frac{1}{\sqrt{2}}\left(\hat{b}_{1x}^{\dagger}\hat{b}_{2y}^{\dagger} - \hat{b}_{1y}^{\dagger}\hat{b}_{2x}^{\dagger}\right)\varnothing\rangle = |\Psi^{+}\rangle_{12\text{out}} \\
|\Phi^{\mp}\rangle_{12\text{in}} &\to \frac{1}{2\sqrt{2}}\left[\hat{b}_{1x}^{\dagger}\hat{b}_{1x}^{\dagger} \pm \hat{b}_{1y}^{\dagger}\hat{b}_{1y}^{\dagger} - \left(\hat{b}_{2x}^{\dagger}\hat{b}_{2x}^{\dagger} \pm \hat{b}_{2y}^{\dagger}\hat{b}_{2y}^{\dagger}\right)\right]\varnothing\rangle \\
&= \frac{1}{2}|(\Phi^{\pm}\rangle_{11\text{out}} - |\Phi^{\pm}\rangle_{22\text{out}}) .
\end{aligned} \tag{1.175}$$

Hence, $|\Phi^{\mp}\rangle$ and photons from $|\Psi^{-}\rangle$ (either $|01\rangle$ or $|10\rangle$) bunch either left or right and $|\Psi^{+}\rangle$ splits in agreement with [222, Eqs. (4.32–4.34), p. 69].

This shows that the Bell states do not preserve their "identity" at a dielectric beam splitter and this property can be used in Bell state detection schemes. Such a scheme can be more efficient than an analogous scheme with a metallic beam splitter simply because the most efficient dielectric beam splitters have about ten times lower losses than the most efficient metallic beam splitters. This is why they were used in, for example, [280].

1.15
Manipulating and Verifying Entanglements: Superdense Coding

In this section, we will investigate whether we can be more efficient in sending information through a circuit or over a network with qubits than we are with bits. To this aim, we shall make use of two entangled particles, but we will manipulate only one of them.

In doing so, we make use of a property of quantum entanglement[23] that when we manipulate one of two quantum particles entangled together, we actually manipulate the other one as well – without touching it as illustrated by Figure 1.32. Their states are correlated, but not deterministically as with classical particles. For a difference to appear with measured values of clicks in an arrangement as in Figure 1.32, we have to choose two angles for P_1 and two for P_2 to carry out measurements. We then introduce the values obtained for any of the Bell states the source

23) Classical entanglement does not possess this property.

photons are in, into the well-known *Bell inequality* [19] to obtain results we can never obtain with classical particles. The inequality is statistical and therefore we need a number of repeated measurements to reveal a difference between quantum and classical particles.

To reveal a difference without the Bell inequalities, we shall use a nonmaximal Bell state, say

$$|\Psi^{+}_{\text{non-max}}\rangle = \frac{1}{\sqrt{\alpha^2+\beta^2}}(\alpha|01\rangle + \beta|10\rangle) \,, \tag{1.176}$$

because this experiment would not work with a maximal Bell state $|\Psi^{+}\rangle$ [118, 304, 305].

We present the experiment carried out by Torgerson, Branning, and Mandel [304]. It considers two perpendicularly polarized photons incident at a dielectric beam splitter as in Figure 1.28. We make use of (1.107) with

$$t_x^2 = t_y^2 = 0.32 \quad \text{and} \quad r_x^2 = r_y^2 = 0.68 \tag{1.177}$$

(where $t_j^2 + r_j^2 = 1,\ j = x, y$) to obtain

$$|\Psi^{+}_{\text{non-max}}\rangle = 0.43|01\rangle + 0.91|10\rangle \,, \tag{1.178}$$

and we choose angles at which we orient our polarizing beam splitters as follows

$$\theta_{1a} = 18^\circ \,, \quad \theta_{1b} = -56^\circ \,, \quad \theta_{2a} = 72^\circ \,, \quad \theta_{2b} = -34^\circ \,, \tag{1.179}$$

where θ_{1a}, θ_{1b} refer to two orientations of the polarizing beam splitters (PBS) in front of D_1 and $D_1^{\perp}$, and θ_{2a}, θ_{2b} to two PBS orientations in front of D_2 and $D_2^{\perp}$.

Following the procedure by which we obtained the probability of coincidence detection at a symmetric beam splitter (1.110) in Section 1.13, we now get

$$\frac{P(\theta_{1a}, \theta_{2b})}{P(\theta_{1a})} = \frac{P(18^\circ, -34^\circ)}{P(18^\circ)} = 1 \,, \quad \frac{P(\theta_{1b}, \theta_{2a})}{P(\theta_{2a})} = \frac{P(-56^\circ, 72^\circ)}{P(72^\circ)} = 1 \,, \tag{1.180}$$

where $P(\theta_{1a})$ [$P(\theta_{1a})$] means detecting photon 1 (2) at one output of the polarizing beam splitter (PBS) oriented along θ_{1a} [θ_{1a}] and photon 2 (1) at both outputs of their PBS. The schematic of the arrangement of PBSs is shown in Figures 1.36 and 1.37.

Equation (1.180) tells us that if photon 1 passes the PBS oriented at angle θ_{1a}, then photon 2 will pass the analyzer set at angle θ_{2b} with probability 1 and if photon 2 passes the PBS set at θ_{2a}, then photon 1 will pass the PBS set at θ_{1b} with probability 1. Classically, since these events occur deterministically (i.e., with probability 1), we could assume that there is a property of each photon related to their passage through PBSs – Einstein would call it an "element of physical reality associated

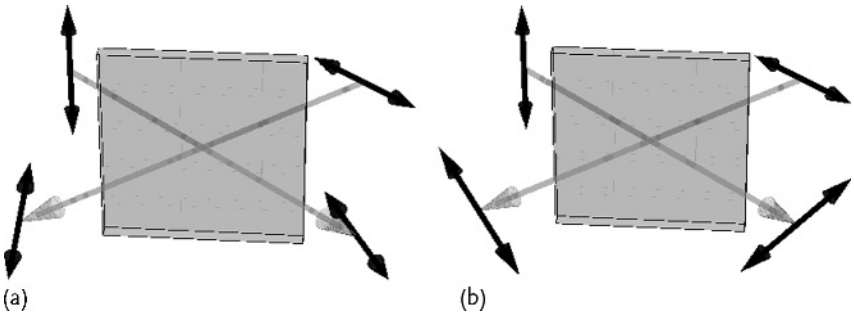


Figure 1.36 Deterministic difference between "classical" and quantum photons. The measurement – was carried out by Torgerson, Branning, and Mandel [304]. Photons come to the beam splitter perpendicularly polarized. After exiting from the other side, they are filtered by PBSs set at angles of (a) 18° and −34° and (b) −56° and 72°. Photons pass them with probability 1.

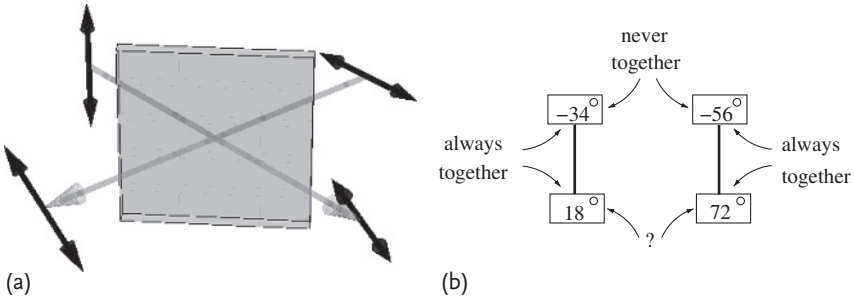


Figure 1.37 Hypothetical elements of physical reality. (a) After exiting the beam splitter, photons enter PBSs set at angles of −56° and −34°. Photons never pass both of them; (b) Classically, $\boxed{18^\circ}$ and $\boxed{72^\circ}$ should never pass PNSs together. But, qubits are quantum particles.

with polarization" in the famous Einstein–Podolsky–Rosen (EPR) Gedankenexperiment[24] [90]. This is as if photons had carried definite polarizations before detections that would then be confirmed after passing through PBSs set at angles given by (1.179) and shown in Figure 1.36. Let us denote these hypothetical "elements of physical reality" by

$$\boxed{18^\circ}\,, \quad \boxed{-56^\circ}\,, \quad \boxed{72^\circ}\,, \quad \text{and} \quad \boxed{-34^\circ}\,. \tag{1.181}$$

Now, we carry out the following measurement, shown in Figure 1.37, by

$$P(\theta_{1b}, \theta_{2b}) = P(-56^\circ, -34^\circ) = 0\,, \tag{1.182}$$

and realize that we can never obtain clicks in both detectors behind PBSs oriented at $\boxed{-56^\circ}$ and $\boxed{-34^\circ}$.

24) Thought experiment.

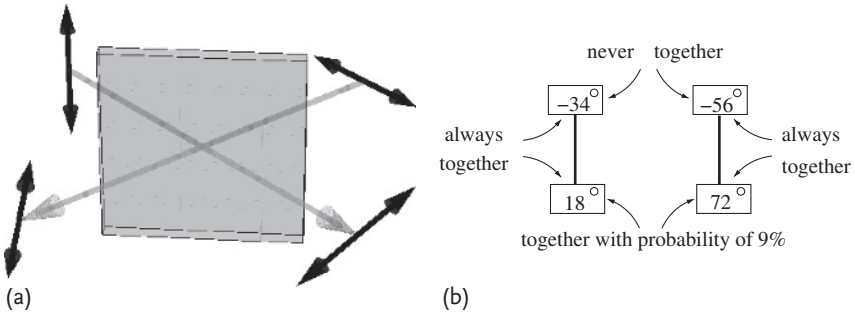


Figure 1.38 Hypothetical elements of physical reality disproved. (a) After exiting the beam splitter, photons enter PBSs set at angles of 18° and 72°. Photons pass both of them with the probability of 9% (b).

Taken together – see Figure 1.37, "elements" $\boxed{18°}$ and $\boxed{-34°}$ always occur together. Elements $\boxed{-56°}$ and $\boxed{72°}$ also always occur together. Elements $\boxed{-56°}$ and $\boxed{-34°}$, on the other hand, never occur together. Now, if these elements really were photon properties, then element $\boxed{18°}$ and $\boxed{72°}$ could never occur together either. To test this assumption, we calculate (see Figure 1.38):

$$P(\theta_{1a}, \theta_{2a}) = P(18°, 72°) = 0.09 \,. \tag{1.183}$$

Thus, we have the probability of 9% for the elements $\boxed{18°}$ and $\boxed{72°}$ to occur together and this disproves the assumption and proves the following.

Property 47

There can be no "elements of physical reality"

$$\boxed{18°}\,, \quad \boxed{-56°}\,, \quad \boxed{72°}\,, \quad \text{and} \quad \boxed{-34°}\,,$$

that is, there are no such properties which we could ascribe to our photons.

This also means that two quantum particles can be in more states than classical particles and that they can transfer more information than classical ones. And, this is exactly what superdense and dense coding are about. If we "entangle" classical particles, the entanglement will be deterministic and when we flip the state of one of them, the states of the other one will flip deterministically too, and we do not obtain any new information – one bit, or two messages, no matter whether we have one or two "classically entangled particles." Let us now see how we can be much better than that.

The entangled states that we get from a type-II and two combined type-I nonlinear crystals are described by

$$|\Psi\rangle = \frac{1}{\sqrt{2}}(|01\rangle + e^{i\theta}|10\rangle)\,, \quad \text{and} \quad |\Phi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + e^{i\theta}|11\rangle)\,, \tag{1.184}$$

respectively. (See Figures 1.27 and 1.26 and the text that follows them.)

By means of an additional birefringent phase shifter or just by a slight rotation of a crystal, we can set $\theta = 0$ and $\theta = \pi$ to obtain $|\Psi^{\pm}\rangle$ and $|\Phi^{\pm}\rangle$. We can also rotate a birefringent crystal in the path of one of the photons to switch the phase from $\theta = 0$ and $\theta = \pi$ and back, that is, perform $|\Psi^{+}\rangle \leftrightarrow |\Psi^{-}\rangle$ and $|\Phi^{+}\rangle \leftrightarrow |\Phi^{-}\rangle$.

However, for an implementation of Bell states in a realistic device, it seems that the most efficient would be to generate only one of the Bell states, say $|\psi^{+}\rangle$, by a down-conversion in a nonlinear crystal, and then to obtain the other three states by inserting half-wave plates in the path of one of the entangled photons. Let us see how we can do that in some detail.

A half-wave plate (HWP) is a phase shifter which rotates polarization states. We use horizontal and vertical polarizations as a polarization basis. The plate is a uniaxial birefringent crystal whose optical axis and the direction of the horizontal polarization enclose angle θ. It is represented by the following matrix:

$$\mathrm{HWP}(\theta) = \begin{pmatrix} \cos 2\theta & \sin 2\theta \\ \sin 2\theta & -\cos 2\theta \end{pmatrix} = \sin 2\theta\, \sigma_x + \cos 2\theta\, \sigma_z \,. \tag{1.185}$$

For $\theta = 0$, we get $\mathrm{HWP}(0) = \sigma_z$ which we can use to change the phase of the vertical polarization:

$$\begin{aligned} &\mathrm{HWP}(0)|0\rangle = \sigma_z|0\rangle = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle = |H\rangle = |\leftrightarrow\rangle \,; \\ &\mathrm{HWP}(0)|1\rangle = -|1\rangle = -|V\rangle = -|\updownarrow\rangle \,. \end{aligned} \tag{1.186}$$

$\theta = \pi/4$ yields $\mathrm{HWP}(\pi/4) = \sigma_x$ which we can use to carry out a polarization flip:

$$\begin{aligned} &\mathrm{HWP}\left(\frac{\pi}{4}\right)|0\rangle = \sigma_x|0\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle \,; \\ &\mathrm{HWP}\left(\frac{\pi}{4}\right)|1\rangle = |0\rangle \,. \end{aligned} \tag{1.187}$$

With $\theta = \pi/8$, we get $\mathrm{HWP}(\pi/8) = \mathrm{HWP}(22.5) = (\sigma_x + \sigma_z)/\sqrt{2}$ which we can use to change linear polarization to diagonal and vice versa. Experimentally, it means that when we turn a HWP (put in the path of a photon) for 22.5 with respect to the direction of the horizontal polarization it rotates the planes of H and V polarization for 45°,

$$\begin{aligned} \mathrm{HWP}\left(\frac{\pi}{8}\right)|0\rangle &= \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \\ &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |\mathrm{D}^{+}\rangle = |\nearrow\rangle \,; \\ \mathrm{HWP}\left(\frac{\pi}{8}\right)|1\rangle &= \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \\ &= \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = -|\mathrm{D}^{-}\rangle = -|\nwarrow\rangle \,, \end{aligned} \tag{1.188}$$

where

$$\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \sigma_x + \sigma_z = \sqrt{2}H \; . \tag{1.189}$$

$H = 1/\sqrt{2}(\sigma_x + \sigma_z)$ is known as the *Hadamard transformation* or *Hadamard gate*. We shall come back to it later on.

For $\theta = \pi/2$, we get HWP(90°) $= -\sigma_z$ which we can use to change the phase of the horizontal polarization:

$$\begin{aligned} \text{HWP}(90)|0\rangle &= -\sigma_z|0\rangle = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \end{pmatrix} \\ &= -|0\rangle = -|H\rangle = -|\leftrightarrow\rangle \; ; \\ \text{HWP}(90)|1\rangle &= |1\rangle = |V\rangle = |\updownarrow\rangle \; . \end{aligned} \tag{1.190}$$

Before we dwell on the superdense coding itself, we shall first determine how our HWPs act on Bell states. First, let us look at the Bell state preparation. However, even before that, let us get introduced to *Alice* and *Bob*. They will be our hosts in many descriptions and figures of various implementations of our designs and protocols. Nowadays, one hardly finds any papers on new quantum cryptography protocols without Alice and Bob. Alice usually sends a message encoding it so as to apply her part of a protocol and Bob deciphers it by applying his part. Often, *Eve* will attempt to *eave*sdrops on their messages, but Eve's role is a very sad one because quantum mechanics prevents her from ever succeeding in her attempts – in principle.

We start with $|\Psi^+\rangle$. HWPs will always act on the right photon in Figure 1.39. Let it be photon 1. Alice first only puts HWP(0°) into this arm to get $|\Psi^-\rangle$, that is,

$$|\Psi^+\rangle \longrightarrow \text{HWP}(0)\frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle) = |\Psi^-\rangle \; . \tag{1.191}$$

Next, Alice takes out HWP(0°) and puts in HWP(45°) to get $|\Phi^+\rangle$:

$$|\Psi^+\rangle \longrightarrow \text{HWP}\left(\frac{\pi}{4}\right)\frac{1}{\sqrt{2}}(|01\rangle+|10\rangle) = \frac{1}{\sqrt{2}}(|11\rangle+|00\rangle) = |\Phi^+\rangle \; . \tag{1.192}$$

To get $|\Phi^-\rangle$, she now puts in HWP(0°) after HWP(45°):

$$\begin{aligned} |\Psi^+\rangle \rightarrow \text{HWP}(0)\text{HWP}\left(\frac{\pi}{4}\right)\frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) &= \frac{1}{\sqrt{2}}(-|11\rangle + |00\rangle) \\ &= |\Phi^-\rangle \; . \end{aligned} \tag{1.193}$$

Bob first tries to identify which Bell states Alice has sent him by only using PBSs (polarizing beam splitters, which let horizontally polarized photons ($|0\rangle$) through and reflect vertically polarized ones ($|1\rangle$)). In Figure 1.39, Bob put his elements to the left and right of Alice's for us to better see how his detection will proceed.

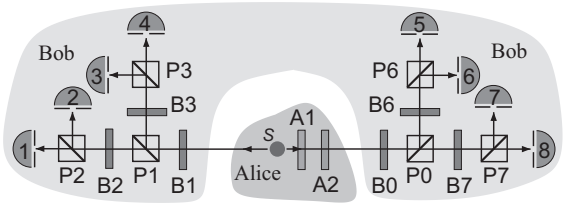


Figure 1.39 Bell state analyzer. Alice's source S produces $|\Psi^+\rangle$. She puts both, one, or none of her HWPs (HWP(45°) and HWP(0°)) in the path of one of the photons from the source to obtain $|\Psi^\pm\rangle$ and $|\Phi^\pm\rangle$. Bob uses his HWP (B0-B7), PBSs, and detectors 1-8 in an attempt to find out which Bell states Alice has sent him, but he cannot distinguish more than two states.

However, in a realistic experiment, Bob's elements can be located in a place which is miles away from Alice's. Alice can send him her two photons over two parallel fibers.

Bob finds out that without wave plates (WP) B1 and B2, photons in both states, $|\Psi^-\rangle$ and $|\Psi^+\rangle$, would either pass PBS P_0 and be reflected from P_1 or the other way around. So, they would be either detected by detectors 3 and 8 or by detectors 1 and 6. Bob can put various plates P_2–P_7 in the photons path, but that would not enable him to distinguish $|\Psi^-\rangle$ from $|\Psi^+\rangle$ because photons from these Bell states disentangle after passing P_1 and P_0.[25] Photons from the pairs receive opposite linear polarization. So, Bob decides instead to put various HWs in positions B0 and B1.

Two HWP($\pi/8$) in positions B0 and B1 act on $|\Psi^+\rangle$ as

$$\mathrm{HWP}\left(\frac{\pi}{8}\right)|\Psi^+\rangle = \frac{1}{\sqrt{2}}\left[\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) + \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\right] = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle) = |\Phi^-\rangle\,. \quad (1.194)$$

For $|\Psi^-\rangle$, we – in effect – get $|\Psi^-\rangle$ out again:

$$\mathrm{HWP}\left(\frac{\pi}{8}\right)|\Psi^-\rangle = \frac{1}{\sqrt{2}}(|10\rangle - |01\rangle) = -|\Psi^-\rangle\,; \quad (1.195)$$

since an overall constant phase factor (-1) does not make two functions experimentally distinguishable.

$|\Phi^+\rangle$ also remains unaffected:

$$\mathrm{HWP}\left(\frac{\pi}{8}\right)|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) = |\Phi^+\rangle\,. \quad (1.196)$$

$|\Phi^-\rangle$ is turned into $|\Psi^+\rangle$:

$$\mathrm{HWP}\left(\frac{\pi}{8}\right)|\Phi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) = |\Psi^+\rangle\,. \quad (1.197)$$

25) It might look too obvious in Figure 1.39 that we cannot gain any more information on the incoming system from P_2–P_7 with respect to P_1 and P_2, but this is only because we do not keep track of the phase difference here. If we had kept it, as we had done in Figure 1.49, we could have gained additional information.

So PBSs P_0 and P_1 will receive the same functions as without HWPs B0 and B1. QWP($\pi/2$) instead of HWPs will give the a similar result. (For instance, QWP($\pi/2$)$|\Psi^+\rangle = |\Phi^+\rangle$, and so on.) This is a consequence of the inability of single linear optical elements to distinguish a phase difference between individual photons. When we do separate measurements on two entangled photons that are entangled in the states $|\Psi^\pm\rangle$, we cannot distinguish them by means of HWPs and PBS since $|\Psi^+\rangle$ will turn into $|\Phi^-\rangle$ and $|\Psi^-\rangle$ will remain $|\Psi^-\rangle$. In the latter case, the photons would again be either detected by detectors 3 and 8 or by detectors 1 and 6. However, in the former case, they would either go through P_0 and P_1, respectively, or be reflected from P_0 and P_1, respectively. Thus, they would be detected either by detectors 1 and 8 or by detectors 3 and 6. Bob has to conclude that with the setup from Figure 1.39, he cannot distinguish more than two of the Bell states and that Alice cannot send more than two messages, that is, one bit, namely, the classical limit presented above.

Bob could distinguish more than two Bell states, but only by using a probabilistic approach and a beam splitter (see (1.135)–(1.140)), similarly to what we obtained above with 0.09 probability of (1.183). But, probabilistic measurements cannot be efficiently used for sending messages.

On the other hand, for deterministic discriminations of Bell states, it has been proved by Lütkenhaus' [184] and Vaidman's [309] groups that one cannot distinguish all Bell states using only linear optics. Moreover, when no conditional measurements are being used, then one cannot deterministically discriminate more than two Bell states, not even with the help of interferometers [60].

So, Alice and Bob decide to investigate how far they can go in transferring messages using only linear optics elements. They realize that they can use a beam splitter (say, a metallic one) to separate Bell states as we did in Section 1.14. Bob merges photons of each of Alice's Bell state at a beam splitter to separate singlet $|\Psi^-\rangle$ from triplets $|\Psi^+\rangle$ and $|\Phi^\pm\rangle$ using the results we obtained in Sections 1.13 and 1.14. In Section 1.13 (see the boxed conclusion after (1.114)), we calculated out that any two photons – incident at two opposite sides of a beam splitter – whose polarizations are not strictly parallel would emerge from the opposite sides of the beam splitter with some probability. When their incident polarizations are parallel, the probability is 0% as shown by the dip in Figure 1.30a. In Section 1.14 (1.135), we showed that when the photons, incident at a metallic polarization preserving beam splitter, are in a Bell state, the probability of them emerging at opposite sides of the beam splitter is 100%. And finally, (1.140) shows that incident states $|\Psi^+\rangle$ and $|\Phi^\pm\rangle$ would all reappear from the same sides of the beam splitter with probability 1.

Alice prepares the Bell states as before and Bob analyzes them with the setup shown in Figure 1.40. When Alice sends $|\Psi^-\rangle$, the photons will emerge from the opposite sides of the beam splitter BS and will trigger either detector 1 or 2 in coincidence with either 3 or 4. When Alice sends $|\Psi^+\rangle$, both photons – perpendicularly polarized – emerge either from the left or from the right side of BS and they will trigger either 1 and 2 in coincidence or 3 and 4 in coincidence. However, Bob cannot distinguish between $|\Phi^-\rangle$ and $|\Phi^+\rangle$ because parallelly polarized photons

emerging from the same side of BS will either both pass or both be reflected in the PBSs because they are entangled (as soon as one of the two photons passes a PBS, the other will pass it as well, and as soon as one of the two photons is reflected from a PBS, the other will be reflected too).

To detect either $|\Phi^+\rangle$ or $|\Phi^-\rangle$ would ideally require just one click (for both photons) in any of the detector. However, since we still do not have reliable and efficient source of photon pairs on demand, we have to make use of detectors that can tell one from two photons in order to be sure that both detectors arrived to detectors and that we really detected $|\Phi^\pm\rangle$ photons. The highest efficiency of such detectors is currently 90% [114, 270]. This is lower than the efficiency of single-photon detectors whose efficiency has reached 99% [173, 213], but still much higher than the highest efficiency of a source of photon pairs on demand which is currently under 0.1% [16, 150].

Now, Alice and Bob want to use their qubit device for sending messages from Alice to Bob and compare it with an analogous classical bit device for sending messages. Alice uses two HWPs to handle just one of the two qubits from $|\Psi^-\rangle$ which she receives from the source – a down-converting BBO crystal (see Section 1.12 and Figure 1.27). The way she obtains $|\Psi^\pm\rangle$ and $|\Phi^\pm\rangle$, and sends them to Bob is the same as described for Figure 1.39. Bennett and Wiesner realized that if Bob were able to distinguish all four states, then Alice would accomplish a treat of sending a message twice as effectively as with a classical particle [24]. The number of bits (m) we can send by means of n messages is $\log_2 2^n = m$. Thus, using a classical particle, Alice would be able to send Bob just one bit of information $\log_2 2^1 = 1$ because the number of messages (n) one can send with the help of a "classical photon" is 2: $\updownarrow$ and $\leftrightarrow$.

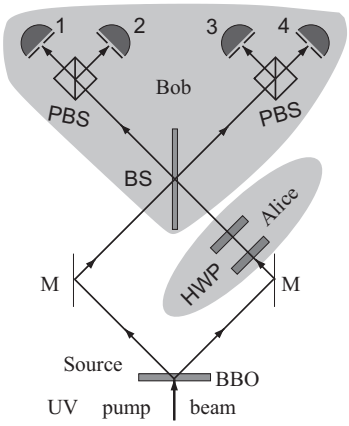


Figure 1.40 Dense coding. The Bell basis gives a dense coding. Alice sends $|\Psi^\mp\rangle$, $|\Phi^\mp\rangle$. $|\Psi^-\rangle$ splits at the beam splitter BS and $|\Psi^+\rangle$ splits at either the left or the right polarizing beam splitter. $|\Phi^\mp\rangle$ do not split on any of these beam splitters, but bunch together. A 2-photon-click of one of the number resolution detectors 1–4 means a detection of either $|\Phi^-\rangle$ or $|\Phi^+\rangle$.

By manipulating a single qubit as described above, and shown in Figure 1.40, Alice is able to send only three messages to Bob $|\Psi^{\pm}\rangle$ and either $|\Phi^{-}\rangle$ or $|\Phi^{+}\rangle$. So she can send Bob $\log_2 3 = 1.58$ bits of information. This is 50% better than via a classical particle. This 1.58 linear optics limit of distinguishing Bell states is called a *dense coding* according to the first such experiment carried out by Zeilinger's group [190].

In the literature, the dense coding is often rendered "deterministic" in the sense that Alice and Bob might agree to make use of only one of the two $|\Phi\rangle$ states (either $|\Phi^{-}\rangle$ or $|\Phi^{+}\rangle$) or Alice and Bob take both $|\Phi^{-}\rangle$ and $|\Phi^{+}\rangle$ to be one and the same message. In a fully implemented quantum computation circuit, such an assumption is not acceptable because that would mean that we discard one of the four possible well defined states in the 4-dim Hilbert space of two computing qubits. Hence, it is more appropriate to call it indeterministic with an efficiency of 75% (100% for $|\Psi^{\mp}\rangle$ and 50% for $|\Phi^{\mp}\rangle$) [242].

But, if Alice managed to send all four messages to Bob, then she would send $\log_2 4 = 2$ bits of information. Twice as much as with a classical particle [24]. This is called a *superdense coding*.

Definition 48 Encoding two entangled qubits so as to enable sending three messages and therefore 1.58 bits of information by manipulating just one of the qubits is called a *dense coding*.

Definition 49 Encoding two entangled qubits so as to enable sending four messages and therefore 2 bits of information by manipulating just one of the qubits is called a *superdense coding*.

With the setup from Figure 1.40, one cannot implement a superdense coding. Let us reconsider the properties of the linear optics we use in the setup to see why and to find out how we can go around the obstacles. In Sections 1.13 and 1.15, we found:

Property 50

Two photons simultaneously incident at a polarization-preserving beam splitter ideally behave as follows.

1. For two unpolarized photons, which are perpendicularly correlated in polarization, and which enter a beam splitter from its opposite sides, the following holds:
   a) photons in the Bell state $|\Psi^{-}\rangle$ always emerge from the opposite sides of a beam splitter and never from the same ones and
   b) photons in the Bell state $|\Psi^{+}\rangle$ always emerge from the same sides of a beam splitter and never from the opposite ones; they appear in either $|01\rangle$ or $|10\rangle$ state.

Therefore, $|\Psi^+\rangle$ splits at the first PBSs after BS in Figure 1.40, although we can never tell whether a horizontally polarized photon detected, for example, by D4, came from Alice or directly from the source.

2. Two unpolarized photons which are parallelly correlated in polarization that enter a beam splitter from its opposite sides never emerge from its opposite sides; among such photons,
   a) photons in both Bell states $|\Phi^+\rangle$ and $|\Phi^-\rangle$ always emerge from the same sides of a beam splitter and never from the opposite ones;
   b) They cannot be distinguished from each other because they disentangle at the first PBSs and afterwards they are in the product state $|00\rangle$ or the product state $|11\rangle$.
3. Two parallel polarized or unpolarized but parallelly correlated in polarization photons that enter a beam splitter from the same sides emerge from its opposite sides with probability 0.5 and from the same sides also with probability 0.5 (see Figure 1.35).

One way to overcome the impossibility of transferring all four Bell states is to use a larger Hilbert space by adding energy or momentum degrees of freedom to the polarization one. States that are not only entangled in polarization but also in energy and/or momentum are called *hyperentangled states*. Let us consider the energy entanglement imposed on photons by their creation in the process of down-conversion. In a down-conversion, a photon from a pump beam (of energy E_0 and frequency ω_0) yields signal and idler photons whose energies E_1, E_2 and frequencies ω_1, ω_2, respectively, must satisfy relations: $E_0 = E_1 + E_2$ and $\omega_0 = \omega_1 + \omega_2$. Since down-conversion happens within femtoseconds, there is always a time correlation between an idler and a signal from the same pair. To make use of such entanglement for distinguishing all four Bell states was first proposed by Kwiat and Weinfurter for an experiment they called *embedded Bell-state analysis* [161]. The experiment is shown in Figure 1.41.

A click of either D_1 or D_2 in coincidence with either D_3 or D_4 means a detection of $|\Psi^-\rangle$. Birefringent elements cause a delay that a horizontally polarized photon experiences with respect to a vertically polarized one. Thus, a delay of ~ 1 ns that the coincidence detectors would detect means a detection of $|\Psi^+\rangle$. $|\Phi^\pm\rangle$ will not experience a delay because both photons are of the same polarization. The problem with the detection is that HWP(22.5°) will turn it into $|\Psi^+\rangle$ and therefore both photons will finish in a single detector. However, HWP is necessary for distinguishing $|\Phi^+\rangle$ from $|\Phi^-\rangle$ because HWP(22.5°) will turn $|\Phi^-\rangle$ into $|\Psi^+\rangle$ according to (1.197), and $|\Phi^+\rangle$ will remain unaffected (see (1.196)). Thus, firing of D_1 and D_2, and D_3 and D_4 means a detection of $|\Phi^-\rangle$ and a detecting of two photons in either D_1, or D_2, or D_3, or D_4 without a delay means $|\Phi^+\rangle$ and with a 1ns delay means $|\Psi^+\rangle$. The detector's task of distinguishing between two photons and additionally, detecting a delay of 1 ns, is a very demanding one. The lowest time resolution of single photon detectors is about 0.5 ns. Then, in order to distin-

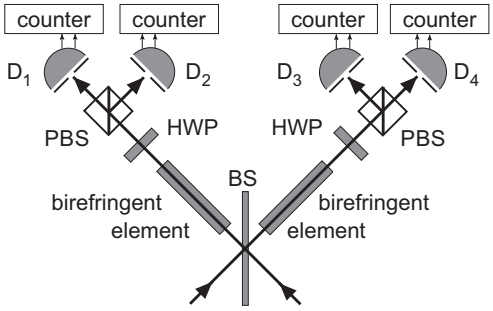


Figure 1.41 Superdense coding – hyperentanglement. Embedded Bell-state analysis [161] makes use of hyperentangled two photon states, entangled in polarization and energy (time). HWP(22.5°) turns $|\Phi^-\rangle$ into $|\Psi^-\rangle$ and PBS splits the photons. $|\Psi^+\rangle$ and $|\Phi^+\rangle$ go into one of D_1–D_4 and counters distinguish them according to 1 ns and 0 ns delays, respectively. $|\Psi^-\rangle$ is split at the beam splitter BS.

guish photons from each other we have to apply the splitting we used for analogous detection in Figure 1.43.

There have been several rather sophisticated experiments with hyperentangled two photon states recently [10, 12, 13, 143, 207, 280]. As opposed to a linear optics approach, they all have two drawbacks. (1) Additional degrees of freedom remain unused for encoding messages, that is, entanglement in momentum and/or energy only serve for a successful recognition of polarization Bell states and therefore for transferring 2 bits of information and polarization superdense coding but do not take part in codification themselves; (2) The experiments are very demanding and although all design of these experiments are capable of transferring 2 bits of information, only the one carried out by Kwiat's group [13] succeeded in beating the linear optics dense coding limit 1.585 by a narrow margin, reaching 1.630. However, the latter experiment succeeded in beating the 1.585 limit because it was limited to superdense coding, as opposed to many other hyperentangled designs that were designed to include teleportation as well (see Section 1.16).

Thus, Alice and Bob decided to reconsider a linear optics approach once again. They recall that Lütkenhaus' proof [184] allows them to achieve a *superdense coding* with photons in the Bell states only near-deterministically (see Section 1.16). But, what about another possible basis? In some other basis, photons might not be entangled when they reach Bob, but Alice and Bob take a minimalist approach – they just want to prove that it is possible to transfer four messages by manipulating only one of the qubits and they want to be as cheap as possible. In order to reach their goal, Alice and Bob first write down all conclusions they have reached so far.

1. $|00\rangle$ and $|11\rangle$ always emerge from the same sides of a beam splitter and can be distinguished form each other even after passing through a PBS, while $|01\rangle$ and $|10\rangle$ emerge from the same and opposite sides and cannot be distinguished in 50% of occurrences.

2. $|\Phi^-\rangle$ and $|\Phi^+\rangle$ always emerge from the same sides of a beam splitter and cannot be distinguished form each other, while $|\Psi^+\rangle$ and $|\Psi^-\rangle$ emerge from the same and opposite sides and can always be distinguished.
3. If Alice sends her qubit through a polarizer oriented along $\updownarrow$ or $\leftrightarrow$, the other qubit from the entangled pair in states $|\Phi^\pm\rangle$ would be immediately set into $\updownarrow$ or $\leftrightarrow$ polarization state, that is, into $|1\rangle$ or $|0\rangle$.
4. One can improve the efficiency of detection of parallelly polarized photons in a same beam by directing them repeatedly to beam splitters from one side as shown in Figure 1.35.

"You see," says Alice, "we have two well behaved states from the computational basis and two well behaved ones from the Bell basis. Why don't we combine them? All bases are equivalent, aren't they?" "Yes, indeed," admits Bob. "Let us combine them into a *mixed basis*. It is not a basis of entangled states – $|00\rangle$ or $|11\rangle$ do not represent an entangled state – but it will serve our purpose of showing that superdense coding can be carried out with photons by means of linear optics, that is, without superentanglement."[26]

Definition 51 A *mixed basis*[27] for two qubits is the orthonormal set of states

$$|\chi^1\rangle = |\Psi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)\,, \quad |\chi^2\rangle = |\Psi^+\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)\,,$$
$$|\chi^3\rangle = |00\rangle\,, \quad |\chi^4\rangle = |11\rangle\,, \tag{1.198}$$

which spans the Hilbert space $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2$.

Equations (1.114), (1.124), and (1.125) clearly show that both, the computational basis $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$ and the Bell basis $|\Psi^\pm\rangle$, $|\Phi^\pm\rangle$, can easily be expressed in the mixed basis. Also, we can easily check that states $|\chi^i\rangle$, $i = 1, \ldots, 4$ are really orthogonal and have the norm 1.

In the absence of a well defined and unique application of the superdense coding, it can be given the following three possible operational definitions and implementations.

a) In the original version of their superdense coding, Bennett and Wiesner [24] assume that Bob sends Alice one qubit from each of his pairs. She manipulates her qubits and sends it back to Bob. Bob expects Alice to return each qubit he sent her.
b) In our version of superdense coding with a mixed basis, Alice knows that some of her qubits are not coded as she wanted. She calls such qubits "*wrong*" qubits. She might decide to proceed as in (c) below or just discard the "wrong" qubits.

26) We stress here, according to Definition 49, that the superdense coding is about transferring messages, not about transferring states. Hence, for the purpose of transferring messages, it is irrelevant whether the photons are entangled when Bob receives them, or not.

27) Adán Cabello [49, 51] also makes use of the mixed basis, but not in the content of previously entangled photon pairs.

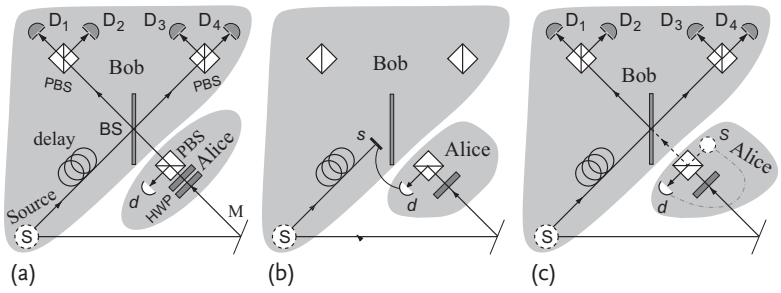


Figure 1.42 Extensions of superdense coding with the mixed basis. (a) Alice puts both, one, or none of her (half-wave plates) HWPs and she either puts or not her PBS in the path of her qubit; (b) *interactive fair-sampling superdense coding*. "Wrong" photons trigger a shutter (*s*) at Bob's side which blocks the other photon; (c) *controlled superdense coding*. Each "wrong" photon triggers detector *d* which in turn triggers a single-photon source (*S*) which emits a cloned photon to Bob. Bob uses his PBSs and detectors D_j to detect $|\chi^j\rangle$, $j = 1, \ldots, 4$. Before they start, they tune in their PBSs. *S* is the source of photon pairs.

While carrying out the latter option, she assumes that Bob applies the *fair sampling assumption* on the detections he records. Under this assumption, only samples of detected qubit pairs belong to the set of coded qubits – single-photon detections do not. We name such coding the *fair-sampling superdense coding*. Instead of Bob just discarding one-photon recoding of the "wrong" messages, Alice's "wrong" photon detections can trigger a shutter at Bob's side which blocks the other photon from each such pair – shown in Figure 1.42b. We call this interactive coding the *interactive fair-sampling superdense coding*.

c) Alice has "control" over her "wrong" qubits in the sense of knowing what she sent. We have two possible scenarios here. (i) Bob receives a single photon, know it is "wrong" because it is not accompanied by an Alice's photon and corrects (bit-flips) the message; (ii) Alice clones the "wrong" photon and sends it to Bob together with a classical message on that; Bob corrects the "wrong" message. We call option (ii) a *controlled superdense coding*.[28)]

These three operational definitions are shown in Figure 1.42.

Bob generates a pair of photons entangled in state $|\Psi^+\rangle$ by means of a down-conversion process in a BBO crystal [190] and sends one of the two photons to Alice. To send $|\chi^2\rangle = |\Psi^+\rangle$ to Bob, Alice puts nothing in the path of her photon. To send $|\chi^1\rangle = |\Psi^-\rangle$, she puts in HWP(0°) (half-wave plate) in the path. It changes the sign of the vertical polarization. To send $|\chi^3\rangle$, she takes out HWP(0°) and puts in HWP(45°) and a polarizing beam splitter (PBS) type so that she can have a feedback about a photon that did not pass through her PBS ($|0\rangle$), but was reflected from it ($|1\rangle$). HWP(45°) turns $|\Psi^+\rangle$ into $|\Phi^+\rangle$ and PBS projects both photons to state $|0\rangle$ in half of the occurrences. In the other half of the occurrences, Alice's photon is

28) However, this is not a superdense coding in its original Bennett–Wiesner sense which does not include a classical channel.

reflected from her PBS and we have both photons in state $|1\rangle$, that is, in state $|\chi^4\rangle$. To send $|\chi^4\rangle$, Alice proceeds analogously.

We stress here that the preparation of $|\chi^3\rangle$ and $|\chi^4\rangle$ includes physics of entangled systems because whenever Alice sends her qubit through a polarizer oriented horizontally or vertically, the other qubit from the entangled pair (originally in the state $|\Phi^+\rangle$) will be immediately set into $|0\rangle$ and $|1\rangle$ state for any subsequent measurement along $\leftrightarrow$ and $\updownarrow$ directions, respectively.

In the computational basis, two parallel polarized photons sent to a beam splitter from its opposite sides will always emerge from the same side, bunched together and showing the so-called Hong–Ou–Mandel interference dip [222, Section 3.2]. It has been calculated that both bunched photons keep the polarization direction they had before they entered the beam splitter [234, 239, 249]. So, we can discriminate $|\chi^3\rangle$ and $|\chi^4\rangle$ from each other and from $|\chi^{1,2}\rangle$ with photon number resolution detectors or up to an arbitrary precision with single-photon detectors. Perpendicularly polarized photons – the other two states of the computational basis – sent to a beam splitter, either bunch together (50%) or emerge from the opposite sides of the beam slitter (50%) [234]. The two photons that are split are correlated in polarization but unpolarized, that is, entangled. Therefore, we cannot distinguish between $|01\rangle$ and $|10\rangle$ in 50% of the events and we end up with the channel capacity $\log_2 3$.

In the Bell basis, we can discriminate between $|\Psi^+\rangle$ and $|\Psi^-\rangle$, but not between $|\Phi^+\rangle$ and $|\Phi^-\rangle$ [222, Section 4.1]. At a polarization preserving (metallic) BS, $|\Psi^-\rangle$ photons split and $|\Psi^+\rangle$ photons bunch together, but have orthogonal polarizations and we can split them at polarizing beam splitters (PBSs) behind BS. $|\Phi^\pm\rangle$ photons also bunch together in an entangled manner, that is, unpolarized but correlated in polarization, both photons from the pair project to either $|0\rangle$ or $|1\rangle$, that is, they randomly either both go through or are both reflected from a PBS. Hence, we cannot distinguish them and we again end up with the channel capacity $\log_2 3$.

By combining states that we can unambiguously discriminate – two from the computational basis, $|00\rangle, |11\rangle$, and two from the Bell basis, $|\Psi^\pm\rangle$ – we obtain the basis, $|\chi^i\rangle$, $i = 1, \ldots, 4$ which we can use to deterministically transfer two bits of information by means of ideal linear-optics elements shown in Figure 1.42 and ideal photon number resolution detectors as we present below. Below, we also show how we can deterministically transfer information arbitrarily close to two bits by means of single-photon detector devices shown in Figure 1.43.

In a preliminary tuning of the devices, Alice informs Bob when she sends him $|\chi^3\rangle$ and when $|\chi^4\rangle$ so that Bob can determine the directions along which he will orient his PBSs to (ideally) let $|0\rangle$ photons through and reflect $|1\rangle$ photons. So, $|\chi^3\rangle$ photons will exit through either port 1 or port 4 and $|\chi^4\rangle$ ones through either 2 or 3. Next, Bob identifies $|\chi^1\rangle = |\Psi^-\rangle$ by means of coincidence clicks of detector in paths 1 (2) and 3 (4). In the end, Bob identifies $|\chi^2\rangle = |\Psi^+\rangle$ by means of coincidence clicks of either detectors in paths 1 and 2 or detectors in paths 3 and 4.

Let us now take a closer look at the implementation details.

a) Bob sends one qubit from each pair to Alice. Alice sends $|\chi^1\rangle$ and $|\chi^2\rangle$ with an efficiency ideally approaching 100%. When sending $|\chi^3\rangle$ or $|\chi^4\rangle$, she only has

a 50% probability of success. Alice and Bob conclude that this scenario is too inefficient because when sending $|\chi^{3,4}\rangle$ neither Alice nor Bob know which of their messages were unsuccessful. So, they switch to the next option.

b) Alice manipulates just one photon, but she can stop the other if her photon chooses a "wrong" exit from her PBS. Alice knows when she was successful and when she was not because her detector will not click when she was and will click when she was not. Bob will not know when Alice's were "wrong." All the messages he will receive will be deterministic.
Let us calculate how successful they were with respect to all generated pairs. Alice's probability to send either $|\chi^3\rangle$ or $|\chi^4\rangle$ successfully, with respect to incoming qubits she tries to send these messages with, is 50% and the probability to fail at the first attempt is also 50%. When she tries to send a message and it passes, she need not repeat it. If it does not pass, Alice has to repeat it. The probability of failing is now $1/2 \cdot 1/2 = 2^{-2}$; and of failing again 2^{-3}, and so on. Thus, all possible ways of failures form a geometric series $\sum_{i=1}^{\infty} 2^{-i} = 1$ and therefore Alice can send $n|\chi^{3,4}\rangle$ messages in $2n$ attempts. We obtain the number of equally distributed $|\chi^{1,2,3,4}\rangle$ messages Alice can send with one hundred given incoming pairs generated from the source as

$$n + n + 2n + 2n = 100 \Longrightarrow n \approx 16.7\,. \tag{1.199}$$

Therefore, $4 \times 16.7 = 66.7\%$ of the incoming pairs can be used to send messages and $1/3$ of them must be discarded. The question emerges – is such a transfer probabilistic or deterministic? It can be rendered deterministic simply because Alice knows what she sent and Bob knows what he received without any ambiguity. How many generated pairs this "costs" is irrelevant for the transfered messages. It could only be considered probabilistic if Alice did not know which message she actually sent as in the scenario (a) above.[29)]

c) Alice makes use of all the photons she is sent.
   i) Alice keeps the "wrong" photons; Bob receives only single photons (his own ones, but altered by Alice's manipulation); he carries out a bit-flip on them to correct them and read off the right message from them;
   ii) Alice clones the "wrong" photons, sends them to Bob, and informs him on that over a classical channel; Bob bit-flips them to set them right.

Bob receives the messages by means of photon detectors which detect photons coming out through the ports 1–4 shown in Figure 1.42. Detectors with photon number resolution can recognize two photons in states $|\chi^{3,4}\rangle$ in one step, hence our discrimination of $|\chi^i\rangle$ is, in principle, deterministic. The coincidence clicks shown in Table 1.5 correspond to a deterministic discrimination of all four $|\chi^i\rangle$ states. "Clicks" for $|\chi^{3,4}\rangle$ mean that the corresponding detectors detected either two photons bunched together or single photons in case c) i).

29) Of course, one can argue that with highly regular repetition of photons on demand we would have gaps in quantum computation, but for such applications we switch to option (c).

Table 1.5 Superdense coding. Ideal discrimination of all four mixed basis states with photon number resolution detectors.

	"Clicks" at		
$\|\chi^1\rangle$	D_1 AND D_3	OR	D_2 AND D_4
$\|\chi^2\rangle$	D_1 AND D_2	OR	D_3 AND D_4
$\|\chi^3\rangle$	D_1	OR	D_4
$\|\chi^4\rangle$	D_2	OR	D_3

However, the highest efficiency of detectors with photon number resolution is currently about 90% [114, 270]. Therefore, we also present a scheme that makes use of single photon detectors, whose highest current efficiency is 99% [173, 213], applied to photons after their passage through a device shown in Figure 1.43. They enter BSs from one side only and therefore behave completely classically; [61, 166] 50% of photons split, that is, emerge from the opposite sides of the BS and 50% emerge bunched together from one of the BS sides. We repeat that n times as shown in Figure 1.43. So, the probability of coincidence detection of split photons by two detectors grows by 50% of the previous growth at each next BS.

The procedure requires $2^n - 1$ beam splitters and 2^n detectors for n steps. The probability of discriminating $|\chi^{3,4}\rangle$ states by detecting photons coming out from the last n-th row of BSs in coincidence is $1-2^{-n}$. For example, for $n = 6$, the probability is 98.4%. The probability of discriminating $|\chi^{1,2}\rangle$ does not depend on n and is always 100% (ideally); the photons in the latter states just have to pass through all BSs. The detection scheme is shown in Table 1.6.

A realistic experiment is trivially feasible with the current technology. As for the losses in the system, they are minimal, since BSs in Figure 1.43 can be the ones with dielectric coating whose losses can be as low as 0.1%.

Let us compare our postselection channel capacity with the best postselection channel capacity of $1.63 > \log_2 3$ [13] achieved so far. In our setup, Alice's postselection does not suffer from any ambiguities except when a simultaneous down-

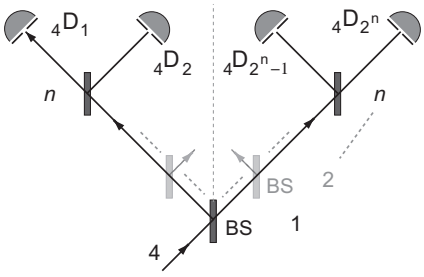


Figure 1.43 Concatenated beam splitter device for splitting $|\chi^{3,4}\rangle$ photons emerging from one of the four ports instead of the detectors shown in Figure 1.42. The figure above is made for port 4. The devices for the other ports are the same.

Table 1.6 Superdense coding. Ideal discrimination of all four mixed basis states with single-photon detectors shown in Figure 1.43 with $n \to \infty$. Expressions in brackets should be dropped for c) i) (single photons). They hold for c) ii).

	Simultaneous "clicks" at		
$\|\chi^1\rangle$	${}_1\mathrm{D}_j$	AND	${}_2\mathrm{D}_k$
$\|\chi^1\rangle$	${}_3\mathrm{D}_j$	AND	${}_4\mathrm{D}_k$
$\|\chi^2\rangle$	${}_1\mathrm{D}_j$	AND	${}_3\mathrm{D}_k$
$\|\chi^2\rangle$	${}_2\mathrm{D}_j$	AND	${}_4\mathrm{D}_k$
$\|\chi^3\rangle$	${}_1\mathrm{D}_j$	(AND	${}_1\mathrm{D}_m$)
$\|\chi^3\rangle$	${}_4\mathrm{D}_j$	(AND	${}_4\mathrm{D}_m$)
$\|\chi^4\rangle$	${}_2\mathrm{D}_j$	(AND	${}_2\mathrm{D}_m$)
$\|\chi^4\rangle$	${}_3\mathrm{D}_j$	(AND	${}_3\mathrm{D}_m$)
	$j, k, m = 1, \ldots, 2^n; m \neq j.$		

conversion of two photon pairs occurs and only one photon from each pair is detected. The probability of such an event is very small and we can almost completely eliminate it by utilizing down-converted photons on demand [16]. Since in a post-selection mode all clicks that correspond to any losses of photons (one or none) at BSs are discarded, our overall efficiency is over 98%. Thus, by using only one BS for each of the four ports ($n = 1$ in Figure 1.43), in this "postselection mode," we can unambiguously transfer $4 \cdot 0.98 = 3.92$ messages via one qubit, that is, our channel capacity is $\log_2 3.92 \approx 1.97$ bit (for a more detailed evaluation see [13]).

Taken together, Alice can carry out a full superdense coding in the mixed basis by manipulating a photon from an entangled pair of photons so as to generate four messages Bob can unambiguously discriminate by a beam splitter and two polarizing beam splitters as shown in Figure 1.42. The four messages are two Bell states $|\Psi^{\pm}\rangle$ and two states, $|00\rangle$ and $|11\rangle$, from the computational basis. Together, they form a *mixed basis* (see (1.198)). We were able to do so because the linear-optics no-go proof by Vaidman and Lütkenhaus does not apply to the mixed basis. We stress here that Alice sends all four messages by a manipulation of her qubit only and that Alice and Bob do *not* have to communicate via a classical channel during the transfer; see otpion c) i). Alice and Bob can unambiguously determine which messages they send and receive, respectively. Scalability of the c) i) scenario is possible in the following way. Upon measuring each of the messages, Bob forwards them as Bell states; measured $\chi^{1,2,3,4}$ he forwards as $|\Psi^-\rangle, |\Psi^+\rangle, |\Phi^-\rangle, |\Phi^+\rangle$, respectively.

1.16 Copying Qubits? No. Teleporting Qubits!

In a standard digital computer, a gate output can be implemented electronically to drive a number of other gates, in other words, to copy data. In a reversible computer, such copying would mean dissipating energy, so it is not allowed. Instead,

we have to do copying by means of a fan-out simulation with the help of a reversible gate as described in Section 1.8 (see Figure 1.19). In a quantum computer, we can "copy" a state of a qubit to another qubit only at the cost of destroying the original state – we say we *teleport* the state.[30] (Below, we prove that we cannot *clone a qubit*.) Teleportation is enabled by a particular kind of correlation between the qubit states called *entanglement* of qubits.

Our inability to copy states in an arbitrary number of copies might seem like a serious limitation imposed on any handling of quantum systems within a quantum computer because we should carry out our computation with just one instance of any state and quantum states are notoriously short lived – they decohere very fast. Entanglement enables us to shuttle single instances of qubit states around and that proves essential for repairing decohered states via quantum error correction, for implementation of existing algorithms, and for communication between qubits. Entanglement also enhances properties of qubits by selecting states to be used for computation. The most basic example is the *superdense coding* we described in Section 1.15.

We cannot take a superposition, also called the *unknown state*,[31]

$$\frac{1}{\sqrt{|\alpha|^2+|\beta|^2}}(\alpha|0\rangle+\beta|1\rangle)\,, \quad \alpha,\beta\neq 0,1\,, \tag{1.200}$$

and a "known state," either $|\psi\rangle = |0\rangle$ or $|\psi\rangle = |1\rangle$, so as to arrive at two replicas of the unknown state:

$$(\alpha|0\rangle+\beta|1\rangle)\otimes|\psi\rangle \nRightarrow \gamma(\alpha|0\rangle+\beta|1\rangle)\otimes(\alpha|0\rangle+\beta|1\rangle)\,, \tag{1.201}$$

where γ is an arbitrary constant.

In general, the following *no-cloning theorem* holds. It is also known as the *no-cloning principle*.

Theorem 52 No-cloning theorem

Unknown quantum states cannot be cloned.

Proof: Let us assume that there exists a cloning operator C. By linearity, we have

$$C(\alpha|0\rangle+\beta|1\rangle) = \alpha C|0\rangle+\beta C|1\rangle\,. \tag{1.202}$$

The left-hand side of (1.202):

$$(\alpha|0\rangle+\beta|1\rangle)\otimes(\alpha|0\rangle+\beta|1\rangle) = \alpha^2|00\rangle+\alpha\beta(|10\rangle+|01\rangle)+\beta^2|11\rangle \tag{1.203}$$

30) Note that we cannot teleport physical carriers of qubits – only their states. We cannot teleport "bodies," as in SCI-FI movies, only states in which the bodies find themselves in.

31) We say "unknown" because when we measure it, we randomly get either a $|0\rangle$-click or a $|1\rangle$-click, although in the long (N) run we, of course, we get $N|\alpha|^2/(|\alpha|^2+|\beta|^2)$ $|0\rangle$-clicks and $N|\beta|^2/(|\alpha|^2+|\beta|^2)$ $|1\rangle$-clicks.

is, however, not equal to its right-hand side

$$\alpha C|0\rangle + \beta C|1\rangle = \alpha|00\rangle + \beta|11\rangle \ , \tag{1.204}$$

for $\alpha, \beta \neq 0$. For $\alpha = 0$ or $\beta = 0$, we can, of course, copy states $|0\rangle$ or $|1\rangle$ since these states are obviously known to us and we can produce an arbitrary number of horizontally or vertically polarized photons, respectively. □

We cannot clone unknown qubits, but we can nevertheless teleport them as we show below. However, we have to pay a price for doing so: the original must be destroyed in the process. To see why, we have to analyze the entanglement from yet another angle, which is essential for understanding its role in teleportation and computation. Let us have a look at the experimental proposal shown in Figure 1.44 [249].

Two independent sources, $S_{\rm I}$ and $S_{\rm II}$, both simultaneously emit two photons correlated in polarization to the left and right. We orient our polarizing beam splitters PBS_1–PBS_4 along angles θ_1–θ_4 with respect to a chosen direction – say of the polarization direction of the pump beam. To point out that we get photons genuinely unprepared, we emphasize that the sources can in principle be atoms exhibiting cascade emission, two independent down-conversion crystals, or two down-conversion crystals pumped by a (split) common beam [235, 249].

The state of the four photons immediately after leaving their triplet-pair sources is described by the product of two entangled states $|\Psi^+\rangle$ (cf. (1.98))

$$|\Psi\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle_1|0\rangle_3 + |1\rangle_1|1\rangle_3\right) \otimes \frac{1}{\sqrt{2}}\left(|0\rangle_2|0\rangle_4 + |1\rangle_2|1\rangle_4\right) \ . \tag{1.205}$$

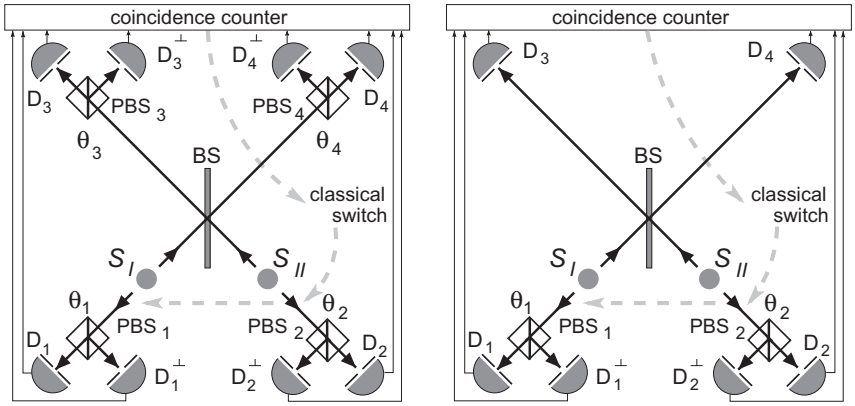


Figure 1.44 Teleportation and entanglement of qubits that never interacted and that come from independent sources. Figure according to Figure 1 from [249]: "Although (photon (1,2)) trajectories never mix or cross (and sources ($S_{\rm I}$, $S_{\rm II}$) are completely independent) they exhibit correlations when the other two photons interfere at a beam splitter (a) even when the latter two do not pass any polarizers at all (b)."

$|0\rangle$ and $|1\rangle$ denote mutually orthogonal photon states. For example, $|0\rangle_1$ means the state of photon 1 leaving source S_I to the left, polarized in direction x. When PBS_1 is oriented at some angle θ_1, its action (filtering) and detection by detector D_1 are represented by the operator $\hat{a}_1 = \hat{a}_{01}\cos\theta_1 + \hat{a}_{11}\sin\theta_1$. The phase that the photon accumulates between the source S_I and the detector D_1 is $e^{i\omega_1(r_1/c + t_0^I - t_1)}$, where ω_1 is the frequency of photon 1, r_1 is the path length from S_I to D_1, c is the speed of light, t_0^I is the time of emission of a pair of photons at S_I, and t_1 is the time of detection at D_1.

To describe the detection of a photon by detector D_1, we apply the operator

$$\hat{E}_1 = (\hat{a}_{01}\cos\theta_1 + \hat{a}_{11}\sin\theta\,1)e^{i\omega_1\left(r_1/c + t_0^I - t_1\right)} \tag{1.206}$$

to the initial state of (1.205). Similarly, the detection of photon 2 at D_2 means we apply

$$\hat{E}_2 = (\hat{a}_{02}\cos\theta_2 + \hat{a}_{12}\sin\theta_2)e^{i\omega_2\left(r_2/c + t_0^{II} - t_2\right)}\ , \tag{1.207}$$

where the symbols are defined analogously.

A detection at D_3 can be caused by the emission of photon 3 by source S_I or the emission of photon 4 by source S_II. The beam splitter BS may have polarization transmittances and reflectances, denoted by T_0, T_1, and R_0, R_1, respectively. The angle of PBS_3 is given by θ_3. The operator describing a photon arriving at the detector D_3 is

$$\begin{aligned}\hat{E}_3 &= \left(\hat{a}_{04}\sqrt{T_0}\cos\theta_3 + \hat{a}_{14}\sqrt{T_1}\sin\theta_3\right)e^{i\alpha} \\ &\quad + i\left(\hat{a}_{03}\sqrt{R_0}\cos\theta_3 + \hat{a}_{13}\sqrt{R_1}\sin\theta_3\right)e^{i\beta} \\ &= (\hat{a}_{04}A_{04} + \hat{a}_{14}A_{14})e^{i\alpha} + i(\hat{a}_{03}A_{03} + \hat{a}_{13}A_{13})e^{i\beta}\ ,\end{aligned} \tag{1.208}$$

where

$$\alpha = \omega_4\left(\frac{r_\mathrm{II} + r_3}{c} + t_0^\mathrm{II} - t_3\right)\ , \quad \beta = \omega_3\left(\frac{r_\mathrm{I} + r_3}{c} + t_0^\mathrm{I} - t_3\right)\ , \tag{1.209}$$

where r_I and r_II denote the distances from the respective sources to BS, and r_3 and t_3 denote the distance from BS to D_3 and the time of detection at D_3, respectively. The meaning of the A's is obvious. For D_4, one defines $\hat{E}_4$ analogously [249]:

$$\begin{aligned}\hat{E}_4 &= \left(\hat{a}_{03}\sqrt{T_0}\cos\theta_4 + \hat{a}_{13}\sqrt{T_1}\sin\theta_4\right)e^{i\gamma} \\ &\quad + i\left(\hat{a}_{04}\sqrt{R_0}\cos\theta_4 + \hat{a}_{14}\sqrt{R_1}\sin\theta_4\right)e^{i\delta} \\ &= (\hat{a}_{03}B_{03} + \hat{a}_{13}B_{13})e^{i\gamma} + i(\hat{a}_{04}B_{04} + \hat{a}_{14}B_{14})e^{i\delta}\ ,\end{aligned} \tag{1.210}$$

where

$$\gamma = \omega_3\left(\frac{r_\mathrm{I} + r_4}{c} + t_0^\mathrm{I} - t_4\right)\ , \quad \delta = \omega_4\left(\frac{r_\mathrm{II} + r_4}{c} + t_0^\mathrm{II} - t_4\right)\ . \tag{1.211}$$

The meaning of the B's is obvious.

Until this point in the calculation, there is no entanglement. For as long as we keep the "wholeness" – as Niels Bohr would say – of the experimental arrangement, a corresponding "complete" wave function of several quantum systems taking part in the experiment can always be described by a tensor product of its parts. Entanglement comes to the stage when we want to make some measurements on some subsystems and do not want to make some other measurements on some other subsystems, that is, when we decide to manipulate our subsystems with the aim of constructing a new quantum reality. Then – as a consequence – we also manipulate the formalism so as to extract the parts we need and disregard those that we do not need.

To arrive at this extraction from the standard formalism, we consider the experiment presented in Figure 1.44. We consider the coincidence probability for all four photons detected by detectors D_1–D_4. (We shall consider detectors $D_1^\perp$–$D_4^\perp$ later on.) The probability reads [249]

$$P(\theta_1, \theta_2, \theta_3, \theta_4) = \langle \Psi | \hat{E}_1^\dagger \hat{E}_2^\dagger \hat{E}_3^\dagger \hat{E}_4^\dagger \hat{E}_4 \hat{E}_3 \hat{E}_2 \hat{E}_1 | \Psi \rangle \ . \tag{1.212}$$

Equations (1.205)–(1.207) yield

$$\hat{E}_2 \hat{E}_1 | \Psi \rangle = \frac{e^{i\epsilon}}{2} (\cos\theta_1 |0\rangle_3 + \sin\theta_1 |1\rangle_3) \otimes (\cos\theta_2 |0\rangle_4 + \sin\theta_2 |1\rangle_4) \ , \tag{1.213}$$

where the meaning of ϵ is obvious.

By applying $\hat{E}_4 \hat{E}_3$ as given by (1.208) and (1.210) to (1.213) and using rules for annihilation operators given in Section 1.13, we obtain

$$\hat{E}_4 \hat{E}_3 \hat{E}_2 \hat{E}_1 | \Psi \rangle = \frac{1}{2} \left[Q_{114} Q_{123} e^{i(\alpha+\gamma)} - Q_{224} Q_{213} e^{i(\beta+\delta)} \right] e^{i\epsilon} |\varnothing\rangle \ , \tag{1.214}$$

where $|\varnothing\rangle$ is the detection vacuum state and where

$$Q_{ijk} = \sqrt{Q_{i0}} \cos\theta_j \cos\theta_k + \sqrt{Q_{i1}} \sin\theta_j \sin\theta_k \ , \quad i, j, k = 1, 2 \ , \tag{1.215}$$

where Q_{10}, Q_{11}, Q_{10}, Q_{11} are T_0, T_1, R_0, R_1, respectively. Equation (1.212) yields

$$\begin{aligned} P(\theta_1, \theta_2, \theta_3, \theta_4) = \frac{1}{4} \big[& (Q_{114} Q_{123})^2 + (Q_{224} Q_{213})^2 \\ & - 2\cos(\alpha + \gamma - \beta - \delta) Q_{114} Q_{123} Q_{224} Q_{213} \big] \ . \end{aligned} \tag{1.216}$$

Assuming $r_I = r_{II}$, $r_3 = r_4$, $\omega_3 = \omega_4$, and $T_0 = T_1 = R_0 = R_1 = 1/2$, we get the coincidence probability

$$P(\theta_1, \theta_2, \theta_3, \theta_4) = \frac{1}{16} \sin^2(\theta_1 - \theta_2) \sin^2(\theta_3 - \theta_4) \ . \tag{1.217}$$

When no polarization is measured on photons 3 and 4, we get (see Figure 1.44b)

$$P(\theta_1, \theta_2, \infty, \infty) = \frac{1}{8}\sin^2(\theta_1 - \theta_2)\ . \tag{1.218}$$

The probability given by (1.218) and describing coincidence detections by D_1 and D_2 corresponds – when multiplied by four – to the *singlet state*:

$$|\Psi^-\rangle = \frac{1}{\sqrt{2}}(|0\rangle_1|1\rangle_2 - |1\rangle_1|0\rangle_2)\ . \tag{1.219}$$

Multiplication by four is for photons that emerge from the same side of BS and which we therefore dropped from our statistics [235].

This is exactly what we call the polarization *entanglement* of photons 1 and 2, which did not in any way directly interact and on distant pairs of which polarization has not been measured at all ... [and whose] trajectories never mix or cross [249].

The result has been verified experimentally [224].

The meaning of this statement for the states of photons 1 and 2 in the experiment shown in Figure 1.44 is the following. Let us remove PBS_3, PBS_4, $D_3^{\perp}$, $D_4^{\perp}$. Then, simultaneous clicks at D_3 and D_4 in either (a) or (b) open a switch which lets photons 1 and 2 to fibers or to space (antennas, satellites, ...) and these two photons will remain entangled indefinitely. The switch will not allow other photons 1 and 2, for which only single detectors either D_3, or $D_3^{\perp}$ (a), or D_4, or $D_4^{\perp}$ (a), have been triggered by both of them, to leave the system. We can say that ideally we prepare an entangled pair of qubits at a definite moment in time. In a realistic preparation, we are in control of all the qubits only with a definite probability, but the probability is the main player in any quantum game and we are improving our skill of dealing with it.

When we look at the probability $\sin^2(\theta_1 - \theta_2)$ in (1.218), we see that for parallel alignment of PBS_1 and PBS_2, we will not get any detection by D_1 and D_2. For such orientation of PBS_1 and PBS_2, we collect the data from D_1 and $D_2^{\perp}$ for which the probability of coincidental detection is given by

$$P(\theta_1, \theta_2^{\perp}, \infty, \infty) = \frac{1}{8}\cos^2(\theta_1 - \theta_2)\ . \tag{1.220}$$

As for detections of D_3–$D_4^{\perp}$, note that while for measurements corresponding to (1.219) we do have entanglement, for other measurements in the considered setup, we do not have entanglement. For example, the overall probability of detecting both photons 3, 4 in one arm of BS and detecting photons 1, 2 by D_1 and D_2 is

given by

$$P(\theta_1,\theta_2,\theta_3\times\theta_4) = \langle\Psi|\hat{E}_1^\dagger\hat{E}_2^\dagger\hat{E}_3^\dagger\hat{E}_3^\dagger\hat{E}_3\hat{E}_3\hat{E}_2\hat{E}_1|\Psi\rangle$$
$$= \frac{1}{16}\big[\cos(\theta_1-\theta_3)\cos(\theta_2-\theta_3)+\cos(\theta_1-\theta_4)\cos(\theta_2-\theta_4)\big]^2 , \qquad (1.221)$$

which for removed PBS_3 and PBS_4, reads

$$P(\theta_1,\theta_2,\infty\times\infty) = \frac{1}{8}\left[1+\cos^2(\theta_1-\theta_2)\right] . \qquad (1.222)$$

We can also see that by removing one of the PBS_1 and PBS_2, say PBS_2, we lose any left–right (Bell-like) spin correlation completely: $P(\theta_1,\infty,\theta_3,\theta_4) = 1/8\sin^2(\theta_3-\theta_4)$, $P(\theta_1,\infty,\infty,\infty) = 1/4$, $P(\theta_1,\infty,\infty\times\infty) = 1/4$ [235]. Hence, the entanglement is just a property of some subsystems of the whole composite system under a particular measurement arrangement.

If we substitute the following product of singlet states

$$|\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle_1|1\rangle_3-|1\rangle_1|0\rangle_3)\otimes\frac{1}{\sqrt{2}}(|0\rangle_2|1\rangle_4-|1\rangle_2|0\rangle_4) \qquad (1.223)$$

for the triplet sources given by (1.205), we get exactly the same entanglement as above, that is, the state (1.219) and the probability (1.218). This outcome reveals an entanglement of photons from such pairs as almost synonymous with *teleportation*.

To see this, let us look at source S1 in Figure 1.45. (Sources S1 and S2 are simultaneously triggered by a common pumping laser beam.) Photons coming out of source S1 are in the singlet state, and therefore their polarizations are completely unprepared but correlated. With such an unprepared polarization, one of the photons from source S1 (photon 2) arrives at beam splitter BS, interferes there with another photon coming from source S2, loses its polarization and *teleports* that polarization to the second photon from source S2, that is, to photon 4. What does this mean? It means that by measuring the polarization of photon 4, we recover the polarization of the photon coming from source S1 to beam splitter BS. How do we

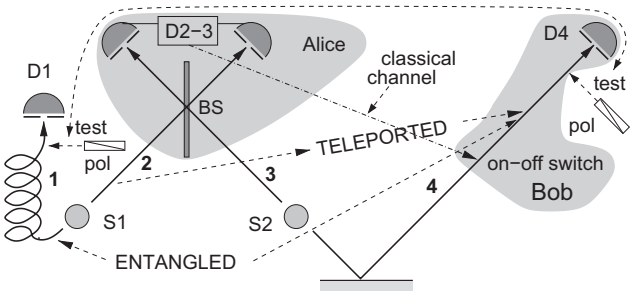


Figure 1.45 Teleportation, entanglement, and correlation. "Two photons from two singlets interfere at a beam splitter, and as a result, the other two photons – which nowhere interacted and whose paths nowhere crossed – exhibit a 100% correlation in polarization, even when no polarization has been measured in the first two photons." [235].

know this? By measuring polarization of photon 1 by detector D_1. (Since the photons coming out of source S1 are in the singlet state, measuring the polarization of photon 1 reveals the polarization the photon *could* have had. The same holds for the triplet states.) This outcome has also been verified experimentally [37]. The experiment actually confirms (1.218). So, both entanglement and teleportation are about engineering particular subsystems with particular properties corresponding to just some parts of a complete mathematical description of the whole system.

An important thing in the above design of teleportation is that any photon from an entangled pair is by itself completely unpolarized. For instance, photon 2 in Figure 1.45, when we do not consider its state correlations with the other photon from the pair 1. It is unpolarized until we decide to measure the polarization of its companion 2. That collapses the states of photon 1 and therefore also photon 4 into the state of perpendicular polarization. Before that, we can say that state of photon 2 is *unknown* to us and that we teleported that *unknown state* to photon 4.

In quantum computation, the notion of unknown state is very important because we must not collapse states of our qubits by any intermediate measurements until the very end of our computation. Actually, if we send a photon in an unknown state

$$|\psi\rangle = \frac{1}{\sqrt{\alpha^2 + \beta^2}}(\alpha|0\rangle + \beta|1\rangle) \tag{1.224}$$

as photon 2 so that it is not entangled with any photon 1, it will project this state to photon 4, which originally belonged to the state $|\Psi^-\rangle$ emitted from S2, whenever a D_1–D_2 coincidence is detected as shown in Figure 1.46. Filtering out a teleported state is a special case of the complete deterministic teleportation procedure explained below.

And while this opens the road towards engineering new quantum states that we can use for quantum computation and communication, the above selection of the states which we can use for teleportation also limit this usage. For, we have to discard all transmitted data for which we have not detected a coincidence by D_1–D_2 in Figure 1.45. If we wanted a complete and deterministic teleportation with only one degree of freedom – polarization – we should make use of the complete set of Bell states and therefore – due to Lütkenhaus' and Vaidman's result – also of

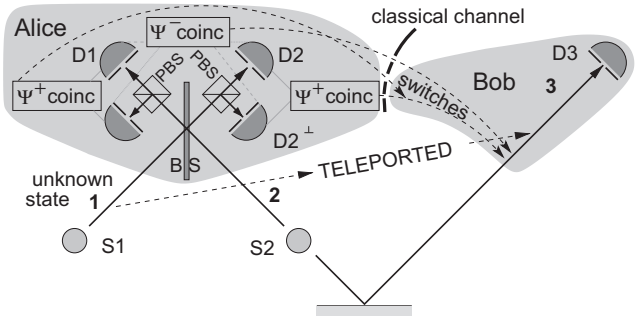


Figure 1.46 Probabilistic teleportation.

nonlinear optics. Let us stress here that the mixed basis (Definition 51, (1.198)) cannot be used for the purpose because $|\chi^3\rangle$ and $|\chi^4\rangle$ are not entangled states.

For the latter purpose several rather different designs have been proposed and some of them have been experimentally verified recently [12, 79, 143, 144, 227, 280, 313]. Up to now, none of them reached the efficiency of linear optics probabilistic teleportation with two Bell states ($|\Psi^{\pm}\rangle$) so that there is no obvious technological candidate for a future application. Therefore, we shall present a design which is conceptually intriguing and yet easy to follow, although it might not be the one which is the easiest to carry out in the laboratory. This is a design based on the so-called Kerr medium and proposed by Vitali, Fortunato, and Tombesi [313].

The theoretical background of all teleportation designs that measure all four Bell states – we call such designs *complete deterministic teleportation* – is as follows.

We let a qubit in an unknown state (1.224) interact with a second qubit initially entangled with a third qubit in an arbitrary Bell state (one of four). In Figure 1.47, we let photon 1 in an unknown state (1.224) interact – in a Kerr medium – with photon 2 which was entangled with photon 3 in a Bell state originated from S. S is a source of photon pairs entangled in one of four Bell states ($|\Psi^{\pm}\rangle$, $|\Phi^{\pm}\rangle$). We choose $|\Psi^{+}\rangle$ for our calculation below. We describe the input states $|\psi\rangle_1$ and $|\Psi^{+}\rangle_{23}$ which enter the Bell analyzer by their tensor product.

A Kerr medium is a medium (usually a crystal) that enables a nonlinear interaction of light with an instantaneous response, related to the nonlinear electronic polarization generated in the medium, which itself modifies the propagation of the light by changing the refractive index. In the considered experiment, photons 1 and 2 change each other's polarization.

In the following tensor product, the order of kets refers to a corresponding qubit and therefore we keep it fixed so as to be able to drop indices that refer to qubits and thus ease the notation. For instance, $|000\rangle$ will mean $|0\rangle_1|0\rangle_2|0\rangle_3 = |0\rangle_1 \otimes |0\rangle_2 \otimes |0\rangle_3$ and $|0\rangle \otimes |00\rangle$ will also mean $|0\rangle_1 \otimes |0\rangle_2|0\rangle_3 = |0\rangle_1 \otimes (|0\rangle_2 \otimes |0\rangle_3) = |0\rangle_1 \otimes |0\rangle_2 \otimes |0\rangle_3$.

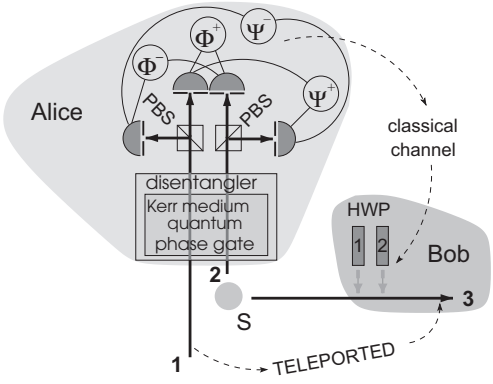


Figure 1.47 Complete deterministic teleportation. A figure according to Figure 1 from [313].

Thus, the input product state is

$$|\psi\rangle_1 \otimes |\Psi^+\rangle_{23} = (\alpha|0\rangle + \beta|1\rangle)\frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$$
$$= \frac{1}{\sqrt{2}}(\alpha|001\rangle + \beta|101\rangle + \alpha|010\rangle + \beta|110\rangle). \tag{1.225}$$

Expressing the left positioned $|00\rangle, \ldots |11,\rangle = |0\rangle_1 0\rangle_2, \ldots, |1\rangle_1 1\rangle_2$ by means of $|\Psi^+\rangle_{12}, \ldots, |\Phi^-\rangle_{12}$, from (1.125) we obtain

$$\begin{aligned}
&\frac{1}{2}[\alpha(|\Phi^+\rangle + |\Phi^-\rangle)|1\rangle + \beta(|\Psi^+\rangle - |\Psi^-\rangle)|1\rangle + \alpha(|\Psi^+\rangle + |\Psi^-\rangle)|0\rangle \\
&\quad + \beta(|\Phi^+\rangle - |\Phi^-\rangle)|0\rangle] \\
&= \frac{1}{2}[|\Psi^+\rangle(\alpha|0\rangle + \beta|1\rangle) + |\Psi^-\rangle(\alpha|0\rangle - \beta|1\rangle) \\
&\quad + |\Phi^+\rangle(\beta|0\rangle + \alpha|1\rangle) + |\Phi^-\rangle(\beta|0\rangle - \alpha|1\rangle)] \\
&= \frac{1}{2}\left[|\Psi^+\rangle_{12}\begin{pmatrix}\alpha\\ \beta\end{pmatrix}_3 + |\Psi^-\rangle_{12}\begin{pmatrix}\alpha\\ -\beta\end{pmatrix}_3\right. \\
&\quad \left. + |\Phi^+\rangle_{12}\begin{pmatrix}\beta\\ \alpha\end{pmatrix}_3 + |\Phi^-\rangle_{12}\begin{pmatrix}-\beta\\ \alpha\end{pmatrix}_3\right].
\end{aligned} \tag{1.226}$$

When photons 1 and 2 are measured to be in one of the Bell states $|\Psi^+\rangle_{12}, \ldots, |\Phi^-\rangle_{12}$ that means that photon 3 is in one of the states described by one-column matrices to the right of the corresponding Bell states in (1.226), respectively.

In the design shown in Figure 1.47, photons 1 and 2 exhibit a cross-Kerr effect on each other in the Kerr medium which serves as a quantum phase gate and with the help of three polarization rotators before and after the gate they disentangle and each of their Bell states is transformed to the following polarization states, respectively, that is,

$$\begin{aligned}
&|\Psi^+\rangle \longrightarrow |01\rangle\,, \quad |\Phi^+\rangle \longrightarrow |00\rangle\,, \\
&|\Psi^-\rangle \longrightarrow |11\rangle\,, \quad |\Phi^-\rangle \longrightarrow |10\rangle\,.
\end{aligned} \tag{1.227}$$

This is graphically shown in Figure 1.47. Thus, four coincidental detection of four pairs of entangled polarized photons each representing one of the four Bell states can switch (e.g., by means of a Pockels cell) the beam 3 to a path containing none, one, or two HWPs and transform its state into the original state of photon 1. (See (1.186) and (1.187).)

Formally, this can be done as follows

$$
\begin{aligned}
|\Psi^+\rangle: \quad & |\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{pmatrix}\alpha\\ \beta\end{pmatrix} = |\psi\rangle\ ,\\
|\Psi^-\rangle: \quad & \mathrm{HWP}(0)(\alpha|0\rangle - \beta|1\rangle)\\
& = \sigma_z \begin{pmatrix}\alpha\\ -\beta\end{pmatrix} = \begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}\begin{pmatrix}\alpha\\ -\beta\end{pmatrix} = \begin{pmatrix}\alpha\\ \beta\end{pmatrix} = |\psi\rangle\ ,\\
|\Phi^+\rangle: \quad & \mathrm{HWP}\left(\frac{\pi}{4}\right)(\beta|0\rangle + \alpha|1\rangle)\\
& = \sigma_x \begin{pmatrix}\beta\\ \alpha\end{pmatrix} = \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}\begin{pmatrix}\beta\\ \alpha\end{pmatrix} = \begin{pmatrix}\alpha\\ \beta\end{pmatrix} = |\psi\rangle\ ,\\
|\Phi^-\rangle: \quad & \mathrm{HWP}(0)\mathrm{HWP}\left(\frac{\pi}{4}\right)(-\beta|0\rangle + \alpha|1\rangle)\\
& = \sigma_z\sigma_x \begin{pmatrix}-\beta\\ \alpha\end{pmatrix} = \begin{pmatrix}\alpha\\ \beta\end{pmatrix} = |\psi\rangle\ .
\end{aligned}
\tag{1.228}
$$

Thus, Ψ^+-click will cause no HWP action – photon 3 is already in the original state $|\psi\rangle$; Ψ^--click will make photon 3 pass HWP(0) (HWP oriented along the horizontal polarization direction); Φ^+-click will cause redirect photon 3 through HWP($\pi/4$) (HWP oriented along angle $\pi/4$ with respect to the horizontal polarization direction); Φ^--click will cause its passing through both HWP($\pi/4$) and HWP(0).

The above teleportation setup has a serious drawback for the time being – its realistic implementation (with an acceptable efficiency) is beyond current technology. Therefore, we also present a setup which is probabilistic, but approaches a deterministic one to an arbitrary precision and could be implemented with today's technology.

As we pointed out above, Vaidman's [309] and Lütkenhaus' [184] *no-go proofs* state that we can reach 100% efficiency of discriminating all four Bell states only "in a limit." If we exclude conditional dynamics and additional photons ("ancillas"), then only 50% can be reached, that is, only two of four Bell states can be unambiguously discriminated [60]. Under "conditional dynamics," "we mean that we monitor one selected mode while keeping the other modes in a waiting loop... Then, we can perform some linear operation on the remaining modes." [184] Conditional dynamics have been used in the above Kerr medium teleportation implementation, but the Kerr medium is nonlinear and a deterministic implementation is allowed.

What we need for a feasible implementation is a linear optics one and that is what we present below following W.P. Grice [109]. It can be considered as a special and feasible example of the so-called Knill–Laflamme–Milburn's approach to linear optics implementations [147]. Knill–Laflamme–Milburn's approach consists of making use of additional photons – the so-called *ancilla photons* or simply *ancillas* – and conditional dynamics using only linear optics elements.

In Grice's implementation, we simply combine the incoming Bell states with the n ancilla photon pairs we prepare in one of the Bell states, say $|\Phi^+\rangle$ as below, on several beam splitters. The clicks of the detectors behind final beam splitters will

discriminate the incoming in a near-deterministic way. The higher the number of ancillas n, the more we approach a deterministic (100%) discrimination (ideally).

The schematic of our implementation is given in Figure 1.48. Inverse creation operator relations are given by (1.160). Hence, for BS_{13}, we have (we omit $\hat{a}^\dagger_{3x}, \hat{a}^\dagger_{1y}$ for brevity; they can easily be written down using $\hat{a}^\dagger_{1x}, \hat{a}^\dagger_{3y}$ and (1.160))

$$\hat{a}^\dagger_{1x} = \frac{1}{\sqrt{2}}\left(\hat{b}^\dagger_{1x} - \hat{b}^\dagger_{3x}\right), \quad \ldots, \quad \hat{a}^\dagger_{3y} = \frac{1}{\sqrt{2}}\left(\hat{b}^\dagger_{1y} + \hat{b}^\dagger_{3y}\right), \tag{1.229}$$

for BS_{24} (again, we omit two terms):

$$\hat{a}^\dagger_{4x} = \frac{1}{\sqrt{2}}\left(\hat{b}^\dagger_{4x} - \hat{b}^\dagger_{2x}\right), \quad \ldots, \quad \hat{a}^\dagger_{2y} = \frac{1}{\sqrt{2}}\left(\hat{b}^\dagger_{4y} + \hat{b}^\dagger_{2y}\right), \tag{1.230}$$

for BS_{12}:

$$\hat{b}^\dagger_{1x} = \frac{1}{\sqrt{2}}\left(\hat{c}^\dagger_{1x} + \hat{c}^\dagger_{2x}\right), \quad \ldots, \quad \hat{b}^\dagger_{2y} = \frac{1}{\sqrt{2}}\left(\hat{c}^\dagger_{2y} - \hat{c}^\dagger_{1y}\right), \tag{1.231}$$

and for BS_{34}:

$$\hat{b}^\dagger_{3x} = \frac{1}{\sqrt{2}}\left(\hat{c}^\dagger_{3x} - \hat{c}^\dagger_{4x}\right), \quad \ldots, \quad \hat{b}^\dagger_{4y} = \frac{1}{\sqrt{2}}\left(\hat{c}^\dagger_{3y} + \hat{c}^\dagger_{4y}\right). \tag{1.232}$$

Photons in Bell states should arrive at BS_{13} and BS_{24} through ports 1 and 2 simultaneously. We describe their total state by tensor products

$$|\Psi^\pm\rangle_{12} \otimes |\Phi^+\rangle_{34}, \quad |\Phi^\pm\rangle_{12} \otimes |\Phi^+\rangle_{34}. \tag{1.233}$$

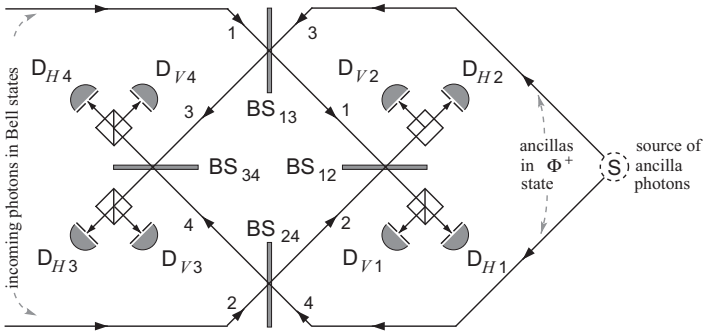


Figure 1.48 Near-deterministic discrimination of all four Bell states with linear optics. Schematic of a setup with one ancilla pair which serves to discriminate between all four detectors clicks $|\Psi^\pm\rangle$ with the probability of 100%. They also discriminate $|\Phi^\pm\rangle$ from $|\Psi^\pm\rangle$ in half of the recording. We obtain a total efficiency of 75% for an unambiguous discrimination of Bell states.

In the creation operator notation, the states given by (1.233) yield

$$\begin{aligned}
&\frac{1}{2}\left(a_{3x}^{\dagger}a_{4x}^{\dagger}+a_{3y}^{\dagger}a_{4y}^{\dagger}\right)\left(a_{1x}^{\dagger}a_{2y}^{\dagger}+a_{1y}^{\dagger}a_{2x}^{\dagger}\right)\\
&=-\frac{1}{8}\left(c_{1x}^{\dagger 2}-c_{2x}^{\dagger 2}+2c_{1x}^{\dagger}c_{3x}^{\dagger}+c_{3x}^{\dagger 2}+2c_{2x}^{\dagger}c_{4x}^{\dagger}-c_{4x}^{\dagger 2}+c_{1y}^{\dagger 2}-c_{2y}^{\dagger 2}\right.\\
&\quad\left.+\,2c_{1y}^{\dagger}c_{3y}^{\dagger}+c_{3y}^{\dagger 2}+2c_{2y}^{\dagger}c_{4y}^{\dagger}-c_{4y}^{\dagger 2}\right)\\
&\quad\times\left[c_{1x}^{\dagger}\left(c_{1y}^{\dagger}-c_{3y}^{\dagger}\right)+c_{3x}^{\dagger}\left(-c_{1y}^{\dagger}+c_{3y}^{\dagger}\right)-\left(c_{2x}^{\dagger}+c_{4x}^{\dagger}\right)\left(c_{2y}^{\dagger}+c_{4y}^{\dagger}\right)\right],\\
&\frac{1}{\sqrt{2}}\left(a_{3x}^{\dagger}a_{4x}^{\dagger}+a_{3y}^{\dagger}a_{4y}^{\dagger}\right)\frac{1}{\sqrt{2}}\left(a_{1x}^{\dagger}a_{2y}^{\dagger}-a_{1y}^{\dagger}a_{2x}^{\dagger}\right)\\
&=\frac{1}{8}\left(c_{1x}^{\dagger 2}-c_{2x}^{\dagger 2}+2c_{1x}^{\dagger}c_{3x}^{\dagger}+c_{3x}^{\dagger 2}+2c_{2x}^{\dagger}c_{4x}^{\dagger}-c_{4x}^{\dagger 2}+c_{1y}^{\dagger 2}-c_{2y}^{\dagger 2}\right.\\
&\quad\left.+\,2c_{1y}^{\dagger}c_{3y}^{\dagger}+c_{3y}^{\dagger 2}+2c_{2y}^{\dagger}c_{4y}^{\dagger}-c_{4y}^{\dagger 2}\right)\\
&\quad\left[c_{2x}^{\dagger}\left(-c_{1y}^{\dagger}+c_{3y}^{\dagger}\right)+c_{4x}^{\dagger}\left(-c_{1y}^{\dagger}+c_{3y}^{\dagger}\right)+\left(c_{1x}^{\dagger}-c_{3x}^{\dagger}\right)\left(c_{2y}^{\dagger}+c_{4y}^{\dagger}\right)\right],
\end{aligned}\tag{1.234}$$

$$\begin{aligned}
&\frac{1}{2}\left(a_{3x}^{\dagger}a_{4x}^{\dagger}+a_{3y}^{\dagger}a_{4y}^{\dagger}\right)\left(a_{1x}^{\dagger}a_{2x}^{\dagger}+a_{1y}^{\dagger}a_{2y}^{\dagger}\right)\\
&=-\frac{1}{16}\left(c_{1x}^{\dagger 2}-c_{2x}^{\dagger 2}-2c_{1x}^{\dagger}c_{3x}^{\dagger}+c_{3x}^{\dagger 2}-2c_{2x}^{\dagger}c_{4x}^{\dagger}-c_{4x}^{\dagger 2}+c_{1y}^{\dagger 2}-c_{2y}^{\dagger 2}\right.\\
&\quad\left.-\,2c_{1y}^{\dagger}c_{3y}^{\dagger}+c_{3y}^{\dagger 2}-2c_{2y}^{\dagger}c_{4y}^{\dagger}-c_{4y}^{\dagger 2}\right)\left(c_{1x}^{\dagger 2}-c_{2x}^{\dagger 2}+2c_{1x}^{\dagger}c_{3x}^{\dagger}+c_{3x}^{\dagger 2}\right.\\
&\quad\left.+\,2c_{2x}^{\dagger}c_{4x}^{\dagger}-c_{4x}^{\dagger 2}+c_{1y}^{\dagger 2}-c_{2y}^{\dagger 2}+2c_{1y}^{\dagger}c_{3y}^{\dagger}+c_{3y}^{\dagger 2}+2c_{2y}^{\dagger}c_{4y}^{\dagger}-c_{4y}^{\dagger 2}\right),\\
&\frac{1}{2}\left(a_{3x}^{\dagger}a_{4x}^{\dagger}+a_{3y}^{\dagger}a_{4y}^{\dagger}\right)\left(a_{1x}^{\dagger}a_{2x}^{\dagger}-a_{1y}^{\dagger}a_{2y}^{\dagger}\right)\\
&=\frac{1}{16}\left(c_{1x}^{\dagger 2}-c_{2x}^{\dagger 2}+2c_{1x}^{\dagger}c_{3x}^{\dagger}+c_{3x}^{\dagger 2}+2c_{2x}^{\dagger}c_{4x}^{\dagger}-c_{4x}^{\dagger 2}+c_{1y}^{\dagger 2}-c_{2y}^{\dagger 2}\right.\\
&\quad\left.+\,2c_{1y}^{\dagger}c_{3y}^{\dagger}+c_{3y}^{\dagger 2}+2c_{2y}^{\dagger}c_{4y}^{\dagger}-c_{4y}^{\dagger 2}\right)\left(c_{1x}^{\dagger 2}-c_{2x}^{\dagger 2}-2c_{1x}^{\dagger}c_{3x}^{\dagger}+c_{3x}^{\dagger 2}\right.\\
&\quad\left.-\,2c_{2x}^{\dagger}c_{4x}^{\dagger}-c_{4x}^{\dagger 2}-c_{1y}^{\dagger 2}+c_{2y}^{\dagger 2}+2c_{1y}^{\dagger}c_{3y}^{\dagger}-c_{3y}^{\dagger 2}+2c_{2y}^{\dagger}c_{4y}^{\dagger}+c_{4y}^{\dagger 2}\right).
\end{aligned}\tag{1.235}$$

We can now use a computer program which verifies all possible clicks triggered by photons in one Bell state vs. those in some other Bell state. For instance, detectors D_{1H}, D_{3V}, D_{2x}, and D_{4x} can only be simultaneously triggered by photons in state $|\Psi^{+}\rangle$ since only its creations operators (see (1.234)) contain $c_{1x}^{\dagger}c_{3y}^{\dagger}c_{2x}^{\dagger}c_{4x}^{\dagger}$. Therefore, a multiplication of the other three equations by $c_{1x}c_{3y}c_{2x}c_{4x}$ would give zero. A multiplication of the operators that correspond to $|\Psi^{+}\rangle$ by the latter product gives 1/8 probability of detecting the states when only these detectors fire. A thorough checking of all possible detections shows that the obtained clicks will unambiguously discriminate $|\Psi^{\pm}\rangle$ from each other and from $|\Phi^{\pm}\rangle$ with the probability of 100% and $|\Phi^{\pm}\rangle$ from each other and from $|\Psi^{\pm}\rangle$ with the probability of 50%. This gives us a total probability of 75% for unambiguous discrimination. If we add a flipped-coin-like "guessing" for the remaining ambiguous $|\Phi^{\pm}\rangle$-clicks,

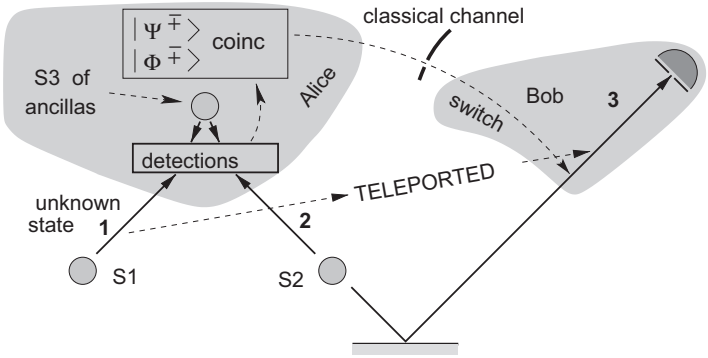


Figure 1.49 Near-deterministic teleportation with linear optics in Knill–Laflamme–Milburn's approach. The detection block represents the device for all-optical discrimination coming from a device shown in Figure 1.48. An arbitrary high number of ancillas and therefore a discrimination of Bell states arbitrary close to 100% is assumed.

we get the total efficiency of 87.5%. With an additional ancilla pair, we would get an unambiguous discrimination with a probability of 87.5% and a total efficiency of 93.75%.

Among all possible triggering of our detectors, there are twofold and fourfold ones, for example, $c_{1x}^{\dagger 4}$. In order to record them via two and four "clicks," respectively, we can use photon number resolution detectors.

However, photon number resolution detectors are of a considerably lower efficiency than single photon detectors. So, we can alternatively make use of concatenated beam splitters BS_i, $i = 1, \ldots, n$ as the ones shown in Figure 1.43. Photons enter BSs from one side only. For example, for horizontal polarization, we obtain

$$\hat{f}_{1x}^{\dagger}\hat{f}_{1x}^{\dagger} = \hat{g}_{1x}^{\dagger}\hat{g}_{2x}^{\dagger} + \frac{1}{2}\left(\hat{g}_{1x}^{\dagger}\hat{g}_{1x}^{\dagger} + \hat{g}_{1y}^{\dagger}\hat{g}_{1y}^{\dagger}\right) . \tag{1.236}$$

For horizontal and vertical polarizations, that is, for $|\Phi^{+}\rangle$, we obtain

$$\begin{aligned}\hat{f}_{1x}^{\dagger}\hat{f}_{1x}^{\dagger} + \hat{f}_{1y}^{\dagger}\hat{f}_{1y}^{\dagger} &= \hat{g}_{1x}^{\dagger}\hat{g}_{2x}^{\dagger} - \hat{g}_{1y}^{\dagger}\hat{g}_{2y}^{\dagger} \\ &+ \frac{1}{2}\left(\hat{g}_{1x}^{\dagger}\hat{g}_{1x}^{\dagger} + \hat{g}_{1y}^{\dagger}\hat{g}_{1y}^{\dagger}\right) + \frac{1}{2}\left(\hat{d}_{2x}^{\dagger}\hat{d}_{2x}^{\dagger} + \hat{d}_{2y}^{\dagger}\hat{d}_{2y}^{\dagger}\right) .\end{aligned} \tag{1.237}$$

So, 50% of photons emerge from the opposite sides of the BS [61, 166]. We repeat that n times as shown in Figure 1.43.

The procedure requires $2^n - 1$ beam splitters and 2^n detectors for n steps. The probability of discriminating $|\Phi^{+}\rangle$ by detecting photons coming out from the last n row of BSs in coincidence is $1 - 2^{-n}$. For example, for $n = 6$, the probability is 98.4%.

Hence, we obtain a near-deterministic implementation of teleportation shown in Figure 1.49.

Here, we should mention that we cannot substitute hyperentangled states of photons for the Knill–Laflamme–Milburn ancillas in the above approach simply

because the incoming photons to be teleported are entangled only in one degree of freedom, for example, polarization (i.e., not at the same time in momentum or energy). "The use of hyperentanglement of photons, unfortunately, does not offer advantages for teleportation . . . having only 50% probability of success." [320] Nonlinear setups, on the other hand, cannot offer teleportation efficiency over 50% [143] with the current technology. Thus, the above near-deterministic teleportation remain, for the time being, the only feasible implementation of teleportation with linear optics and the current technology.

A realistic experiment is feasible with the current technology. As for the losses in the system, they are minimal. Metallic beam splitters can be gold-coated with losses as low as 1%. For all the other beam splitters, we can utilize dielectric BSs (with the losses of 0.1%; they were used, for example, in [280]). This means that we can easily carry out a postselection experiment, though what we need for quantum computation and communication are preselected photons.

A source of preselected *event ready* photons, that is, *photons on demand*, is currently the least efficient element of the setup. Four photon [237] and six photon [292] down-conversion schemes for obtaining such a source have been proposed. Both are probabilistic. The former one just increases the probability of the idler being in its beam once the signal is detected (by making use of a much bigger pinhole for the idler than for the signal; see Figure 1.50b). The latter one has recently been experimentally realized [16]. In it, a detection of four photons herald the remaining two, provided only six (and not 8 or more) photons have been generated. The success probability of such an event is 10^{-6} [150].

One of the main reasons for the inefficiency of obtaining the photon pairs from a nonlinear crystal is that they are not born at a spot, but within the volume of the crystal. As shown in Figure 1.50b, we can increase the probability of obtaining one of the photons controlled by the other with the help of a much larger pinhole (ph) for the former photon. When we combine two such pairs as shown in Figure 1.50a, we can increase the efficiency of obtaining a photon pair consisting of, for example, idlers from each pair controlled by the two signals.

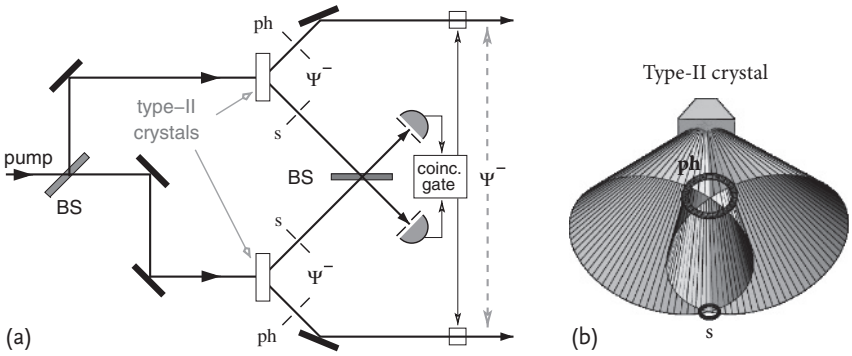


Figure 1.50 (a) Photons almost on demand according to [237]; The case of both pairs coming from the same crystal can be eliminated by an additional beam-splitter (Mach–Zehnder), not shown here; (b) down-converted photon cones; see Figure 1.27.

An even more effective way (than using pinholes of different sizes) to increase the efficiency by an order of magnitude is to put the crystal in a cavity as shown in Figure 1.51 [279]. The cavity not only enhances the production of the photon pairs, but also bunches the down-converted photons at the exit. The perpendicularly polarized idler and signal photons can then be split by a polarizing beam splitter. Two such cavities can be combined within a setup shown in Figure 1.50a to give a more efficient source of photons on demand.

We can obtain entangled photons (ideally) on demand using two cavity enhanced down-conversions (shown in Figure 1.51) combined as in Figure 1.50.

It is obvious that the implementation of teleportation would be much more efficient and feasible without ancillas. Lütkenhaus' [184] *no-go proofs* does allow for such a near-deterministic implementation of it, but it is not clear whether the theorem can be strengthened to actually exclude nonambiguous discrimination of more than two Bell states or if we should do a computer search for all possible combinations that first disentangle the incoming states and then only partially reentangle them.

In any case, a straightforward recombination of whole or disentangled states is not possible. Here are some of the reasons.

- In order to distinguish Ψ and Φ states from each other, we have to make use of the phase shift between the horizontal and vertical polarizations; for that, we have to let them through the beam splitter and/or wave plates; on a HWP, Ψ^- and Φ^+ remain unchanged and Ψ^+ and Φ^- turn into each other; so, we do not get anything; on a beam splitter, either Ψ^- or Ψ^+ splits, but the other two do not.
- We have to start to manipulate incoming Bell states by allowing them to implement their correlation, that is, the phase difference between their polarization; the reason is that each of the photons in a Bell pair is genuinely unpolarized; the only obvious linear optics element in which photons can exhibit their correlations and phase shifts, apart from the above considered wave plates, is a beam splitter; it, let us say, splits Ψ^- and sends Ψ^+ and $\Phi^\mp$ either left or right; when bunched together in one spatial mode, the $\Phi^\mp$ photons keep their states in; say, in mode 1 (left outgoing port of the beam splitter), their states are $|\Phi^\mp\rangle_{11} = (|00\rangle_{11} \mp |11\rangle_{11})/\sqrt{2}$. We cannot have $\Psi^\mp$ in one spatial mode because: $|\Psi^-\rangle_{11} = (|01\rangle_{11} - |10\rangle_{11})/\sqrt{2} = 0$ and $|\Psi^+\rangle_{11} = (|01\rangle_{11} + |10\rangle_{11})/\sqrt{2} = \sqrt{2}|10\rangle_{11}$ (notice the forbidden probability 2). So, we can only have $|01\rangle_{11} (= |10\rangle_{11})$ in one spatial mode.

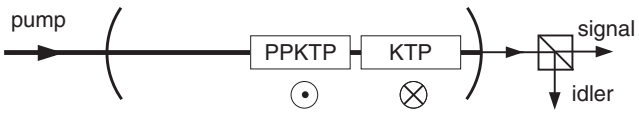


Figure 1.51 Cavity-enhanced down-conversion. The down-conversion occurs in PPKTP (periodically poled potassium titanyl phosphate ($KTiOPO_4$)); the other KTP is used for path compensation (KTP is a birefringent crystal and horizontal and vertical beams have different paths).

- Once we have photons in states $|01\rangle$ and $|\Phi^{\mp}\rangle$ in the same channel (of course, not simultaneously), there is no obvious combination of linear elements which would preserve the states $|\Phi^{\mp}\rangle$ (in order to be able to separate them from each other by a wave plate) and at the same time separate their channel from another channel $|01\rangle$ photons would take. For example, if we send $|\Phi^{\mp}\rangle$ to a Mach–Zehnder interferometer which consists of two beam splitters (BS) of the same kind (see Properties 46 in Section 1.14), then we will have the following transformations (in-1st-BS→out-of-1st-BS→2nd-BS-out) for $|\Phi^{\mp}\rangle$:

$$\begin{aligned} a_{2x}^{\dagger 2} \mp a_{2y}^{\dagger 2} &\to \frac{1}{2}\left(b_{1x}^{\dagger 2} \mp b_{1y}^{\dagger 2} + b_{2x}^{\dagger 2} \mp b_{2y}^{\dagger 2}\right) + b_{1x}^{\dagger} b_{2x}^{\dagger} \mp b_{1y}^{\dagger} b_{2y}^{\dagger} \\ &\to c_{1x}^{\dagger 2} \mp c_{1y}^{\dagger 2} \end{aligned} \tag{1.238}$$

and for $|01\rangle$:

$$a_{2x}^{\dagger} a_{2y}^{\dagger} \to \frac{1}{2}\left(b_{1x}^{\dagger} b_{1y}^{\dagger} + b_{2x}^{\dagger} b_{2y}^{\dagger} + b_{1x}^{\dagger} b_{2y}^{\dagger} + b_{2x}^{\dagger} b_{1y}^{\dagger}\right) \to c_{1x}^{\dagger} c_{1y}^{\dagger} . \tag{1.239}$$

In other words, what comes in, goes out. With a Mach–Zehnder interferometer which consists of two beam splitters (BS) of different kinds (see Properties 46 in Section 1.14) (or by simply changing the sign of a vertical polarization in one arm of the previous Mach–Zehnder setup), we can split the photons as follows

$$\begin{aligned} a_{2x}^{\dagger 2} \mp a_{2y}^{\dagger 2} &\to \frac{1}{2}\left(b_{1x}^{\dagger 2} \mp b_{1y}^{\dagger 2} + b_{2x}^{\dagger 2} \mp b_{2y}^{\dagger 2}\right) + b_{1x}^{\dagger} b_{2x}^{\dagger} \pm b_{1y}^{\dagger} b_{2y}^{\dagger} \\ &\to c_{1x}^{\dagger 2} \mp c_{2y}^{\dagger 2} \end{aligned} \tag{1.240}$$

and for $|01\rangle$:

$$a_{2x}^{\dagger} a_{2y}^{\dagger} \to \frac{1}{2}\left(-b_{1x}^{\dagger} b_{1y}^{\dagger} + b_{2x}^{\dagger} b_{2y}^{\dagger} + b_{1x}^{\dagger} b_{2y}^{\dagger} - b_{2x}^{\dagger} b_{1y}^{\dagger}\right) \to c_{1x}^{\dagger} c_{2y}^{\dagger} . \tag{1.241}$$

Now, the photons that originally were in Φ states are not in that state any more – the photons that originated from either Φ^{+} or Φ^{-} are now just two photons of the same polarization bunched together and there is no way to tell one from the other.
- We can set the photons apart, for example, by means of a polarizing beam splitter, and then put them partially together to beams splitter, wave plates, resonators, or any other linear optics element and then combine various detections at the end of various photon paths and see whether any combination can serve us to unambiguously discriminate the states with an efficiency of more than 50%. So far, no one successfully carried out the task and perhaps we will soon have a proof that something like that is impossible.

At the end of this teleportation section, we would like to stress that it is not only that because of the *No cloning theorem* (Theorem 52) we cannot broadcast a pure unknown state (only teleport it) – we also cannot broadcast a noncommuting mixed state [11]. This result is called the *no broadcasting theorem* and for mixed states, it is a generalization of the *no cloning theorem*.

1.17
Unperformed Measurements Have no Values: Kochen–Specker Sets

In the previous sections, we have seen that the main quantum communication protocols – superdense coding and teleportation – are based on entanglement of qubits. We have also seen that observables of entangled qubits cannot in general be ascribed definite values which would correspond to experimental detections – see Property 47 in Section 1.15.

Here, we stress that four pairs of measurements we presented in Figures 1.36–1.38 cannot be performed simultaneously. We just assume that we performed them and that brings us into a contradiction. Such a reasoning is called *counterfactual reasoning* and such imagined but unperformed experiments are called *counterfactual measurements*. Asher Peres used to say: "Unperformed measurements have no values." [254, Section 6.4]

Another way to express this property is to say that any measurement of a quantum system has a value independent of other compatible measurements carried out at the same time, that is, that the value depends on the *context*. For example, measurements of detectors in Figures 1.36a and 1.38a will give different results for each detector independently of outcomes of the other two measurements that also include two of these detectors. This is called *quantum contextuality*. A quantum theory is therefore a contextual theory and a classical theory is a noncontextual theory. In a classical theory, a variable has a value which it keeps no matter with which other variable we simultaneously measure it.

Constructive proofs of quantum contextuality are provided by the so-called *Kochen–Specker sets* – see Definition 54. They offer us a straightforward blueprint for experimental setups of the contextuality proofs. Kochen–Specker sets are likely to find applications in the field of quantum information, similar to those recently found for the Bell setups in implementing entanglements [57, 121] because a recent result of Adàn Cabello [53] shows that local contextuality can be used to reveal quantum nonlocality. Also, it has been shown that Kochen–Specker sets can enhance quantum correlations. On the other hand, Kochen–Specker sets have recently enabled a particular Hilbert space description of concatenated states and their evolution as well as many-qubit interactions [199, 240, 243].

An additional reason to describe Kochen–Specker sets here in some detail is that they, by definition, include blueprints for quantum gates that prepare states that cannot be given a classical rendering. For quite some time, Kochen–Specker sets were considered exotic because only very few of them were (about a dozen) known – of those that were applicable to qubits, less than 10. Recently however, it was discovered that there are billions of Kochen–Specker sets.

A series of Kochen–Specker experiments have been carried out in the last ten years. The most recent ones made use of quantum gates and employed recently developed quantum information techniques of handling, manipulating, and measuring of qubits by means of quantum circuits of such gates. The experiments were proposed, designed, and carried out for spin $-1/2 \otimes 1/2$ particles (correlated photons or spatial and spin neutron degrees of freedom) [5, 15, 50, 56, 120, 128, 145,

174, 204, 210, 289]. The Kochen–Specker sets that were used in these experiments were either from the 24-24 class of Kochen–Specker sets (set with 18 through 24 vectors and 9 through 24 orthogonal vector tetrads) or the Mermin set [247]. Both approaches aim to find a particular valuation of the Kochen–Specker observables that prove quantum contextuality and disprove any noncontextual classical valuations of those observables.

Let us first briefly present Mermin's proof [201]. It considers the following Kochen–Specker set (cf. (1.41)).

$$\begin{aligned}
&\Sigma_{11} = \sigma_z^{(1)} \otimes I^{(2)}\,, \quad \Sigma_{12} = I^{(1)} \otimes \sigma_z^{(2)}\,, \quad \Sigma_{13} = \sigma_z^{(1)} \otimes \sigma_z^{(2)}\,,\\
&\Sigma_{21} = I^{(1)} \otimes \sigma_x^{(2)}\,, \quad \Sigma_{22} = \sigma_x^{(1)} \otimes I^{(2)}\,, \quad \Sigma_{23} = \sigma_x^{(1)} \otimes \sigma_x^{(2)}\,,\\
&\Sigma_{31} = \sigma_z^{(1)} \otimes \sigma_x^{(2)}\,, \quad \Sigma_{32} = \sigma_x^{(1)} \otimes \sigma_z^{(2)}\,, \quad \Sigma_{33} = \sigma_y^{(1)} \otimes \sigma_y^{(2)}\,.
\end{aligned} \tag{1.242}$$

The definition of the tensor product $A \otimes B$ is given by Definition 22.

The product $\Sigma_{13}\Sigma_{23}\Sigma_{33}$ applied to whatever vector, say a triplet $|\Psi^+\rangle = |\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle$ – represented in (1.243) by means of 1-column 4-row matrix – gives

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} = - \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}, \tag{1.243}$$

that is, $\Sigma_{13}\Sigma_{23}\Sigma_{33} = -I$ which yields $-(|\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle)$. All the other products in all rows and columns of (1.242) are equal to $+I$. Let us denote $R_i = \Sigma_{i1}\Sigma_{i2}\Sigma_{i3}$, $C_j = \Sigma_{1j}\Sigma_{2j}\Sigma_{3j}$, $i, j = 1, 2, 3$. Thus, $R_1 = R_2 = R_3 = C_1 = C_2 = I$ and $C_3 = -I$.

Now, we compare quantum contextual theory with a hypothetical noncontextual theory. In the literature [145, 202], the above observables are assigned predetermined valuations $v(\Sigma_{ij}) = \pm 1$. If we wanted to reproduce quantum eigenvalues by means of such a valuation, valuations of rows should be $v(R_1) = v(\Sigma_{11})v(\Sigma_{12})v(\Sigma_{13}) = 1$, $v(R_2) = 1$, and $v(R_3) = 1$, and of columns $v(C_1) = 1$, $v(C_2) = 1$, and $v(C_3) = -1$. These six equations for $v(\Sigma_{12}), \ldots, v(\Sigma_{33})$ do not have a solution though, that is, there can be no such valuation. This is taken to rule out any noncontextual theory.

On the other hand, B.R. La Cour [163] has recently shown that noncontextual models need not be so simple and that one can construct a more sophisticated noncontextual theory based on probabilistically defined numerical valuation for the above observables. For such a nonlocal theory, the experiments carried out so far prove to be inconclusive. Several "patches" were offered in [111, 163]. Numerical valuations themselves are incompatible with quantum mechanics not only through the many known Kochen–Specker sets, but also through the recently proven result according to which the Kochen–Specker theorem follows from the impossibility of assigning numerical valuation to quantum observables [47].

A meticulous critic can object to the above numerical valuation as being too stretched because the considered Mermin's wave vectors are not always eigenvectors of Σ_{ij}, although it is claimed that the setups apply to all wave vectors, that is, that they are *state independent*.

To see this, let us have a look at how Σ_{11} and Σ_{21} act on triplet $|\Psi^+\rangle$

$$\left(\sigma_z^{(1)} \otimes I^{(2)}\right)|\Psi^+\rangle = |\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle \qquad \left(I^{(1)} \otimes \sigma_x^{(2)}\right)|\Psi^+\rangle = |\uparrow\uparrow\rangle + |\downarrow\downarrow\rangle$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ -1 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \tag{1.244}$$

These observables change the triplet state into other states (singlet and another triplet). So, Σ_{11} and Σ_{21} are not equal to $\pm I$ and ascribing it a valuation ± 1 is ungrounded because 1 or -1 do not correspond to eigenvalues in a measurement. Only $R_1 = I, \ldots, C_3 = -I$ are eigenvalues for any state vector and therefore their valuations $v(R_1) = 1, \ldots, v(C_3) = -1$ are acceptable as predetermined values for a noncontextual model. But, since $|\Psi^+\rangle$ is not an eigenvector of all Σ_{ij}, we do not have a clear cut experimental procedure that would force us to ascribe numerical values to such Σ_{ij}s. Therefore, we shall also consider another type of Kochen–Specker set where experimental outcomes, that is, "clicks," are directly connected to assumed classical values.

The non-Mermin types of Kochen–Specker sets we are going to consider are 4-dim spin systems that can only exit subsequent gates through one of the ports and receive either a valuation 1 or a valuation 0. This corresponds to an observable which has the wave vectors of the system exiting through a particular port (spin projection) as its eigenvectors. When we consider spin-3/2 systems that form a 4-dim Hilbert space, possible experimental implementations are obvious. For example, we consider a series of Stern–Gerlach devices and orient them at angles that determine vectors of a Kochen–Specker set. We let a system through one device then bend its trajectory adiabatically, let it through another device and so on. This will become clear through our elaboration of Kochen–Specker systems below. In the end, we shall come back to the two-qubit case.

To find Kochen–Specker vectors, we shall first restate the Kochen–Specker theorem where the Kochen–Specker sets stem from.

Theorem 53 Kochen–Specker (1967)

There exists a set of measurements that can be carried out on a finite dimensional quantum system in such a way that if one assumed that the values of measured observables are completely independent of all other observables that can be measured on the same system, then one would run into a contradiction [149].

Hence, a quantum system cannot generally possess a definite value of a measurable property prior to a measurement, and quantum measurements (essentially

detector clicks) carried out on quantum systems cannot always be ascribed predetermined values (say zero and one). To arrive at the claim, one considers an orthonormal set of vectors $\{|\psi_1\rangle, \ldots, |\psi_n\rangle\}$ in n-dim Hilbert space, $\mathcal{H}^n$, $n \geq 3$. Projectors onto these states satisfy: $\sum_{i=1}^{n} P_i = I$, where $P_i = |\psi_i\rangle\langle\psi_i|$. Now, Theorem 53 is proved as soon as we prove that there is no function $f : \mathcal{H} \to \mathbb{R}$ satisfying the Sum Rule $\sum_{i=1}^{n} f(P_i) = f(\sum_{i=1}^{n} P_i) = f(I)$ for all sets of projectors P_i. Since that means that there is at least one set of projectors $\{P_i, P_i', \ldots\}$ and the corresponding set of vectors $\{|\psi_i\rangle, |\psi_i'\rangle, \ldots\}$ for which the Sum Rule is not satisfied, the theorem can be proved by any such set of vectors which we call a *Kochen–Specker set* and for which Jason Zimba and Roger Penrose [345] found the following formulation.

Definition 54 The Kochen–Specker set is a set of vectors $|\psi_i\rangle, |\psi_i'\rangle, \ldots$ in $\mathcal{H}^n$, $n \geq 3$ to which it is impossible to assign 0s and 1s in such a way that:

1. No two orthogonal vectors are both assigned the value 1;
2. In any subset of n mutually orthogonal vectors, not all them are assigned the value 0.

Definition 55 Vectors from a Kochen–Specker set are called *Kochen–Specker vectors*.

Lemma 56

Each Kochen–Specker set proves the Kochen–Specker theorem.

The "only" problem was how to find Kochen–Specker sets. Let us first formulate the problem.

Kochen–Specker vectors from an n-dim Hilbert space form subsets of n mutually orthogonal vectors. We arrive at each of them from a previous subset by a series of rotation in 2-dim planes around $(n-2)$-dim subspaces as explained below. Thus, any two "neighboring" subsets (that is, those that follow from each other by just one rotation) share at least one vector which is orthogonal to all other vectors in both subsets and in an n-dim space, two subsets can share up to $n-2$ vectors. The Kochen–Specker vectors correspond to the directions of the quantization axes of the measured eigenstates within experiments which have no classical counterparts, and when we speak of finding Kochen–Specker vectors, we mean finding these directions. We stress here that it is not our aim to give yet another proof of the Kochen–Specker theorem, but to determine the class of all Kochen–Specker vectors in an arbitrary $\mathcal{H}^n$ as well as the class of all *non-Kochen–Specker vectors*, that is, vectors from the remaining sets of vectors from $\mathcal{H}^n$. By the class of non-Kochen–Specker vectors, we mean vectors that allow `0-1` states and that correspond to the directions of the quantization axes of the measured eigenstates within experiments which do have classical counterparts, and when we speak of finding non-Kochen–Specker vectors, we mean finding the latter directions.

Mutual orthogonality of a subset of four vectors in a 4-dim Hilbert space is represented by the following six nonlinear equations:

$$\begin{aligned}
\boldsymbol{a}_A \cdot \boldsymbol{a}_B &= a_{A1}a_{B1} + a_{A2}a_{B2} + a_{A3}a_{B3} + a_{A4}a_{B4} = 0\,,\\
\boldsymbol{a}_A \cdot \boldsymbol{a}_C &= a_{A1}a_{C1} + a_{A2}a_{C2} + a_{A3}a_{C3} + a_{A4}a_{C4} = 0\,,\\
\boldsymbol{a}_A \cdot \boldsymbol{a}_D &= a_{A1}a_{D1} + a_{A2}a_{D2} + a_{A3}a_{D3} + a_{A4}a_{D4} = 0\,,\\
\boldsymbol{a}_B \cdot \boldsymbol{a}_C &= a_{B1}a_{C1} + a_{B2}a_{C2} + a_{B3}a_{C3} + a_{B4}a_{C4} = 0\,,\\
\boldsymbol{a}_B \cdot \boldsymbol{a}_D &= a_{B1}a_{D1} + a_{B2}a_{D2} + a_{B3}a_{D3} + a_{B4}a_{D4} = 0\,,\\
\boldsymbol{a}_C \cdot \boldsymbol{a}_D &= a_{C1}a_{D1} + a_{C2}a_{D2} + a_{C3}a_{D3} + a_{C4}a_{D4} = 0\,.
\end{aligned} \tag{1.245}$$

Now, it might seem that the problem can be approached by a brute computational force. We ascribe values 0 and 1 to various sets of connected quadruples of vectors according to the rules from Definition 54 and as soon as we find a set for which we cannot do that, it is a Kochen–Specker. However, this is easier said than done because there are billions of such sets, even if we limit ourselves only to unit components along each of four axes for all vectors.

> Actually, to find even the smallest Kochen–Specker sets by brute force, all computers on the Globe, calculating just that and nothing else, would need more time than the Age of the Universe.

Therefore, we reasoned

- the hardest problem is the coordinatization problem, that is, to generate set of vectors with definite coordinates;
- a much easier problem is to generate sets of subsets of mutually orthogonal vectors with unspecified coordinates and ascribe them 0 and 1 according to the rules from Definition 54;
- but what are "mutually orthogonal vectors with unspecified coordinates?"
- there can be elements of a set in which we can consistently define *orthogonality* between them;
- to do that, we consider the meaning of orthogonality in an n-dim space.

Orthogonal vectors denote directions of spin projections. Subsets of orthogonal vectors are connected by one of them. Around this direction, we rotate a system with respect to previous spin orientations of the system. The definition of rotation plays an important role here.

A 2-dim rotation is a rotation by an angle around a fixed point in the plane. A 3-dim rotation is a rotation in a 2-dim plane – a subspace of the 3-dim space – by an angle around a fixed axis perpendicular to this plane. A 4-dim rotation is a rotation in a 2-dim plane by an angle around a fixed 2-dim plane. What is common to all these rotations is that they always take place in a 2-dim plane around its complement. Hence, we define an n-dim rotation as a rotation in a 2-dim plane by an angle around a chosen $(n-2)$-dim subspace [88].

Thus, the orthogonality of two vectors can be viewed in the light of rotation as follows. Since a 2-dim Hilbert systems can always have a classical interpretation, we shall start with 3-dim systems. In it, two mutually orthogonal vectors always uniquely determine a ray (line) that contains a third vector perpendicular to both of them. That means we always have unique triples of vectors in the sense that we cannot have two vectors $\boldsymbol{c}$ and $\boldsymbol{d}$ which are both perpendicular to $\boldsymbol{a}$ to $\boldsymbol{b}$ and which are not parallel or antiparallel to each other. In other words,

- in a 3-dim space, two triads can have only one vector in common.
- in a 4-dim space, two tetrads can have at most two vectors in common;

  In a 4-dim space, the rotation is of one plane around another plane, two mutually orthogonal vectors in each plane keep their orthogonality across the planes when we rotate them both in any of the planes. Thus, two tetrads of vectors can have two vectors in common, for instance, the following tetrads:

$$\begin{aligned}
&\boldsymbol{a} = \{0,0,0,1\}\,, \quad \boldsymbol{b} = \{1,0,0,0\}\,, \quad \boldsymbol{c} = \{0,0,1,0\}\,, \quad \boldsymbol{d} = \{0,1,0,0\}\,,\\
&\boldsymbol{a} = \{0,0,0,1\}\,, \quad \boldsymbol{b} = \{1,0,0,0\}\,, \quad \boldsymbol{e} = \{0,1,1,0\}\,, \quad \boldsymbol{f} = \{0,1,-1,0\}\,,\\
&\boldsymbol{a} = \{0,0,0,1\}\,, \quad \boldsymbol{b} = \{1,0,0,0\}\,, \quad \boldsymbol{g} = \{0,1,2,0\}\,, \quad \boldsymbol{h} = \{0,2,-1,0\}\,,\\
&\text{and so on;}
\end{aligned} \tag{1.246}$$

- in a 5-dim space, a rotation in a plane takes place around a 3-dim subspace, so a pentad can have three vectors – from the 3-dim subspace – in common;
- in an n-dim space, two n-ads can have at most $n-2$ vectors in common.

This minimal characterization of orthogonality proves to be necessary and sufficient for our purpose of finding unspecified "potential" vectors. Our idea is to find vectors without coordinates – "just letters" – and see whether we can ascribe them values 0 and 1 according to our rules. When we find a set that *cannot* be ascribed 0 and 1, then we have to check whether we *can* ascribe coordinates to potential vectors represented by vertices of the set. If we can, potential vectors turn into real ones and we have found a Kochen–Specker set. For most of the sets, this cannot be done, that is, the corresponding set of equations like the one shown by (1.245).

Since potential vectors are just letters organized in connected n-ads, their graphical representation boils down to hypergraphs with letters as vertices and n-ads as edges. The above properties of n-ads gives us the following definition.

Definition 57 McKay–Megill–Pavičić (MMP) hypergraphs (diagrams)

a) Every vertex belongs to at least one edge;
b) Every edge contains at least three vertices;
c) Edges that intersect each other in $n-2$ vertices contain at least n vertices.

We encode MMP hypergraphs by means of alphanumeric and other printable ASCII characters. Each vertex is represented by one of the following characters: 1 2 3 4 5 6 7 8 9 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z ! " # $ % & ' ( ) * - / : ; < = > ? @ [\ ] ^ _ ` { | } ~ and then again all these characters prefixed by "+", then prefixed by "++", and so on. There is no upper limit on the number of characters.

Rules for encoding *MMP* hypergraphs in *ASCII* characters:

a) Each edge is represented by a string of characters that represent vertices (without spaces).
b) Edges are separated by commas (without spaces).
c) All edges in a line form a representation of a hypergraph.
d) The order of the edges is irrelevant, however, we shall often present them starting with edges forming the biggest loop to facilitate their possible drawing.
e) The line must end with a full stop.
f) Skipping of characters is allowed.

To obtain MMP hypergraphs, we start with isomorphism-free generation of MMP hypergraphs which consists in the following procedure. Deleting an edge from an MMP hypergraph, together with any vertices that lie only on that edge, yields another MPP hypergraph (perhaps the vacuous one with no vertices). Consequently, every MMP hypergraph can be constructed by starting with the vacuous hypergraph and adding one edge at a time, at each stage having an MMP hypergraph.

In this process we deal with MMP hypergraphs whose vertices and edges have unique labels. We can view the MMP hypergraphs as "rooted trees." The vacuous hypergraph is at the root of a tree, and for any other hypergraph, its parent node is the hypergraph formed by deleting the edge with the highest label. The isomorph rejection problem is to prune this tree until it contains just one representative of each isomorphism class of a hypergraph. This can be achieved by the application of two *rules*.

Rules for generating isomorphism-free MMP hypergraphs

Rule 1 Given a hypergraph D, we can identify the valid positions to add a new edge such that conditions a)–c) from Definition 57 are enforced. According to the symmetries of D, some of these positions are equivalent. The rule is that

- exactly one position in each equivalence class of positions is used; a node in the tree formed by adding an edge in any other position is deleted together with all its possible descendants.

Rule 2 To understand the second rule, consider a hypergraph D′ with at least one edge. We label the edges of D′ in canonical order, which is an order independent of any previous labeling. Then, we define the *major class* of edges as those

that are equivalent under the symmetries of D′ to the edge that is last in canonical order. The rule is:

- when D′ is constructed by adding an edge e to a smaller hypergraph, delete D′ (and all its possible descendants) unless e is in the major class of edges of D′.

According to the theory developed by Brendan D. McKay in [195], application of both rules together is sufficient: exactly one hypergraph from each isomorphism class remains in the tree. Our implementation used `nauty` [194] for computing symmetries and canonical orderings. The method allows for a very efficient parallelization of the computation. A generation tree for MMP hypergraphs with nine vertices and the smallest loops of size 5 is shown in the Figure 1.52.

This definition also serves us to build a constructive algorithm for generating MMP hypergraphs. The other ingredient of the algorithm is the geometry of rotation. For instance, if we consider a tetrad (4-tuple) `1234` and if `1` $= [0, 0, 0, 1]$, then the fourth components of `2`, `3`, `4` are equal to 0. Then, we use value constraints that we obtain from the very definitions and the geometry to make tables in a preliminary pass. These tables are what we use in subsequent passes. We filter the obtained hypergraphs by means of a program based on a simple algorithm which consists in exhaustive assigning values 0 and 1 to MMP edges according to the rules from Definition 54. Smaller hypergraphs – the smallest are shown in Figure 1.53 – and the huge majority of all the others do not have a solution, that is, they cannot be represented by vectors.

In order to find hypergraphs that allow a vector representation, we either let them through another filter that attempts coordinatization based on predetermined component values, for example, from the set $\{-1, 0, 1\}$, or let them through an *interval analysis* solver that reverses hypergraphs to original nonlinear equations and tries

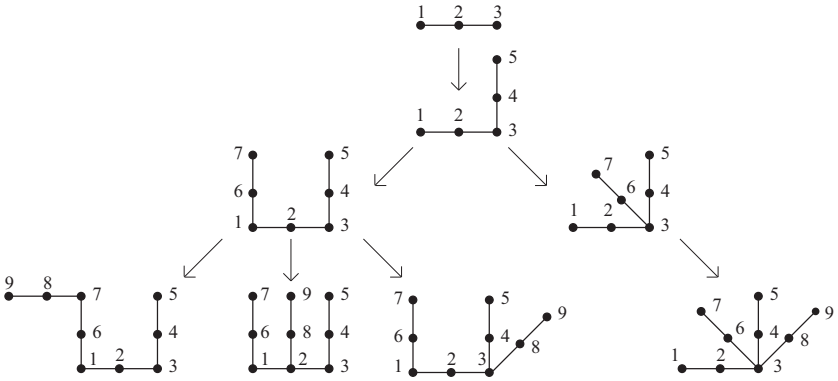


Figure 1.52 An example of a generation tree for connected MMP hypergraphs: nine vertices and the smallest loop of size 5 (for nine vertices, a loop cannot be formed; the first loop appears with ten vertices: `123`, `345`, `567`, `789`, `9A1`). Cf. [196].

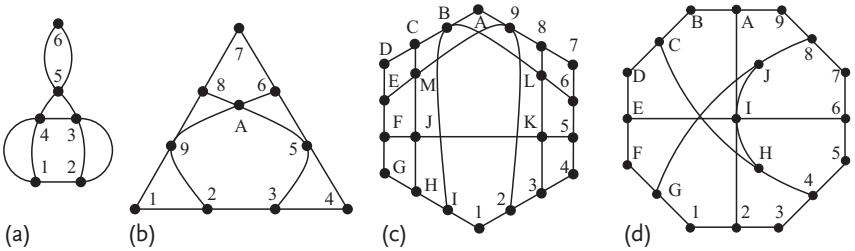


Figure 1.53 Smallest MMP hypergraphs without 0–1 states: (1) four vertices per edge: (a) loops of size 2: six vertices – three edges; (b) loops of size 3: 10-5; (c) loops of size 4: 22-11; (2) three vertices per edge: (d) loops of size 5: 19–13 – one of two smallest 3-dim Kochen–Specker sets. No one of (a–d) MMPs can be given a vector representation in a Hilbert space.

to solve them with the help of interval analysis. Hypergraphs whose equations have a solution and therefore allow vector representation and coordinatization are the sought after Kochen–Specker sets.

Our implementation of the above algorithms proved the feasibility of the exhaustive generation of Kochen–Specker sets because it turned out that although the generation is extremely computationally demanding, its complexity is not exponentially growing.

> Although all three types of algorithms – MMP hypergraph generation with geometry-based elimination, 0-1 filtering, and interval analysis solving of nonlinear equations – have, in principle, an exponential complexity, our implementation of the algorithms shows that their complexity reduces to a statistically polynomial one.

Almost all results presented in this section are obtained by running programs based on various MMP algorithms on large clusters with 500 processors over a week or longer. An upper realistic limit on today's clusters seem to be a generic generation of KS sets with more than 24 edges and vertices. Fortunately, there is a much faster way of generating KS sets: a stripping generation (*stripping technique*). For instance, all the KS sets (1,233 sets) with up to 24 edges we obtained by isomorphism-free generation presented above, follow from a single 24-24 KS shown in Figure 1.54d. The stripping technique consists in removing one edge at the time and it takes just a few minutes on any PC to generate all the remaining 1,232 sets. By this technique we are now able to handle sets with over 100 edges and vertices and generate billions of KS sets from them.

The aforementioned statistical polynomiality of our KS generation is the result of both the structure of our algorithms with geometry-based reduction, backtracking, preliminary passes, pruning, and the structure of hypergraphs and equations where the orthogonality and rotations help filtering and searching to give outcomes in nonexponentially increasing times in the majority of cases. As an example, we

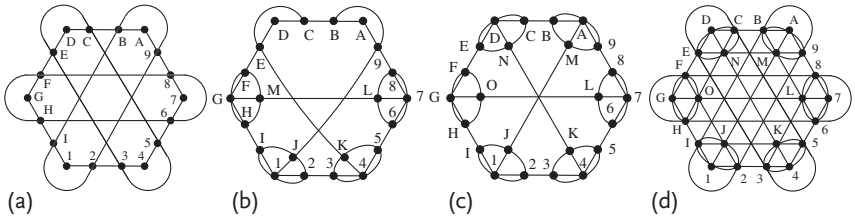


Figure 1.54 MMP hypergraphs of some Kochen–Specker sets from the 24-24 class. (a) `18–9`, (b) `22–13`, and (c) `24–15` are critical; (d) `24–24` contains all 1,232 smaller Kochen–Specker sets with vector component values from $\{-1, 0, 1\}$.

present a reduction in time we achieve with the algorithms for the task of finding the smallest 4-dim Kochen–Specker set with 18 vectors and 9 tetrads (`18–9`).

brute force	10^{100} ? equations	Age of the Universe
	$\Downarrow$	
bare MMP isomorp. generation	$2.9 \cdot 10^{16}$ MMP hypergraphs	30 million years on a 2 GHz CPU
	$\Downarrow$	
	we start here	
	$\Downarrow$	
geometry filtered MMP generation	100220 MMP hypergraphs	$<$ 30 min on a 2 GHz CPU
	$\Downarrow$	
$0-1$ filtered MMP hypergraphs	26800 MMP hypergraphs	$<$ 5 sec on a 2 GHz CPU
	$\Downarrow$	
$\{-1, 0, 1\}$ coordinate filtered MMPs	one $\{-1, 0, 1\}$–vector Kochen–Specker set	$<$ 1 sec on a 2 GHz CPU
	$\Downarrow$	
interval analysis non–$\{-1, 0, 1\}$ vectors	several non–$\{-1, 0, 1\}$ Kochen–Specker sets	$<$ 2 min on a 2 GHz CPU

(1.247)

The set `18–9` is shown in Figure 1.54a wherefrom we can easily read off the MMP hypergraphs in its notation:

`1234,4567,789A,ABCD,DEFG,GHI1,I29B,35CE,68FH.`

Once we have MMP hypergraphs whose equations have a solution we can just ascribe the coordinates of their vector components to the edges of a hypergraph. For instance, possible $\{-1, 0, 1\}$ coordinates for vectors of the `18–9` set are (the

first are of edge/vector 1, the second of 2, ... and the last of I), that is,

$$\{\{0,0,0,1\},\{0,0,1,0\},\{1,-1,0,0\},\{1,1,0,0\},\{0,0,1,-1\},$$
$$\{1,-1,-1,-1\},\{1,-1,1,1\},\{1,1,-1,1\},\{1,0,0,-1\},\{0,1,1,0\},$$
$$\{1,0,0,1\},\{1,1,-1,-1\},\{1,-1,1,-1\},\{1,1,1,1\},\{0,1,0,-1\},$$
$$\{1,0,-1,0\},\{1,0,1,0\},\{0,1,0,0\}\}.$$

We can easily verify that all vectors from each tetrad are mutually orthogonal. For instance,

$$1234 = \{\{0,0,0,1\},\{0,0,1,0\},\{1,-1,0,0\},\{1,1,0,0\}\}$$

obviously are. That the above set really is a Kochen–Specker set we can check using the following rule called the *parity proof*.

Parity proof for Kochen–Specker sets with an odd number of edges and vertices that share an even number of edges.

a) Since each edge according to the rules of Definition 54 must contain at least one 1, each set contains an odd number of 1s and therefore there should be an odd number of vertices equal to 1.
b) Every vertex shares an even number of edges. Hence, there should be an even number of vertices equal to 1 because every 1-vertex would be repeated twice.

a)↔b) a) and b) clash, so no predetermined 0,1 values can be ascribed to the vertices.

When we inspect all the Kochen–Specker sets with up to 24 vertices and 24 edges obtained by means of the above algorithms in a 4-dim Hilbert space, we find the following results [247].

Property 58

Characterization of the 24-24 class of Kochen–Specker vectors.

a) There are altogether at least 1,233 sets with vector component values from $\{-1,0,1\}$. We say that they form the 24-24 Kochen–Specker class.
b) There are at least 37 sets with component values from sets other then $\{-1,0,1\}$ [247] and these sets are not isomorphic to any of the sets from a). Each of them properly contains at least one of the sets from a) as well as some additional edges [247, Figure 3]. The smallest of them is a 22–11 and the largest a 24–14.
c) There are altogether six *critical* KS *sets*: 18–9, two 20–11, two 22–13, and a 24–15. A critical KS set is a KS set which ceases to be a Kochen–Specker (KS) set when we strip it of any of its edges (leaving all those of its vertices that belong to other edges). Only critical sets are experimentally relevant because

additional orthogonalities between already present vertices cannot lead to new measurement results. One can carry out the parity proof on all six critical sets above.
  - 18–9 was first found by Adán Cabello, José M. Estebaranz, and Guillermo García–Alcaine [55] in the "prealgorithmic era" of the Kochen–Specker research;
  - one of the two 20–11 was also found in that era by Michael Kernaghan [141];

d) All 24-24 MMP hypergraphs (billions of them) have been generated and among them there is only one Kochen–Specker set – 24–24 shown in Figure 1.54d. All sets with up to (and including) 23 vertices have been exhaustively generated over two months on a cluster with 500 3.4 GHz CPUs. On the other hand, obtaining all 1,232 sets from 24–24 set from Figure 1.54d by the stripping technique took less than 2 min on a single 3.4 GHz CPU.
  - 24–24 set was first found in the prealgorithmic era by Asher Peres [253];

e) Apart from 24-24 sets, sets with 24 vertices have never been exhaustively generated. (Exhaustive generation of all 24-24 sets took more than two months on the aforementioned cluster.)
  - It is interesting that Asher Peres [254] used a computer program to prove that his 24–24 set really was a Kochen–Specker one – the 24–24 set cannot be given a parity proof. Had he applied a stripping technique to his set, he would have obtained all 1233 KS sets from the 24-24 class already 20 years ago. This vividly illustrates how hypergraphs and their visual representation reveal properties that remain indiscernible in the standard vector approach.

f) All Kochen–Specker sets have a maximal loop of order six – six edges, a hexagon.

Let us sum up all results known about Kochen–Specker set in a 4-dim Hilbert space until 2010.

- Two aforementioned Kochen–Specker sets (Properties 58c)) and several other larger ones, found by P.K. Aravind and Forest Lee-Elkin [6], were known in the "prealgorithmic era." The latter sets were obtained from a 600-cell, which is a regular polytope in four-dimensional Euclidean space with 120 vertices distributed symmetrically on the surface of a four-dimensional sphere.
- By means of the above algorithms altogether, six empirically distinguishable critical sets have been found in the 24–24 class.
- Two other bigger critical sets, derived from the vertices of the 600-cell were found by Mordecai Waegell and P.K. Aravind [314].

So, altogether, very few quantum sets that do not allow a classical representation have been known and to convincingly support the quantumness of quantum formalism, a much larger number of such systems is required and actually were expected to exist and to be found. Therefore, in 2010, Mordecai Waegell, P.K. Aravind, Norman D. Megill and Mladen Pavičić conjectured that there should be abundantly

many "600-cell Kochen–Specker" sets and joined their algorithms to find them. Essentially, they started with a 60-75 Kochen–Specker set containing 60 vertices and 75 edges (Mordecai Waegell and P.K. Aravind derived from the 600-cell [314]) and in a few months time, they obtained millions of nonisomorphic sets and thousands of critical ones among them [246, 316].

Mordecai Waegell and P.K. Aravind developed algorithms for parity proofs with an odd number of edges in the MMP hypergraph representation and terminology or with an odd number of bases in the author's original terminology. Norman Megill and Mladen Pavičić developed algorithms based on the stripping algorithms they used for generating KS sets from the 24-24 KS set. These algorithms are general in the sense of yielding all KS sets – those with an even number of edges as well as with an odd number of edges; it turned out that among the latter ones, there are also many KS sets that do not have a parity proof.

And while both the parity proof algorithms and the stripping algorithms proved to be extremely fast for the small KS sets from the 24-24 class, for the new 60-75 and even more for the newest 60-105 class, they require months on the biggest clusters. So far, we have generated billions of 60-75 and 60-105 KS sets and 150 millions of critical empirically distinguishable KS sets only from the 60-75 class, but there is no way we could exhaustively generate even a tiny percentage of them in the near future (a few percents have been obtained so far). This confirms our conjecture that there really exists an abundant number of sets of quantum states achievable with a corresponding set of quantum gates that cannot be given a classical rendering and that can serve us in implementing future nonlocal and entangled states [54] (analogous to recent applications of the Bell states in *one way quantum computation*) and algorithms in our would-be quantum computers.

Since an exhaustive generation of 60-75 and 60-105 Kochen–Specker sets is not possible with the known algorithms, a probabilistic approach based on random samples has been applied. However, not in the bottom-up approach as originally with KS sets from the 24-24 class, but in the top-down approach by stripping master sets 60-75 and 60-105 that were originally found by P.K. Aravind with the help of geometrical reasoning. This approach enabled us to identify almost every kind of critical KS set in the 60-75 KS class, though the properties of KS sets from these classes are not yet fully recognized and we will just present those that emerged so far from random samples. Some of the sets are shown in Figures 1.55 and 1.56.

Property 59

Characterization of the 60-75 class of Kochen–Specker vectors.

a) Kochen–Specker sets that can be obtained by stripping of one edge at a time from the `60-75` (obtained from the 600-cell) form a class which we call the 60-75 Kochen–Specker class. The 60-75 class does not overlap with the 24-24 class.
b) It is estimated that there might be more than 10^{17} nonisomorphic Kochen–Specker sets and so far more than 10^{13} such sets have been observed;

c) Altogether, over 0.3 billion of *critical sets* have been observed. For them, the following holds. Critical sets with an odd number of edges (which allow for a parity proof) have a different distribution than the ones with an even number of edges. The smallest set of the former kind, `26–13` found by N.D. Megill and M. Pavičić [246], is shown in Figure 1.55a; it has an octagon as a maximal edge loop. The smallest set of the latter kind, `38–22`, is shown in Figure 1.55c. It also has an octagon as a maximal edge loop. The biggest sets of the former and latter kinds are `60–41` and `60–40`, respectively, and with hexadecagon and octadecagon maximal loops, respectively; It has been estimated that the number of critical sets is somewhat larger than the 300 million of them observed so far;

- `30–15` set shown in Figure 1.55b is one of two `30–15` previously found by Mordecai Waegell and P.K. Aravind [314];
- `38–22` set shown in Figure 1.55c is one of the smallest critical Kochen–Specker sets from the 60-75 class with an even number of edges. It therefore does not allow for a parity proof. However, the proof can be carried out by the program `states01` [248] in less than 1 s. The program simply exhaustively checks all the possibilities to ascribe 0 and 1 to vertices and confirms that it is impossible to do so as to satisfy the rules (i) and (ii) of Definition 54. There are also many criticals with an odd number of edges that do not have a parity proof.
- sets shown in Figure 1.56 are small and middle-sized critical Kochen–Specker sets which illustrate the difference in structure of the two classes; 60-105 class overlaps with the 24-24 class. For instance, both classes contain the 18-9 critical as their smallest set. However, all the other small sets, like the one in Figure 1.56a, found so far, are not isomorphic with any of the 24-24 sets.

The above sets were found by the stripping technique described above. Successive extensive applications of the stripping algorithm have been applied on 10^{10} KS sets with over 60 edges. Smaller critical sets are saturated and it is not likely that any more of them exist. For instance, there is one set with 26 edges and 13 vertices

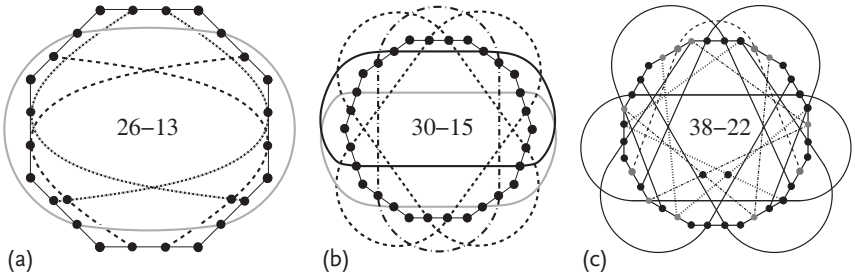


Figure 1.55 Kochen–Specker sets from the 60-75 class. (a) The smallest critical set; (b) set with a decagon as its maximal loop. They all vividly illustrate the parity proof: all their vertices share two edges and they all have an odd number of vertices. (c) The smallest set with an even number of edges. Letters are not shown to stress that they can be arbitrarily assigned to any MMP hypergraph vertex and then used for further computer processing.

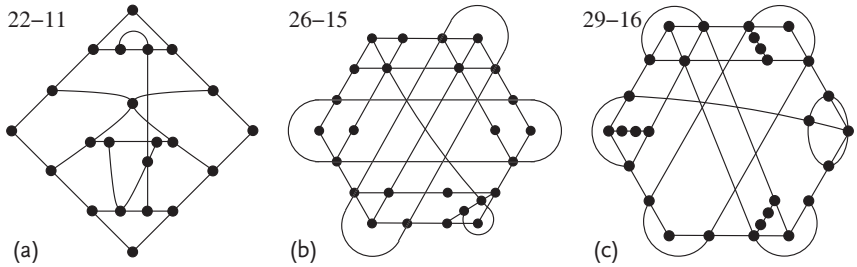


Figure 1.56 Kochen–Specker sets from the 60-105 class. (a) One of the smallest critical sets non-isomorphic with any set from the 24-24 set; (b) one of the smallest sets with an odd number of edges that do not have a parity proof; (c) one of the smallest sets with an even number of edges (compare it with 38-22 from the 60-75 class from Figure 1.55c).

(26-13), 3 30-15, 8 with 17 edges, 26 with 19, 116 with 21, 5 with 22 (22-38), 539 with 23, 946 with 24, and 10 055 of them with 25 edges. In contrast, the number of bigger criticals with 26 to 41 edges increases with each subsequent new stripping. They too show slow saturation but those at the peek (criticals with 32 edges) almost double with repeating the same number of runs. So far, over 300 million criticals have been found among 10^{23} MMP hypergraphs. They are presented in Figure 1.57. Since the total number of KS sets is unknown and the actual distribution of obtained criticals in successive runs changes from edges to edges and from vertices to vertices, we will not attempt to estimate the total number of criticals. The outcome will be known sooner or later anyhow since the generation shows a definite although slow saturation. The point is that no classical statistical estimation method seems to work with the (quantum) KS criticals; compare Figure 1.57 with Figure 3 from [197].

The experiments that one can make using 3- and 4-dim Kochen–Specker sets can be straightforwardly designed. In 3-dim Hilbert space realistic experiments require spin-1 (qutrit) and, as we mentioned above, have been successfully carried out. However, what we would need in quantum computation are KS sets for qubits. A pair of qubits lives in a 4-dim Hilbert space, but classes 24-24 and 60-75 are constructed in a real Hilbert space for vectors that primarily take values in that space for single spin-3/2 systems.

It is possible to find some operators that would enable us to handle a pair of qubits in their 4-dim space so as to define a simple KS set (say 18-9), but what we really need are operator defined Kochen–Specker sets defined for n qubits. And that is exactly how the KS sets in the 60-105 class are defined.

The basis sets of the 60-105 sets are from the complex Hilbert space, so, we can manipulate them by general operator to prepare and get any state. More importantly, we can generate each of them by applying well defined operators on any arbitrary state.

On the other hand, this approach in the complex Hilbert space generalizes a separation between the quantum theory and any classical model for a considered quantum system. Classical variables that we usually consider in refutations of any

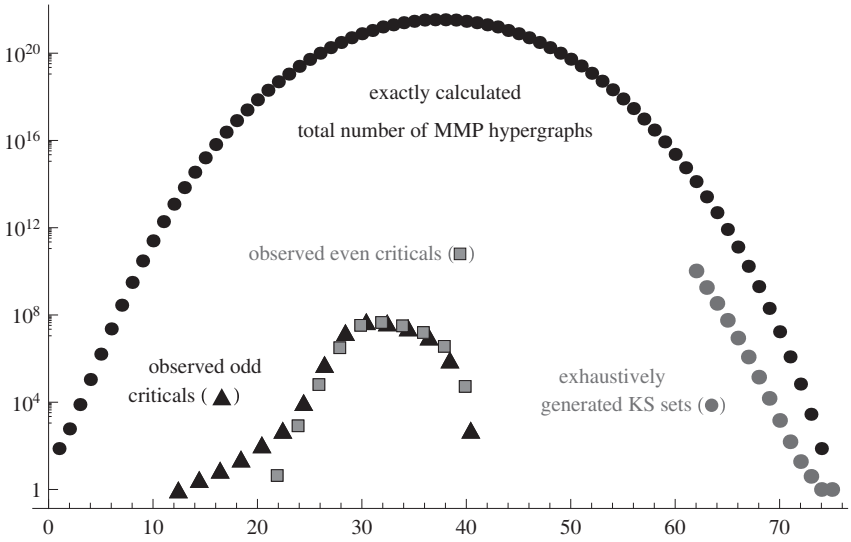


Figure 1.57 Logarithmic plot of a bit over 300 millions of Kochen–Specker critical sets generated in the 60-75 KS class from the `60-75` master set by means of the *stripping technique* and by an algorithm for random generation of critical KS sets.

possible classical ("hidden variable") theory are real numbers ascribed to possible measurements of observables while – see (1.244) and the discussion following it – intermediate unknown eigenvalues of observables during their evolution in a Hilbert space might not even exist for the initial eigenvector or might be complex. Quantum theory does not have problems with any of these aspects of its formalism because in the end, we always measure a mean value of an observable and do not relate amplitude of a wave function while it evolves with any possible measurement outcome.

Nevertheless, if we wanted to refute a possible classical application we should be able to correlate some parts of the wave function as it evolves through the quantum gate with assumed predetermined measurement outcomes. And, the 60-105 KS sets do give us just that. We will be able to ascribe values zero and one to parts of a function in evolution that will have an experimental meaning for both classical and quantum theories. This is because the measurements will not give us an eigenvalue for a total wave function, but rather just confirm that both qubits appear in a particular state when we carry out a measurement on them. We shall present the main blueprint following the original idea of P.K. Aravind and his group [315] and add several outcomes that we obtained on supercomputers using the stripping technique (see Property 58d) earlier in this section.

The idea is to obtain sets of mutually ortogonal eigenvectors of tensor products of Pauli operators. The idea is to extend Mermin's set given earlier in this section and to find a larger set of Kochen–Specker vectors. There are six products of Pauli operators with unit operators: three acting on the first qubit and three on the sec-

ond qubit (e.g., $\Sigma_{11} = \sigma_z^{(1)} \otimes I^{(2)}$ and $\Sigma_{12} = I^{(1)} \otimes \sigma_z^{(2)}$ from (1.242)) and nine mixed products (e.g., $\Sigma_{31} = \sigma_z^{(1)} \otimes \sigma_x^{(2)}$); altogether 15. A simple computer program finds that there 60 different eigenvectors of these operators such that each triple of mutually commuting operators (of the aforementioned ones) have a tetrad of mutually orthogonal eigenvectors in common ($4 \cdot 15 = 60$). Some of them are shown in Table 1.7 where we use quantum computation labels X, Y, Z instead of $\sigma_x, \sigma_y, \sigma_z$. Their full list is given in [315].

Another simple computer program shows that we can recombine the same eigenvectors so as to give us 90 more tetrads of orthogonal eigenvectors. Together, these make $15 + 90 = 105$ tetrads built up by 60 eigenvectors (hence the name `60-105`.) Each of the eigenvectors appears in 7 tetrads; thus we have: $60 \cdot 7/4 = 105$.

To further explore the 60-105 sets we proceed as follows.

We represent the 60-105 sets of eigenvectors with the help of MMP hypergraphs: vertices represent eigenvectors and edges represent tetrads of mutually orthogonal eigenvectors. We encode eigenvectors by means of ASCII characters as `60-105`=1234,..., 9AZa, ..., jlxy. Which vector we denote by which of 60 ASCII characters 1, ..., y is irrelevant. The only thing which is relevant are 105 edges each of which contains four vertices that correspond to four mutually orthogonal eigenvectors. Using the aforementioned algorithms and programs, we find that the `60-105` MMP set is a Kochen–Specker (KS) set. Therefore 60 eigenvectors of 15 Pauli product operators also form a KS set. Smallest experimentally verifiable subsets are KS critical sets and we obtain them by the stripping technique.

The experiments are easy to design but rather complicated to carry out, although it is possible with today's technology. We have to let two qubits through a series of quantum gates described by single qubit operators and CNOT operators. CNOT gate is here to handle one qubit with respect to the other and it is through this interconnection of the two qubits where the contextuality of the Hilbert space enters.

What makes the experiment complicated, is that we have to carry out all the gating with our qubit pairs. Otherwise, we cannot counterfactually assume that all classical variables corresponding to quantum eigenvalues have predetermined

Table 1.7 A sample from a complete list of 15 Pauli operator products and their eigenvectors given in [315].

Pauli product triples	**4 eigenvectors of each product from the triple**			
$X_1 \otimes I_2, I_1 \otimes Z_2, Z_1 \otimes Z_2$	$\|1000\rangle$	$\|0100\rangle$	$\|0010\rangle$	$\|0001\rangle$
$X_1 \otimes I_2, I_1 \otimes X_2, X_1 \otimes X_2$	$\|1111\rangle$	$\|1-11-1\rangle$	$\|11-1-1\rangle$	$\|1-1-11\rangle$
..., ..., ...	...	...	...	...
$Y_1 \otimes I_2, I_1 \otimes Z_2, Y_1 \otimes Z_2$	$\|10i0\rangle$	$\|010i\rangle$	$\|10i0\rangle$	$\|010i\rangle$
..., ..., ...	...	...	...	...
$X_1 \otimes Y_2, Y_1 \otimes X_2, Z_1 \otimes Z_2$	$\|100i\rangle$	$\|01-i0\rangle$	$\|01i0\rangle$	$\|100-i\rangle$
..., ..., ...	...	...	...	...

values from the moment they entered the first gate till the moment they exited the last gate and were measured. In doing so, we indisputably show that predetermined values cannot be ascribed.

We stress here that in contrast to Mermin's procedure given earlier in this section, we can always ascribe an eigenvalue to any of the four eigenvectors from each tetrad. So, it is enough to assume that "if a measurement had been carried out on the qubits, it would have given two clicks in two detectors – no matter in which ones" to run into a contradiction. And, this is the beauty of 60-105 experiments. We ascribe one to any of the "assumed intermediate" results and zero to the other three – all four outcomes are equally possible and it is irrelevant to which we ascribe one and to which 0 s – and still the two qubits appear in the end.[32] So, our KS experimental outcome can be reformulated under the following paradoxical guise: "We cannot assume that the qubits passed through all the gates after they passed through all of them."

How can we design such experiments? We have to know all the tetrads and how they are related to each other via orthogonalities of their eigenvectors to other tetrads. To find this out, we could find all possible critical KS sets by finding all possible corresponding MMP hypergraphs. However, the number of KS subsets is so huge (at least billions of them) that this is simply not feasible with today's computational resources. Therefore, we either exploit possible geometrical symmetries [315] or carry out random searches. The latter approach proved to be much more efficient and the partial results are shown in Table 1.8. They are obtained within a few hours on a cluster. Other randomly picked sets show similar abundances.

The results of a search for "higher criticals," that is, for criticals with more than 40 edges, are shown in Table 1.9. Actually, we can randomly obtain any `60−105` subset which exists, provided we run a program sufficiently long (up to a day) on a cluster. This is, in most cases, not possible by a geometric reasoning [315].

By looking at Tables 1.8 and 1.9 and Figure 1.57, we see that the classes 60-75 and 60-105 show some similarities – the highest numbers of vertices for criticals are 41 and 40 respectively. However, at the lower end, these two classes are very different. While 60-75 starts where 24-24 ends, 60-105 and 24-24 share the smallest set, that is, `18−9`. Both 60-105 and 24-24 also have criticals with 24 or less vertices, but they are not isomorphic.

Hence, we can design experiments starting with the smallest `18−9` KS critical shown in Figure 1.54. Using a computer program, we find eigenvalues that correspond to eigenvectors shown in Table 1.7. Then, we can start with the tetrad $\{|1000\rangle, |0100\rangle, |0010\rangle, |0001\rangle\}$ which corresponds to two gates with two ports each. In Figure 1.54, it corresponds to, say, a 1234-edge. Now, since all tetrads are defined by unitary operators as their eigenvectors, we can evolve 1234 gate states into I29B ones, or 34CE, or 4567, or 1IHG, or any of the remaining eight gate states. After we evolved the initial state through all of them, we measure the outcoming states

32) We stress here that ascribing 1 to possible intermediate experiments is fully compatible with the fact that all four vectors are eigenvectors of the operators that define the tetrad; the absolute value of their eigenvalues is always 1.

Table 1.8 KS criticals generated from a randomly obtained 40–40 set from the 60-105 Class.

	criticals generated from a random 40–40 subset											
edges	**9**	**11**	**13**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**
ver												
18	1											
20		2										
21		2										
22		3										
23			4									
24			19									
25			17									
26			6	20								
27				64								
28				74		1						
29				26	1	31						
30				2	9	292	7					
31						395	212	7				
32						111	286	304				
33						1	116	100	11			
34							37	1038	1525			
35								581	1535	361		
36								53	616	1144	8	
37									88	459	285	
38									18	384	363	8
39										46	135	23
40											28	12

Table 1.9 Results of random generation of criticals with more than 40 edges directly from the master set 60–105. Scanning was stopped as soon one critical was found.

	scanning 60-105 for bigger criticals												
edges	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**	**31**	**32**	**33**	**34**	**40**
ver													
41	1												
42			1										
44	1		1										
45			1										
46			1										
50					1								
51							1	1	1				
53											1		
54											1	1	
55										1			
60													1

of qubits. In any imagined transition of the two qubits, they should exit through two of four ports of each gate they have been subjected to. If we try to ascribe 1 to

whatever imagined states and 0 to three others, then we can easily see by applying the parity proof given earlier in this section that this is not possible. On the other hand, this avoids ascribing value 1 (−1) to a possibly nonexistent eigenvalue during the evolution of an incoming state mentioned earlier in this section below (1.244). Yet, the two ignorant qubits will happily trigger the detectors in the end.

Let us present this in more detail. For example, we want to arrive at $\{|0010\rangle, |0001\rangle, 1/\sqrt{2}|1100\rangle, 1/\sqrt{2}|1-100\rangle\}$ starting from $\{|1000\rangle, |0100\rangle, |0010\rangle, |0001\rangle\}$. To achieve this, we make use of the following operators

$$W = (U_1 \otimes U_2) \cdot \mathrm{CNOT} \cdot (I_1 \otimes U_2) \,, \tag{1.248}$$

where

$$U_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \,, \quad U_2 = \frac{1}{2}\left(\sqrt{2+\sqrt{2}}\right)\begin{pmatrix} -1 & -1+\sqrt{2} \\ -1+\sqrt{2} & 1 \end{pmatrix} \,. \tag{1.249}$$

This yields

$$W = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \tag{1.250}$$

and we can verify

$$\begin{aligned} W : \{&|1000\rangle, |0100\rangle, |0010\rangle, |0001\rangle\} \\ \longrightarrow \Big\{&|0010\rangle, |0001\rangle, \frac{1}{\sqrt{2}}|1100\rangle, \frac{1}{\sqrt{2}}|1-100\rangle\Big\} \,. \end{aligned} \tag{1.251}$$

In the circuit notation, this can be written as shown in Figure 1.58.

In Section 1.18, we show that any 4-dim operator in a 2 ⊗ 2-dim Hilbert space can be expressed by one qubit and CNOT operators. So, we can transform any two basis into each other by means of single qubit gates and CNOT gates and therefore let a pair of qubits through all the gates of billions of Kochen–Specker sets from the 60-105 class, as, for instance, the `22-11` set shown in Figure 1.56a or `18-9` shown in Figure 1.54a.

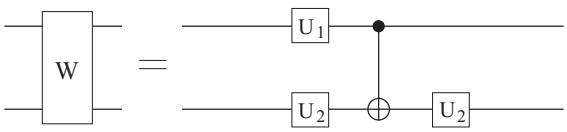


Figure 1.58 Evolving 60-105 KS bases from each other. An example is given (see text).

1.18 Controlling Qubits: Quantum Gates and Circuits

A qubit is a quantum system and is therefore described – as any quantum system – by the time-dependent *Schrödinger equation*:

$$\boxed{i\hbar\frac{\partial}{\partial t}|\psi(t)\rangle = \hat{H}|\psi(t)\rangle}\,, \tag{1.252}$$

where $\hat{H}$ is the *Hamilton operator* commonly called the *Hamiltonian*. It describes all interactions between particles and fields and determines the state of the system in time and space.

Knowing the state of a qubit at the initial time (t=0), we can solve its Schrödinger equation (1.252) to obtain the state at any later time. For a time-independent Hamiltonian, we can write it down as follows

$$|\psi(t)\rangle = e^{-\frac{i}{\hbar}\hat{H}t}|\psi(0)\rangle\,. \tag{1.253}$$

The exponential operator on the right-hand side of the Schrödinger equation is usually defined by the corresponding power series in $\hat{H}$. We say that we take a *matrix exponential* – see (1.267).

We call the operator

$$\boxed{\hat{U} = e^{-\frac{i}{\hbar}\hat{H}t}} \tag{1.254}$$

the *time evolution operator* of a quantum system. It is obviously a unitary operator. When the Hamiltonian is time-independent, it belongs to a one-parameter unitary group.

The Hamiltonian $\hat{H}$ of a system corresponds to its energy in the sense that the energy E is its eigenvalue

$$\boxed{\hat{H}|\psi\rangle = E|\psi\rangle} \tag{1.255}$$

and its expectation value is

$$\boxed{\langle\hat{H}\rangle = \langle\psi|\hat{H}|\psi\rangle = E}\,. \tag{1.256}$$

We will often use unit qubit space vectors, $|\psi_n\rangle$, that span the space in the following way. To decompose the state, we take the projection operator to the whole space – *identity operator*:

$$P_I = \sum_n P_n = \sum_n |\psi_n\rangle\langle\psi_n| = I\,, \tag{1.257}$$

and introduce it into the evolution operator to obtain

$$\hat{U}(t) = e^{-\frac{i}{\hbar}\hat{H}t}\sum_n |\psi_n\rangle\langle\psi_n| = \sum_n e^{-i\omega_n t}|\psi_n\rangle\langle\psi_n|\,, \tag{1.258}$$

where $\omega_n = E_n/\hbar$.

Such an evolution operator applied to an initial state yields

$$|\psi(t)\rangle = \sum_n e^{-i\omega_n t}|\psi_n\rangle\langle\psi_n|\psi(0)\rangle$$
$$= \sum_n e^{-i\omega_n t}c_n|\psi_n\rangle = \sum_n c_n(t)|\psi_n\rangle \ . \tag{1.259}$$

By manipulating unit states and obtaining appropriate coefficients $c_n(t)$, we can obtain any combination and superposition of unit states. And, this is what quantum gates are all about. Manipulations of trapped ions, all-optical photon manipulation, nuclear-magnetic resonance, solid states, superconductive and any other qubit setups that are candidates for the would-be quantum computers and which we present in Chapter 2, are all carried out by means of quantum gates which are described by either giving a Hamiltonian, or projectors, or energies, or states, but the final aim of all of them is to provide an evolution and specify an evolution operator which will govern a state from an initial to a final state, that is, to the result of a calculation.

Like reversible (see Section 1.9) and unlike classical (see Section 1.4) gates, quantum gates are reversible by definition because they are represented by unitary operators from a Hilbert space and therefore represented by reversible matrices. In the previous sections, we have explicitly or implicitly used a number of quantum gates. In this section, we shall give their general formal definitions, but will also refer to any previously mentioned one whenever we come to it.

Formal definitions of quantum gates and quantum circuits they belong to proved to be of utmost importance because they often introduce new physical phenomena and applications. For example, the teleportation was first conceived in a completely formal way by Bennett, Brassard, Crépeau, Jozsa, Peres, and Wootters [23], and only later considered and formulated for physical systems [249, 346] and verified experimentally [37]. Superdense coding was first put forward completely formally by Bennett and Wiesner [24] and only years later experimentally verified. Quantum error correction and quantum algorithms are yet other examples. All these quantum protocols started with general gate formalism and resulted in new physics. Then, why have we not started with formal quantum gates? Because we only know what is an important formal expression or theorem once we found their physical counterparts and applications. Only afterwards, can we have a targeted usage of a formal approach as we do now.

We introduce quantum gates starting with those that are already well-known from the standard quantum mechanics. Since we shall use quantum gates to build quantum circuits, from now on, we shall use their circuit names which we introduce in the first column of Table 1.10. Often, their names will differ from those used in quantum mechanics (e.g., NOT-gate vs. Pauli σ_x matrix) and quantum optics, [for example, H-gate (Hadamard gate) vs. HWP(π/8)] that we used in previous sections.

Table 1.10 2-dim quantum gates.

Two-dimensional quantum gates		
Quantum circuit name(s)	**Matrix**	**q. mechanics or q. optics name**
NOT (X, bit-flip)	$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	Pauli σ_x, HWP($\pi/4$)
Y	$\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$	Pauli σ_y
Z	$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$	Pauli σ_z, HWP(0)
H (Hadamard)	$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$	HWP($\pi/8$)
$\sqrt{\text{NOT}}$	$\frac{1+i}{2}\begin{pmatrix} 1 & -i \\ -i & 1 \end{pmatrix} = \begin{pmatrix} \frac{1+i}{2} & \frac{1-i}{2} \\ \frac{1-i}{2} & \frac{1+i}{2} \end{pmatrix}$	
$\sqrt{\text{NOT}}'$	$\frac{1-i}{2}\begin{pmatrix} 1 & i \\ i & 1 \end{pmatrix}$	$\frac{1-i}{2}$ (beam splitter matrix)
S (phase, $\sqrt{Z}$)	$\begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$	QWP – fast axis $\parallel y$
T ($\pi/8$, $\sqrt{S}$, $\sqrt[4]{Z}$)	$\begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$	
$P(\theta)$ (Phase shift)	$\begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}$	
$R_x(\theta)$ (x-rotation gate)	$\begin{pmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$	SU(2) matrix
$R_y(\theta)$ (y-rotation gate)	$\begin{pmatrix} \cos(\theta/2) & \sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$	SU(2) matrix
$R_z(\theta)$ (z-rotation gate)	$\begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix}$	SU(2) matrix

Definition 60 We define elementary two-dimensional quantum gates NOT, Y, Z, H, $\sqrt{\text{NOT}}$, S, T, $P(\theta)$, $R_x(\theta)$, $R_y(\theta)$, $R_z(\theta)$, and U as given in Table 1.10. (Alternative names that are also in use are given in brackets.)

Some of the gates deserve elaboration and comments.

Matrix parts of both $\sqrt{\text{NOT}}$ and $\sqrt{\text{NOT}}'$ are equal to two special cases of a general form of the so-called *beam splitter matrix* B [61, 219], that is,

$$B = e^{i\phi_0}\begin{pmatrix} \cos\theta\, e^{i\phi_t} & \sin\theta\, e^{i\phi_r} \\ -\sin\theta\, e^{-i\phi_r} & \cos\theta\, e^{-i\phi_t} \end{pmatrix}, \tag{1.260}$$

where $\cos\theta$ and $\sin\theta$ are transmission and reflection amplitudes, respectively, and ϕ are phases.

The determinant of B is

$$\det(B) = e^{2i\phi_0} . \tag{1.261}$$

So, B is a unitary operator.

Hence, all the choices we can make for a photon gate (beam splitter) are equivalent. For a symmetric beam splitter ($\theta = \pi/4$) we get:

$$\frac{1}{\sqrt{2}}\begin{pmatrix}1 & i\\ i & 1\end{pmatrix} , \quad \frac{1}{\sqrt{2}}\begin{pmatrix}1 & -i\\ -i & 1\end{pmatrix} , \quad \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ -1 & 1\end{pmatrix} , \tag{1.262}$$

for $\{\phi_0 = 0, \phi_t = 0, \phi_r = \pi/2\}$, $\{\phi_0 = 0, \phi_t = 0, \phi_r = 3\pi/2\}$, and $\{\phi_0 = \phi_t = \phi_r = 0\}$, respectively.

We used the first choice to describe action of a beam splitter on photons that enters it in (1.5), (1.6), (1.7), (1.100), (1.208), and (1.210). It is shown in Figure 1.59. We apply a $\sqrt{\text{NOT}}'$ or a $\sqrt{\text{NOT}}$ gate to a photon that exhibits interference of the second (one photon) and the fourth (two photons) order, but we have to bear in mind that the function we obtain for these two gates will differ in some signs due to differences in phases. Compare (1.5) and (1.263).

When we calculate actions of a gate, we use it as any other Hilbert space operator. For example,

$$\sqrt{\text{NOT}}|0\rangle = \frac{1+i}{2}\begin{pmatrix}1 & -i\\ -i & 1\end{pmatrix}\begin{pmatrix}1\\ 0\end{pmatrix} = \frac{1+i}{2}\begin{pmatrix}1\\ -i\end{pmatrix} = \frac{1+i}{2}(|0\rangle - i|1\rangle) \tag{1.263}$$

$$\begin{aligned}\sqrt{\text{NOT}}\sqrt{\text{NOT}}|0\rangle &= \left(\frac{1+i}{2}\right)^2\begin{pmatrix}1 & -i\\ -i & 1\end{pmatrix}^2\begin{pmatrix}1\\ 0\end{pmatrix}\\ &= \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}\begin{pmatrix}1\\ 0\end{pmatrix} = \begin{pmatrix}0\\ 1\end{pmatrix} = |1\rangle.\end{aligned} \tag{1.264}$$

However, when we want to handle a gate as a device and incorporate it in a circuit, then we put its name in a box and connect it with its input and output by means of lines as shown in Figure 1.59.

The following gates are also special cases of the general beam splitter matrix B: NOT, Y, and Z for $\{\theta = \pi/2, \phi_0 = \pi/2, \phi_r = -\pi/2\}$, $\{\theta = \pi/2, \phi_0 = 0, \phi_r = -\pi/2\}$, and $\{\theta = 0, \phi_0 = \pi/2, \phi_t = -\pi/2\}$, respectively.

S and T are special cases of $P(\theta)$ for $\theta = \pi/2$ and $\theta = \pi/4$. The gate T also has the name $\pi/8$ because one can write it as

$$T = \begin{pmatrix}1 & 0\\ 0 & e^{i\pi/4}\end{pmatrix} = e^{i\pi/8}\begin{pmatrix}e^{-i\pi/8} & 0\\ 0 & e^{i\pi/8}\end{pmatrix} . \tag{1.265}$$

Gates X (NOT), Y, and Z are generators for the rotation gates:

$$R_x(\theta) = e^{\frac{-i\theta X}{2}} , \quad R_y(\theta) = e^{\frac{-i\theta Y}{2}} , \quad R_z(\theta) = e^{\frac{-i\theta Z}{2}} . \tag{1.266}$$

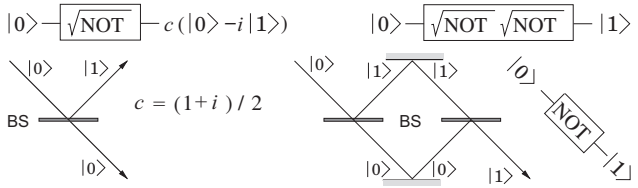


Figure 1.59 $\sqrt{\text{NOT}}$ and NOT gates. $|0\rangle$ and $|1\rangle$ denote paths (not polarizations). $\sqrt{\text{NOT}}$ is up to an overall phase equivalent to a matrix that describes an action of a beam splitter on photons that enter it. Note that all upper paths must have the same label. A repeated usage of two $\sqrt{\text{NOT}}$ gates is equivalent to a NOT gate and describes a Mach–Zehnder interferometer.

To prove this, we have to define the matrix exponential.

Definition 61 The *matrix exponential* is defined as follows [127]:

$$\exp(A) = e^A = \sum_{n=1}^{\infty} \frac{A^n}{n!} = \mathbb{1} + A + \frac{AA}{2!} + \frac{AAA}{3!} \dots \tag{1.267}$$

The series converges for any square matrix.

Taylor series for e^x, $\sin x$, and $\cos x$ are

$$\begin{aligned} e^x &= 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots \\ \sin x &= x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots \quad \cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots \quad \text{for all } x. \end{aligned} \tag{1.268}$$

Since $X^2 = Y^2 = Z^2 = I$, we have $X^{2k} = Y^{2k} = Z^{2k} = I$ and $X^{2k+1} = X$, $Y^{2k+1} = Y$, $Z^{2k+1} = Z$, for $k = 1, 2, 3, \dots$ This yields

$$\begin{aligned} & e^{\begin{pmatrix} 0 & i\theta/2 \\ -i\theta/2 & 0 \end{pmatrix}} = e^{\frac{-i\theta Y}{2}} \\ &= I - i\frac{\theta}{2} Y - \frac{\theta^2}{2!2^2} Y^2 + i\frac{\theta^3}{3!2^3} Y^3 + \frac{\theta^4}{4!2^4} Y^4 - i\frac{\theta^5}{5!2^5} Y^5 \\ &= \left(1 - \frac{\theta^2}{2!2^2} + \frac{\theta^4}{4!2^4} - \dots\right) I - i\left(\frac{\theta}{2} - \frac{\theta^3}{3!2^3} + \frac{\theta^5}{5!2^5}\right) Y \\ &= \cos\left(\frac{\theta}{2}\right) I - i \sin\left(\frac{\theta}{2}\right) Y = \begin{pmatrix} \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2}) \\ \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) \end{pmatrix}. \end{aligned} \tag{1.269}$$

We prove the other two expressions in (1.266), analogously. It is now easy to prove an even more general result: "For $A^2 = I$, the following holds: $e^{iAx} = \cos x\, I + i \sin x\, A$, where x is a real number."

In the end, it is easy to verify that all the above gates are unitary and therefore reversible. Quantum computation does not use and cannot use any gate that would be forwardable (be able to forward an unknown state) and not unitary.

Now, we can start to build quantum circuits which essentially consist of elements similar to those we considered in the previous sections. Simplistically, our goal is to preserve qubits in their states at each step of their calculation, repair their states whenever steps take too long, enable them to exchange states and results, and above all, to constantly change each other's states. This changing of states and bringing qubits into superposition and entanglement that we have seen in the previous sections is the very essence of quantum computation and without a parallel in the classical computation.

To be able to compute anything with the help of quantum gates, some qubits in a device, that is, a quantum circuit, should manipulate other qubits by means of the gates. We call the former ones *control qubits* and the latter ones *target qubits*. Control and target qubits can change their roles at another moment of computation.

These two features – reversibility and control-target gate design – have obvious similarities to the classical reversible gates as it was first realized by Feynman [95]. We presented these gates in Section 1.9 and now we shall partly apply the terminology and design of n-bit reversible gates to n-qubit quantum gates.

First, we have to find a way to express reversible control-target design by means of unitary matrices. The formalism of n-qubit gates exploits and extends two main features of the reversible gates:

a) reversibility – implemented as reversibility in time through the requirement that operators and matrices be unitary;
b) the control-target model of building circuits.

A target is a single qubit, and a gate either acts on it as described by a general 2×2 unitary matrix, provided that all the control qubits are in the state $|1\rangle$, or leaves it unchanged if they are not:

$$|\psi_j'\rangle = \begin{cases} \hat{U}|\psi_j\rangle & \text{for } |\psi_1\rangle = |1\rangle, \ldots, |\psi_{n-1}\rangle = |1\rangle \\ |\psi_j\rangle & \text{otherwise} , \end{cases} \tag{1.270}$$

where $|\psi_j\rangle$, $j = 1, \ldots, n$ can be equal to $|0\rangle$, to $|1\rangle$, or to $\alpha_j|0\rangle + \beta_j|1\rangle$. To match such a description of a gate, its matrix must consist of $2^n/2 - 1$ unit matrices of type 2×2 as shown in Figure 1.60.

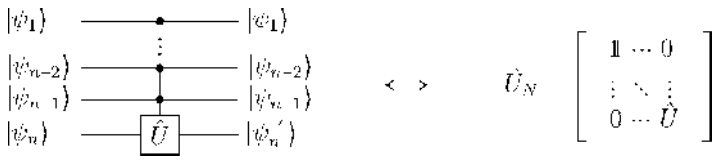


Figure 1.60 n-qubit circuit diagram is equivalent to an $N \times N$ matrix $\hat{U}_N$, where $N = 2^n$. $|\psi_1\rangle$, $\ldots$, $|\psi_{n-1}\rangle$ are control qubits and $|\psi_n\rangle$ is the target qubit. The diagonal has $n - 1$ unit matrices (2×2), $\mathbb{1}$, and a general two-dimensional 2×2 unitary matrix, $\hat{U}$.

It is instructive to compare (1.270) and Figure 1.60 with the following action of the classical CCNOT gate

$$\mathrm{CCNOT}a = \begin{pmatrix} 1 & & & & & & & \\ & 1 & & & & 0 & & \\ & & 1 & & & & & \\ & & & 1 & & & & \\ & & & & 1 & & & \\ & & & & & 1 & & \\ & & 0 & & & & 0 & 1 \\ & & & & & & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = b\,. \qquad (1.271)$$

This action of the reversible CCNOT clarifies the way the conditional in (1.270) works. If we write down n-qubit vectors as

$$|\underbrace{0\ldots0}_{2^n \text{ times}}\rangle = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad |\underbrace{0\ldots0}_{2^n-1 \text{ times}}1\rangle = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}, \ldots, \quad |\underbrace{1\ldots1}_{2^n \text{ times}}\rangle = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix}, \qquad (1.272)$$

we see that $\hat{U}_N$ will never change any of the first $2^n - 2$ vectors, and we obtain $\mathbb{1}$s. $\hat{U}_N$ can only change – through $2 \times 2\hat{U}$ – the last two vectors. Just as with the classical reversible gate given above by (1.271).

In classical reversible computation, this property is however not universal because, for example, the CNOT gate is not universal. Only three level gates may be universal, that is, can serve to express all other gates. In quantum computation, however, almost every two-level gate is universal, including of course the CNOT as the following theorem proves.

Theorem 62 "Two-level gates are universal"

Any $2^n \times 2^n$ unitary matrix $\hat{U}^{(2^n)}$ can be expressed as a product of matrices that act nontrivially on only two vector components.

Proof: Consider the following $N-1$ ($N = 2^n$) products, where the second matrix is always the same unitary matrix $\hat{U}^{(N)}$ and the first one is $\hat{\alpha}_j^{(N)}$ with $j = 2, \ldots, N$:

$$\begin{pmatrix} \alpha_{11} & 0 & \cdots & \alpha_{1j} & \cdots & 0 \\ 0 & \alpha_{22} & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ \alpha_{j1} & 0 & \cdots & \alpha_{jj} & \cdots & 0 \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & \alpha_{NN} \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1j} & \cdots & u_{1N} \\ u_{21} & u_{22} & \cdots & u_{2j} & \cdots & u_{2N} \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ u_{j1} & u_{j2} & \cdots & u_{jj} & \cdots & u_{NN} \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ u_{N1} & u_{N2} & \cdots & u_{Nj} & \cdots & u_{NN} \end{pmatrix}, \qquad (1.273)$$

where

$$\alpha_{11} = \frac{u_{jj}}{\sqrt{|u_{1j}|^2 + |u_{jj}|^2}}\,, \quad \alpha_{1j} = \frac{-u_{1j}}{\sqrt{|u_{1j}|^2 + |u_{jj}|^2}}\,,$$
$$\alpha_{j1} = \alpha_{1j}^*\,, \quad \alpha_{jj} = -\alpha_{11}^*$$
$$a_{22} = \ldots = a_{j-1j-1} = a_{j+1j+1} \ldots = a_{NN} = 1 \quad \text{for} \quad j \neq 2, N\,,$$
$$a_{33} = \ldots = a_{NN} = 1 \quad \text{for} \quad j = 2\,,$$
$$a_{22} = \ldots = a_{N-1N-1} = 1 \quad \text{for} \quad j = N\,. \tag{1.274}$$

The element in the jth column of the first row of the product matrix from (1.273) is therefore equal to zero. Also, since $\hat{U}^{(N)}$ is unitary and therefore $u_{ji} = u_{ij}^*$, the element in the jth row of the first column of this product matrix is equal to zero. The first matrices in (1.273), that is, $\hat{\alpha}_j^{(N)}$, are, given the condition (1.274), unitary: $\hat{\alpha}_j^{(N)} \alpha_j^{(N)\dagger} = \mathbb{1}$, where $\mathbb{1}$ is an $N \times N$ unit matrix. Since $U^{(N)}$ is also a unitary matrix, so is $\alpha_j^{(N)} U^{(N)}$.

Now, $U^{(N-1)} = \hat{\alpha}_{2^n}^{(N)} \hat{\alpha}_{2^n-1}^{(N)} \ldots \hat{\alpha}_2^{(N)} U^{(N)}$ is a matrix that has all the elements in the first row equal to zero, except the first one from the first column (which is one). This matrix must also be unitary since $\alpha_j^{(N)} U^{(N)}$ is unitary for any j, $j = 2, 3, \ldots, 2^n$. Thus, we get

$$U^{(N-1)} = \hat{\alpha}_{2^n}^{(N)} \hat{\alpha}_{2^n-1}^{(N)} \ldots \hat{\alpha}_2^{(N)} U^{(N)} = \begin{pmatrix} 1 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & u'_{22} & \cdots & u'_{2j} & \cdots & u'_{2N} \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ 0 & u'_{j2} & \cdots & u'_{jj} & \cdots & u'_{NN} \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ 0 & u'_{N2} & \cdots & u'_{Nj} & \cdots & u'_{NN} \end{pmatrix}. \tag{1.275}$$

We repeat our procedure on $U^{(N-2)}$, $U^{(N-3)}, \ldots$ to obtain

$$\begin{aligned} U^{(N-2)} &= \hat{\alpha}_{2^n}^{(N-1)} \ldots \hat{\alpha}_3^{(N)} U^{(N-1)} = \alpha_{2^n}^{(N-2)} \ldots \hat{\alpha}_2^{(N)} U^{(N)}\,, \\ U^{(N-3)} &= \hat{\alpha}_{2^n}^{(N-3)} \ldots \hat{\alpha}_2^{(N)} U^{(N)}\,, \\ &\vdots = \vdots \\ \mathbb{1} = U^{(1)} &= \hat{\alpha}_{2^n}^{(1)} \ldots \hat{\alpha}_2^{(N)} U^{(N)}\,, \end{aligned} \tag{1.276}$$

from which we get

$$U^{(N)} = \hat{\alpha}_{2^n}^{(1)\dagger} \ldots \hat{\alpha}_2^{(N)\dagger}\,. \tag{1.277}$$

There are $(N-1) + \cdots + 1 = N(N-1)/2 = 2^n(2^n-1)/2$ such α's. They all act nontrivially only on at most two rows of any N-row single column matrices, which proves the claim of the theorem. □

This result enables us to reduce any unitary gate to a chain of cascaded controlled gates, denoted controlled-controlled-...-U or controlled^n-U, and therefore we should engage the gates for each particular qubit within a specific time window. Applications, however, require that the number of qubit gates that are engaged within a time window be reduced to a minimum. Again, we can learn from the reversible gate approach. In Section 1.9, we said that there are many 3-bit universal reversible gates. We can show a similar result for quantum gates only simpler, 2-qubit ones. For example, we can substitute controlled^4-U with a sequence of Toffoli gates, as shown in Figure 1.61.

Here, we reach the limits of comparing quantum with reversible circuits and come to a point where quantum circuits essentially surpass reversible ones. Three-bit universal gates – for example, the Toffoli gate – are the smallest universal reversible gates [303], while – as follows from Theorem 62 and the elaboration below – almost any two-qubit quantum gate is universal.

To see this and at the same time to better understand the correspondence between quantum circuit diagrams and unitary operators, let us consider the example presented in Figure 1.62. In this figure, we take a controlled-controlled-$\hat{U}$ gate and express it by means of several CNOT and controlled-$\hat{V}$ gates, where $\hat{V}^2 = \hat{U}$ ($\hat{V}$ is, of course, also unitary). Note that the Toffoli gate is a special case of the controlled-controlled-$\hat{U}$ gate.

Every quantum gate presented in a quantum circuit diagram corresponds to a unitary matrix defining it. So, in Figure 1.62, matrix U_0 is given as

$$\begin{pmatrix} \mathbb{1} & & \cdots & 0 \\ & \mathbb{1} & & \vdots \\ \vdots & & \mathbb{1} & \\ 0 & \cdots & & U \end{pmatrix}, \tag{1.278}$$

where the $\mathbb{1}$s are 2×2 unit vectors and U is a 2×2 single-qubit unitary gate matrix.

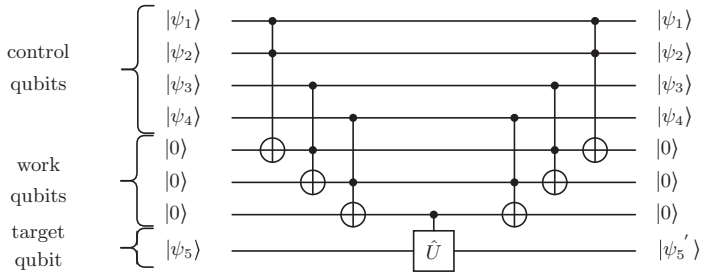


Figure 1.61 5-qubit controlled^4-U gate of the type shown in Figure 1.60 is implemented here by means of three Toffoli gates. When, for instance, $|\psi_i\rangle = |1\rangle$, $i = 1, \ldots, 4$, the first and second Toffoli gates change the states of the first and second *work qubits* (also called *ancillas*) from $|0\rangle$ to $|1\rangle$. The third Toffoli gate changes the third work qubit into a control qubit $|1\rangle$ for the target qubit gate U. In general, control qubits are in a superposition state.

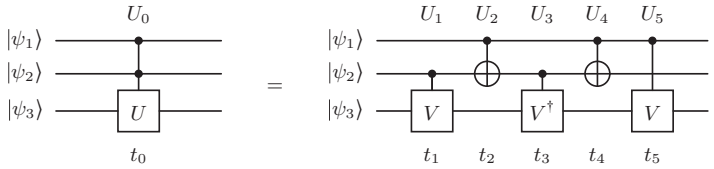


Figure 1.62 3-qubit gate controlled^2-U expressed by means of five 2-qubit gates. U_j, $j = 1, \ldots, 5$, are the matrices describing the gates. t_j, $j = 1, \ldots, 5$, are the corresponding times. $V^2 = U$.

To determine U_j, $j = 1, \ldots, 5$, we shall first reconsider the correspondence between qubit vectors and their column matrices from (1.272). The basis vectors of the first qubit are $|0000\rangle$ and $|0001\rangle$, and in the one-column matrix representation, these are the ones with 1s in the first and second row as in the first two matrices of (1.272). The state $|1\rangle$ of the second qubit we find in the third, fourth, seventh, and eighth ket ($|010\rangle$, $|011\rangle$, $|110\rangle$, and $|111\rangle$, respectively). This means that the one-column matrices have 1s in the third, fourth, seventh, and eighth row of the qubit matrices, and that we have to have one 2×2 matrix acting on the third and fourth row of the latter one-column matrices and another on the seventh, and eighth row. In a similar way, we determine the other U_j, $j = 2, \ldots, 5$ matrices so as to eventually have

$$U_1 = \begin{pmatrix} \mathbb{1} & & & 0 \\ & V & & \\ & & \mathbb{1} & \\ 0 & & & V \end{pmatrix}, \; U_2 = \begin{pmatrix} \mathbb{1} & & & 0 \\ & \mathbb{1} & & \\ & & 0 & \mathbb{1} \\ 0 & & \mathbb{1} & 0 \end{pmatrix}, \; U_3 = \begin{pmatrix} \mathbb{1} & & & 0 \\ & V^\dagger & & \\ & & \mathbb{1} & \\ 0 & & & V^\dagger \end{pmatrix},$$

$$U_4 = \begin{pmatrix} \mathbb{1} & & & 0 \\ & \mathbb{1} & & \\ & & 0 & \mathbb{1} \\ 0 & & \mathbb{1} & 0 \end{pmatrix}, \quad \text{and} \quad U_5 = \begin{pmatrix} \mathbb{1} & & & 0 \\ & \mathbb{1} & & \\ & & V & \\ 0 & & & V \end{pmatrix}. \tag{1.279}$$

Now, we simply have to check that

$$U_5 U_4 U_3 U_2 U_1 = U_0 \tag{1.280}$$

to prove that any quantum gate can be expressed by means of two-qubit controlled gates.

According to Theorem 62, any quantum gate can be expressed by means of CNOTs and single gates. CNOTs allow us to submit target qubits to control qubits, and single qubit gates allow us to rotate target states to desired ones. The following theorem narrows down their number.

Theorem 63 Vidal–Dawson [312]

An arbitrary unitary gate U from a $2 \otimes 2$-dim Hilbert space can be decomposed in terms of three (or less) CNOT gates and single-qubit unitary gates U_i, V_i, $i = 1, \ldots, 4$ as shown in Figure 1.63.

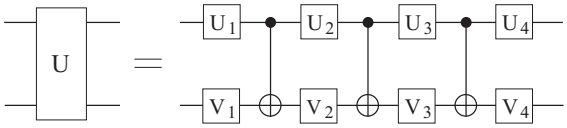


Figure 1.63 Arbitrary unitary gate expressed by three CNOT gates and single qubit gates.

Let us see how we can apply CNOT to entangle qubits since without an entangled state we certainly cannot perform quantum computation. And, after a rather involved calculation carried out in Section 1.13, in order to obtain the entanglement, we might appreciate the elegance of the formal approach.

Formally, we entangle states by means of a CNOT operator. As we know from Section 1.9, the CNOT will flip the target qubit only if the control qubit is in state $|1\rangle$. Matrices below are the CNOT and $|00\rangle$, $|01\rangle|10\rangle$, and $|11\rangle$ as given by (1.39) and (1.40) in Section 1.11, that is,

$$\begin{aligned}
\mathrm{CNOT}|00\rangle &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = |00\rangle\ , \quad \mathrm{CNOT}|01\rangle = |01\rangle\ , \\
\mathrm{CNOT}|10\rangle &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = |11\rangle\ , \quad \mathrm{CNOT}|11\rangle = |10\rangle\ .
\end{aligned} \tag{1.281}$$

Now, we can entangle our qubits. The most important thing is that the control qubit is in a superposition of two states. Its circuit is shown in Figure 1.64.

$$\mathrm{CNOT}(\alpha|0\rangle + \beta|1\rangle)|0\rangle = \alpha\mathrm{CNOT}|0\rangle|0\rangle + \beta\mathrm{CNOT}|1\rangle|0\rangle = \alpha|00\rangle + \beta|11\rangle\ . \tag{1.282}$$

An entangled qubit will tend to decohere fast. Fortunately, we can fight decoherence and sustain the entangled states as described in the next section.

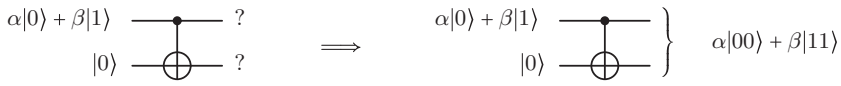


Figure 1.64 *Entanglement circuit*. Entanglement is special, even for a quantum circuit. Entangled qubits share lines. This is denoted by a curly bracket which signifies that neither of the qubits is in a definite state of its own.

1.19
Self-Sustaining Qubits: Quantum Error Correction

At the dawn of the quantum information era, the no-cloning Theorem 52 in Section 1.16 and the short coherence time of quantum states were perceived as the main obstacles for a successful handling of the states and ultimately for a quantum computation. Then, in the late nineties, a quantum error correction procedures came to the rescue. [59, 288, 297]

Quantum error correction is a method of recovering the initial states of original qubits by means of additional ancillary ones. The theory of quantum error correction adapts classical error correction theory to quantum states. Therefore, we shall briefly consider the classical theory first. It is used in classical computation algorithms and classical digital computer software.

Error correction schemes are based on some kind of preparation (encoding) of our bits. One of the simplest such encodings for "classical Alice" is to add the so-called *parity bit* to all messages she sends to Bob. Here, *parity* is the quality of being odd or even. Alice and Bob agree that she should choose the parity bit 0 if the number of 1s in the message is even, and 1 if the number is odd. So if she wants to send 1001, she should add the parity bit 0 (because the number of 1s in 1001 is even) and send 10010 instead of 1001; if she wants to send 1101, she should add the parity bit 1 (because the number of 1s in 1101 is odd) and send 11011. If a *bit-flip* occurs and Bob receives, say 11111 in the last message, he would calculate $1 \oplus 1 \oplus 1 \oplus 1 = 0$ and would know that something went wrong because the parity bit 1 indicated that the parity of the message should have been odd.

We can formalize the procedure as follows. Alice encodes her messages by means of the following 4×5 matrix:

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

$$\begin{pmatrix} 1 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 \end{pmatrix}. \tag{1.283}$$

Note that for calculating the last column, we used XOR: $1 \oplus 1 \oplus 0 \oplus 1 = 1$. So, if there are no errors, Bob will always get 0 for the XOR value of encoded words: $1 \oplus 0 \oplus 0 \oplus 1 \oplus 0 = 0$, $1 \oplus 1 \oplus 0 \oplus 1 \oplus 1 = 0$. If there is an error, say 1111 as above, he will get $1 \oplus 1 \oplus 1 \oplus 1 \oplus 1 = 1$, which will detect it.

Of course, if two flips occur, Bob will not detect any error. Also, when Bob detects an error, he cannot tell which bit flipped. There are many other classical codes that can enable him to spot *and* correct the flipped bit. The first one is "brute force," that is, just to repeat the same message three times in a row (two times would not suffice because Bob would not be able to not tell which one was right and which wrong).

The information rate of brute force is very low: 1/3. There are, however, other codes that have a much higher information rate as well as better error detection and correction ability, which are the reasons why the error correction theories are so important. Here, we shall consider one of the latter codes, the *Hamming scheme*, which we will also use for a quantum error correction scheme.

We start with some definitions:

- A *codeword* is a word over $\{0, 1\}$. The number of its bits is n.
- A *code* is a set C of codewords.
- An *error* is a change of bits on the way from Alice to Bob.
- *Data bits* make a message within a codeword Alice sends to Bob. Their number is d. We denote a codeword containing d data bits by (n, d).
- *Parity bits* are check bits within a codeword Alice sends to Bob. Parity bits enable Bob's error correction. Their number is p.
- The *Hamming distance between two words* is the number of bit positions in which the words disagree.
- The *Hamming distance of a code C*, denoted $D(C)$, is the minimum distance of two codewords in a code. A code (n, d) having a distance $D(C)$ will be denoted by $[n, d, D]$.
- A *Hamming code* is a code having $D(C) = 3$.
- The *information rate* of a code C of length n over alphabet F with size $|F| = q$ is $\log_q |C|/n$.

One can prove the following results [186]:

- The number of parity check bits required for each message is given by the following Hamming rule:

$$d + p + 1 \leq 2^p \ . \tag{1.284}$$

- A code with distance D is $(D-1)$ error detecting and $(D-1)/2$ error correcting. Hence, the Hamming code allows us to detect two errors and correct one of them.
- For $D = 3$, only the following (n, d) codes are possible:

$$(n, d) = (2^i - 1, 2^i - 1 - i) \ ,$$

  where $i \geq 3$ is an integer. So, for instance, we can have the following Hamming codes: (7,4), (15,11), (31,26), and so on.
- An (n, d) code is a d-dimensional subspace of F^n, whose size is $q = |F|$. Since $|C| = q^d$, the information rate of the code is $\log_q |C|/n = d/n$.

To see how Bob can correct an error using the Hamming scheme, let us consider the Hamming code (7,4), for which the information rate is $d/n = 4/7 \approx 0.57$, that is, about 1.7 times higher than the previously mentioned triple sending. It follows from (1.284) that $p = 3$. One can easily check that among $2^7 = 128$ possible 7-bit

codewords, there are only 16 valid ones (with $D = 3$). These can be generated by, for example, the following *generation matrix*:

$$G = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} , \tag{1.285}$$

with the help of which we encode message m so as to get the codeword $c = m \cdot G$. Note that in the multiplication of matrices, we make use of XOR, that is, of adding their elements modulo 2; the dot "$\cdot$" refers to this. For 4-bit messages, Alice gets the codewords in Table 1.11 below and sends them to Bob.

Bob receives the codeword c, and using the *check matrix* H, defined below in (1.286), he gets $H \cdot c^{\mathrm{T}} = s$. In the case of no errors in the above codeword, he will always get $s = 0$, as one can easily check, and then he recovers the original message according to Table 1.11. If, for example, the codeword 0011001 had the sixth bit flipped to 1 and Bob received the string 0011011, he would obtain

$$H \cdot c^{\mathrm{T}} = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} = s = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} . \tag{1.286}$$

This s is called the *syndrome*, and it tells us which column we should look at in H. In this case, it is the sixth column, meaning that the sixth bit in the string has been flipped. (Syndrome s in (1.286) is written in the binary notation: $1 = (000)^{\mathrm{T}}, \ldots, 6 = (110)^{\mathrm{T}}, 7 = (111)^{\mathrm{T}}$.) So, after correcting it, Bob gets the original codeword 0011001. Note that H is a submatrix of G: it equals the second through fourth rows of G. Also, note that the codeword 0011001 is a unique codeword with distance 1 from string 0011011. This is why we need codewords with distances of at least 3 from each other. The following general results hold:

- For one error in the ith bit, the syndrome s is the ith column of H.
- Each error of *weight* (number of erroneous bits) up to $(D - 1)/2$ has a unique syndrome.

Table 1.11 Unique Hamming codewords $c = m \cdot G$ (top lines) for messages m (bottom lines) that Alice sends to Bob.

0000	0001	0010	0011	0100	0101	0110	0111
0000000	1010101	0110011	1100110	0001111	1011010	0111100	1101001
1000	1001	1010	1011	1100	1101	1110	1111
1111111	0101010	1001100	0011001	1110000	0100101	1000011	0010110

We can look at the error correction in the following way. The codeword u Alice sent is affected by noise in the communication channel so that it changes into $u' = u + e$, where e is the error caused by the noise and where the addition is modulo 2. In the above example, $u = 0011001$, $u' = 0011011$, and $e = 0000010$. From u', Bob can uniquely recover u using the code C, but he needs not learn how the word actually reads because

$$H \cdot u' = H \cdot (u + e) = H \cdot u + H \cdot e = H \cdot e = s \,, \tag{1.287}$$

and this means that he can learn the syndrome without ever learning the word. (For the code [7,4,3], there are 16 codewords and only 7 syndromes.) Of course, in the classical case, he can always look at Table 1.11, but in the quantum case, this amounts to correcting a quantum state without disturbing it, and this outcome is what we are looking for in quantum error correction and quantum cryptography [299].

While classical error correction protocols encode bits that Alice transmits to Bob by means of additional parity bits, quantum protocols must be able to encode superposed qubits by entangling them with additional qubits because we cannot make more replicas of a superposed state (see the no-cloning theorem (Theorem 52) in Section 1.16). Superpositions are what we essentially have both, in quantum computation and in quantum cryptography.

The errors that can occur in transmission (in quantum cryptography as well as in quantum computation) of an arbitrary qubit state

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \,, \quad \text{where} \quad |\alpha|^2 + |\beta|^2 = 1 \,, \tag{1.288}$$

are a *bit-flip*,

$$X|\psi\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} |\psi\rangle = \beta|0\rangle + \alpha|1\rangle \,, \tag{1.289}$$

a *phase shift*,

$$Z|\psi\rangle = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} |\psi\rangle = \alpha|0\rangle - \beta|1\rangle \,, \tag{1.290}$$

or both:

$$Y|\psi\rangle = \mathrm{ZX}|\psi\rangle = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} |\psi\rangle = \beta|0\rangle - \alpha|1\rangle \,, \tag{1.291}$$

where $X = \hat{\sigma}_x$, $Z = \hat{\sigma}_z$, and $Y = i\hat{\sigma}_x$ (cf. (1.41)).

Let us first consider bit-flip correction. If we encode a single qubit in the state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ by means of entangled qubits whose sequence corresponds to classical words, we will be able to use classical error correction applied not to the original superposition, but to a superposition of such quantum codewords. And this is exactly what Steane did with the Hamming code [296]. A similar approach was taken by Calderbank and Shor [59]. Hence, the name *CSS codes*.

The idea put forward in [296, 297] was to generate Hamming codewords using quantum gates so as to enable error detection and correction analogous to that given by (1.286). The encoding is shown in Figure 1.65.

In this approach, we do not have four data bits that we encode by means of additional three bits, as in the classical Hamming code, but we will still be able to use this classical code for the error correction. Instead of four bits, we encode just one qubit in an unknown state by entangling it with six other qubits, all initially in state $|0\rangle$. The qubit we encode is $|\psi\rangle$ in Figure 1.65, given by (1.288). As in Figure 1.64, when we apply the control qubit $|\psi\rangle$ (third qubit) to the fifth and sixth qubits ($|0\rangle_5$ and $|0\rangle_6$) as its targets, we entangle them:

$$|\psi_{356}\rangle = \alpha|0\rangle_3|0\rangle_5|0\rangle_6 + \beta|1\rangle_3|1\rangle_5|1\rangle_6 \,. \tag{1.292}$$

Consequently, from this point on, the lines that originally represented the third, fifth, and sixth qubit now represent one and the same state. The next time point corresponds to $|\psi_1\rangle = |0\rangle_1 + |1\rangle_1$, acting as a control qubit on the third, fifth, and sixth qubits. The third and fifth qubits are entangled with the sixth qubit, and so it is brought in as well. The resulting state is an entanglement of all five qubits:

$$\begin{aligned}|\psi_{13567}\rangle = {} & \alpha|0\rangle_1|0\rangle_3|0\rangle_5|0\rangle_6|0\rangle_7 + \beta|0\rangle_1|1\rangle_3|1\rangle_5|1\rangle_6|0\rangle_7 \\ & + \alpha|1\rangle_1|1\rangle_3|1\rangle_5|0\rangle_6|1\rangle_7 + \beta|1\rangle_1|0\rangle_3|0\rangle_5|1\rangle_6|1\rangle_7 \,.\end{aligned} \tag{1.293}$$

At the next level, $|\psi_2\rangle = |0\rangle_2 + |1\rangle_2$ entangles all but the fourth qubit:

$$\begin{aligned}|\psi_{123567}\rangle = {} & \alpha|0\rangle_1|0\rangle_2|0\rangle_3|0\rangle_5|0\rangle_6|0\rangle_7 + \beta|0\rangle_1|0\rangle_2|1\rangle_3|1\rangle_5|1\rangle_6|0\rangle_7 \\ & + \alpha|1\rangle_1|0\rangle_2|1\rangle_3|1\rangle_5|0\rangle_6|1\rangle_7 + \beta|1\rangle_1|0\rangle_2|0\rangle_3|0\rangle_5|1\rangle_6|1\rangle_7 \\ & + \alpha|0\rangle_1|1\rangle_2|1\rangle_3|0\rangle_5|1\rangle_6|1\rangle_7 + \beta|0\rangle_1|1\rangle_2|0\rangle_3|1\rangle_5|0\rangle_6|1\rangle_7 \\ & + \alpha|1\rangle_1|1\rangle_2|0\rangle_3|1\rangle_5|1\rangle_6|0\rangle_7 + \beta|1\rangle_1|1\rangle_2|1\rangle_3|0\rangle_5|0\rangle_6|0\rangle_7 \,.\end{aligned} \tag{1.294}$$

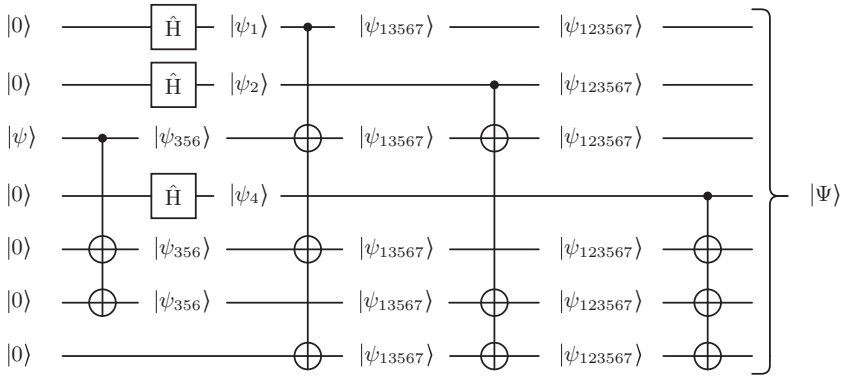


Figure 1.65 *Hamming codewords.* A 7-qubit encoding of $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ into $|\Psi\rangle = \alpha|0\rangle_{1-7} + \beta|1\rangle_{1-7}$ according to [297]. H is the Hadamard matrix (see Table 1.10).

In the end, $|\psi_4\rangle = |0\rangle_4 + |1\rangle_4$ gives

$$\begin{aligned}|\Psi\rangle &= \alpha(|0000000\rangle + |1010101\rangle + |0110011\rangle + |1100110\rangle \\ &\quad + |0001111\rangle + |1011010\rangle + |0111100\rangle + |1101001\rangle) \\ &\quad + \beta(|1111111\rangle + |0101010\rangle + |1001100\rangle + |0011001\rangle \\ &\quad + |1110000\rangle + |0100101\rangle + |1000011\rangle + |0010110\rangle) \\ &= \alpha|0\rangle_{1-7} + \beta|1\rangle_{1-7}\,, \end{aligned} \tag{1.295}$$

and these are nothing but the Hamming codewords from Table 1.11.

Hence, for correcting errors, we can use classical Hamming theory if we assume that we will mostly have only one-flip errors, that is, that the probability of having two such errors within a transmission of one codeword is negligible. The check matrix H given by (1.286) then tells us how to design the error-correcting scheme. In the matrix multiplication of codewords by H modulo 2, 1s in H turn 1 s in a codeword into 0s and 0s into 1s. In other words, they behave like control qubits. Recalling (1.287), we see that their targets can be three syndrome qubits $|0\rangle$, that is, three additional check qubits – usually called *ancillas*. In this way, we get around the lack of four data bits we would have in the classical Hamming code. How this can be carried out is shown in Figure 1.66.

The rows in H given by (1.286) determine the control qubits acting on each of three ancillas in Figure 1.66. Checking all terms in (1.295), we see that the parity of all bits corresponding to the first, second, and third row of H is always even, and when an even number of flips occurs, each ancilla $|0\rangle$ will be flipped twice and detectors D1–3 will find them in the state $|0\rangle$. If an odd number of flips, that is, an error in transmission, occurs, one, two, or all three detectors will detect the state $|1\rangle$. According to the correspondence with the columns in H, the flipped qubit state will be corrected by means of X. For example, if D1 and D2 detected the state $|1\rangle$,

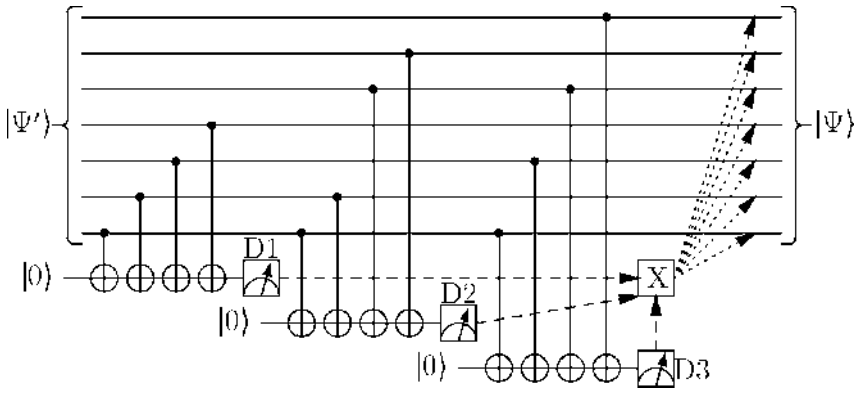


Figure 1.66 A 7-qubit error-correcting code. The control qubits are chosen so as to correspond to 1 s in H, as given by (1.286). Detectors D1–3 measure the ancillas and apply flip-correcting X to a qubit found to have suffered a bit-flip in transmission. The input state $|\Psi'\rangle$ is $|\Psi\rangle$ (from Figure 1.65) with possibly a flipped qubit state.

the state of the sixth qubit will be flipped back by means of X because $[110]^{\mathrm{T}}$ is the sixth column in H (1.286).

After the correction has been carried out and the codestate $|\Psi\rangle$ restored, Bob has to decode $|\Psi\rangle$ so as to obtain $|\psi\rangle$ sent by Alice. (We stress again here that this process is the same whether Alice and Bob are simply parts or stages of quantum computation or parties in quantum communication.) Bob decodes the message by reversing the procedure given in Figure 1.65 while substituting Z for H. To clarify the reversed procedure, let us just consider its last step, restoring $|\psi\rangle$ from $|\psi_{356}\rangle$, as given in Figure 1.67.

We can say that the state $|\psi_{356}\rangle$ given by (1.292) "acts on itself" in the following sense. First, within each product state (codeword) of the state $|\psi_{356}\rangle$, the third qubit state acts as a control qubit on the sixth qubit state. Hence, $|0\rangle_3$ from $|0\rangle_3|0\rangle_5|0\rangle_6$ acts on $|0\rangle_6$ from this product and leaves it unchanged, while $|1\rangle_3$ from $|1\rangle_3|1\rangle_5|1\rangle_6$ flips $|1\rangle_6$ into $|0\rangle_6$. As a consequence, the sixth qubit disentangles from the third and the fifth one. In the next step, the fifth qubit disentangles from the third one and we recover the original state $|\psi\rangle$ of the latter qubit.

To see how we can correct the phase shift in Steane's 7-qubit code, let us start with (1.295):

$$|\Psi\rangle = \alpha|0\rangle_{1-7} + \beta|1\rangle_{1-7} \,. \tag{1.296}$$

Notice that by comparing terms in (1.295), we can see that $|1111111\rangle = \overline{X}|0000000\rangle$, $|0101010\rangle = \overline{X}|1010101\rangle$, and so on, where $\overline{X} = X \otimes \cdots \otimes X$, and where $X = \mathrm{NOT} = \sigma_x$. Therefore, $|1\rangle_{1-7} = \overline{X}|0\rangle_{1-7}$.

Using the Hadamard gate (Table 1.10), we can introduce the following new basis for each qubit. Here (till the end of this section) we shall denote the Hadamard operator by means of an upright font (H) so as to distinguish it from the check matrix (H).

$$\begin{aligned} |\overline{0}\rangle_i &= \mathrm{H}|0\rangle_i = \frac{1}{\sqrt{2}}(|0\rangle_i + |1\rangle_i) \\ |\overline{1}\rangle_i &= \mathrm{H}|1\rangle_i = \frac{1}{\sqrt{2}}(|0\rangle_i - |1\rangle_i) \,, \quad i = 1, \ldots, 7 \,, \end{aligned} \tag{1.297}$$

where we stripped the hat "ˆ" from H to ease the notation.

An attractive feature of our code is that upon substituting $|0\rangle_i = (|\overline{0}\rangle_i + |\overline{1}\rangle_i)/\sqrt{2}$ and $|1\rangle_i = (|\overline{0}\rangle_i - |\overline{1}\rangle_i)/\sqrt{2}$, $i = 1, \ldots, 7$ into (1.295), that is, into (1.296), we get

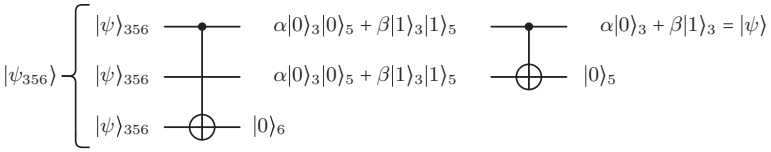


Figure 1.67 The last two levels of decoding a superposition of 7-qubit codewords $|\Psi\rangle$ encoded in Figure 1.65 and corrected in Figure 1.66.

(the process is straightforward but tedious):

$$|0\rangle_{1-7} = \frac{1}{\sqrt{2}}(|\overline{0}\rangle_{1-7} + |\overline{1}\rangle_{1-7})$$
$$|1\rangle_{1-7} = \frac{1}{\sqrt{2}}(|\overline{0}\rangle_{1-7} - |\overline{1}\rangle_{1-7}) \,. \tag{1.298}$$

This outcome means that the new basis only contains those states that are in the Hamming code – actually, the same states that the old basis contains. In the new basis, $|\Psi\rangle$ reads

$$|\Psi\rangle = \frac{1}{\sqrt{2}}[(\alpha+\beta)|\overline{0}\rangle_{1-7} + (\alpha-\beta)|\overline{1}\rangle_{1-7}] \,. \tag{1.299}$$

Now, a phase shift in the old basis

$$|\Psi\rangle = \alpha|0\rangle_{1-7} - \beta|1\rangle_{1-7} \tag{1.300}$$

transforms in the new basis to

$$|\Psi\rangle = \frac{1}{\sqrt{2}}[(\alpha-\beta)|\overline{0}\rangle_{1-7} + (\alpha+\beta)|\overline{1}\rangle_{1-7}] \,, \tag{1.301}$$

which is therefore nothing but a bit-flip (by definition – cf. (1.289) in the latter basis.

Alternatively, we can use

$$\alpha|0\rangle_{1-7} - \beta|1\rangle_{1-7} = \overline{Z}\,|\Psi\rangle = \overline{\mathrm{H}X\mathrm{H}}\,|\Psi\rangle \,, \tag{1.302}$$

where $\overline{\mathrm{H}} = \mathrm{H}\otimes\cdots\otimes\mathrm{H}$ and $\overline{Z} = Z\otimes\cdots\otimes Z$, since we can show that the following transformations hold:

$$\overline{\mathrm{H}}|0\rangle_{1-7} = \frac{1}{\sqrt{2}}(|0\rangle_{1-7}+|1\rangle_{1-7}), \quad \overline{\mathrm{H}}|1\rangle_{1-7} = \frac{1}{\sqrt{2}}(|0\rangle_{1-7}-|1\rangle_{1-7}) \,. \tag{1.303}$$

Equations (1.299) and (1.302) mean not only that we can reduce the phase shift correction to a bit-flip correction with the same error correcting code, but also that we can completely separate the corrections and do them in sequence, as shown in Figure 1.68.

The error correction scheme presented above enables us not only to correct errors in transmission, but also to correct malfunctioning of the gates (CNOT) used

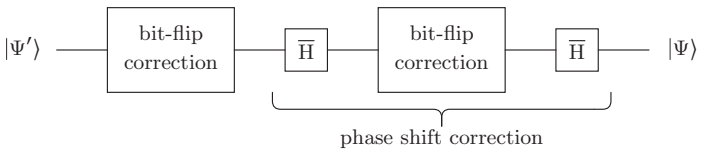


Figure 1.68 The 7-qubit correction code carries out bit-flip and phase-shift corrections with the same correction circuit (Figure 1.66) in two bases.

to encode the message (cf. Figure 1.65). Such quantum computation, where we can correct malfunctioning of circuits, is called *fault-tolerant* computation [299].

For quantum computation and transmission with more than one error per code, classical codes with larger Hamming distances capable of correcting multiple errors [186] can be applied.

1.20 Flying Qubits Connecting Quantum Chips and Computers: Quantum Repeaters

When teleportation devices such as the one shown in Figure 1.78 are used for communicating unknown states from one point in space to another, such devices are called *quantum repeaters*.

Every block B_1, B_2, ..., B_{n-1}, shown in Figure 1.69 represents a Bell state discriminator lie the one shown in Figures 1.47 or 1.48.

As we already emphasized (below (1.287)), according to Theorem 52 (Section 1.16), we cannot copy a quantum state and therefore we cannot amplify a quantum signal, but we can teleport the signal through fibers. The question is whether we can cover a greater distance with the same losses using repeaters than with a single fiber.

Fiber couplers have also reached a high efficiency recently and have been used for building concatenated beam splitters and concatenated Mach–Zehnder interferometers for single photons and for two-photon interferometry [126, 157, 170, 328].

The repeaters must use photons on demand because the photons must arrive to blocks B_i simultaneously to get entangled. We have seen in Section 1.16 that a probabilistic generation of entangled photons by means of double and triple down-conversion have been proposed by M. Pavičić [237] (see Figure 1.50) and C. Śliwa, and K. Banaszek [292], respectively (see also [150, 151]). The latter proposal has been implemented by S. Barz, G. Cronenberg, A. Zeilinger, and P. Walther [16] in Vienna, Austria and by C. Wagenknecht, C.-M. Li, A. Reingruber, X.-H. Bao, A. Goebel, Y.-A. Chen, Q. Zhang, K. Chen, and J.-W. Pan [317] in Hefei, China. However, these schemes cannot give photon pairs on demand (their success probability is too low) and therefore they are not likely candidates for quantum repeaters.

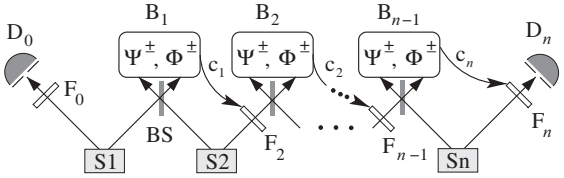


Figure 1.69 *Quantum repeater*. A schematic that takes into account (1.228) for restoring a state by means of detected Bell states $|\Phi^{\pm}\rangle$, $|\Phi^{\pm}\rangle$. S1, ..., Sn are sources of entangled photons in the Bell states. Blocks B_1, ..., B_{n-1} are blocks for a discrimination of all four Bell states. The c's are classical channels for restoring the state imposed on the first photon by the filter F_0.

The problem arise from the fact that we must have a comparatively small probability of obtaining an entangled pair of photons already in a single crystal because we have attenuated the intensity of the pump beam to avoid generation of more than one pair at the exit. For three crystals, the probabilities multiply and in the aforementioned experiments, the probability of heralding an entangled pair on demand within a time window of 1 ns is about 0.0001%. That would mean that for 100% (which would be required for a series of repeaters; not much lower anyhow for many concatenated repeaters), we should use a million such down-conversions with a million of optical switches for each repeater and that, of course, cannot be implemented [150]. The result cannot be improved because the down-conversion process is a probabilistic process and increasing the intensity of the pump beam would generate unwanted multiple pairs within the time window.

Another candidate for a source of entangled photons on demand that we can apply in repeaters are quantum dots. C.L. Salter, R.M. Stevenson, I. Farrer, C.A. Nicoll, D.A. Ritchie, and A.J. Shields [272] have devised and implemented a particularly promising "electrically driven source of entangled photon pairs consisting of a quantum dot embedded in a semiconductor light-emitting diode (LED) structure." They showed that the device emits entangled photon pairs under DC and AC injection, the latter achieving an entanglement fidelity of up to 0.82 (maximally entangled photons have the entanglement fidelity 1). They also estimated that with improved setup and equipment, they could reach the fidelity of 0.93.

However, the efficiency of the "entangled-light-emitting diode" (ELED) is even lower than that of the aforementioned down-converted source, but since the entangled pairs from an ELED are genuine single pairs, and therefore deterministic, an increase in efficiency only depends on physical features of quantum dots. The features impose a very demanding working conditions on an ELED: only one in 100 quantum dots can be successfully produced and quantum dots for the time being do not work at temperatures higher than 5 K [63].

Therefore, we need to develop efficient photon pairs on demand to obtain efficient repeaters. With increased efficiency, we can multiplex many sources and use the one which offers a photon pair within a required time window. However, today it would mean multiplexing 100 000 sources for each repeater and that is clearly impossible.

Another, apparently more promising approach, includes quantum repeaters based on atomic ensembles and linear optics of their collectively emitted and absorbed photons [273]. We will come back to them in Section 3.1.4.

This rounds up the bits and pieces of our would-be quantum network. Their realistic implementations are still far from fitting to each other smoothly as we will see in the next chapter. But, there is a part of the quantum network that has already developed far better than the others and it entered a stage which enables its commercial implementation. We are going to elaborate more on quantum cryptography in the following sections.

1.21
Why Classical Cryptography Cannot Keep Secrets for Long ...

To see how the classical encryption works, we start with a primer of classical bit representation of a decimal number because this is essential for understanding the classical encryption problems. The binary representation of a decimal number N is given by a binary digit string

$$N_2 = \alpha_{n-1}\alpha_{n-2}\ldots\alpha_1\alpha_0\,, \tag{1.304}$$

where α_i, $i = 0, \ldots, n-1$ are determined from the following equation:

$$N_2 = \alpha_{n-1}2^{n-1} + \alpha_{n-2}2^{n-2} + \cdots + \alpha_1 2^1 + \alpha_0 2^0 = \sum_{i=0}^{n-1} \alpha_i 2^i\,. \tag{1.305}$$

So, to obtain a binary representation of, for example, number 36, we divide 36 by 2 and get 18 and the remainder is equal to 0 $\Rightarrow$ the last binary digit is 0; $18/2 = 9$, the remainder is 0 $\Rightarrow$ the one but last digit is 0, and hence, 00; $9/2 = 4$, the remainder is 1 $\Rightarrow$ 100; $4/2 = 2 \Rightarrow$ 0100; $2/2 = 1 \Rightarrow$ 00100; $1/2 = 0$, the remainder is 1 $\Rightarrow$ 100100. Hence, the binary representation of 36 is 100100, that is, a number with six digits and this number of digits is what makes a lot of tasks on classical computers intractable. Conversion from a binary to decimal representation runs as follows: $2^2 + 2^5 = 36$.

All tasks on classical computers from writing a text, over massive computing, to watching videos are reduced to handling binary numbers consisting of bits – 0 s and 1 s – and bits are the only input the classical computer hardware can work with. A computer or any digital device handle binary numbers by means of classical logic gates (Section 1.4), that is, transistors. They manipulate strings of bits and carry out arithmetic operations on them. A combination of gates is required to manipulate these strings and to carry out addition. Other operations can be reduced to addition. We add single bits as follows $0 + 0 = 0$, $0 + 1 = 1 + 0 = 1$. Already, $1 + 1$ requires a 2-bit string: $1 + 1 = 10$. We get the string using the so-called *half adder* shown in Figure 1.70a, where the sum $\boldsymbol{S} = \boldsymbol{A} \oplus \boldsymbol{B}$ in the last case is 0 and the *carry* $\boldsymbol{C}_{\text{out}} = \boldsymbol{AB}$ is 1.[33] For bigger numbers, we need to reuse the carry. The *full adder* shown in Figure 1.70b serves this purpose.

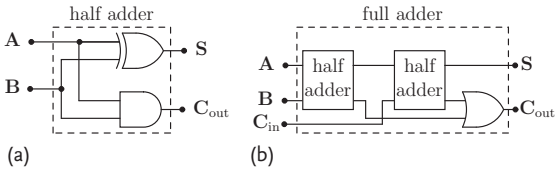


Figure 1.70 (a) $\boldsymbol{S} = \boldsymbol{A} \oplus \boldsymbol{B}$, $\boldsymbol{C}_{\text{out}} = \boldsymbol{AB}$; (b) $\boldsymbol{S} = (\boldsymbol{A} \oplus \boldsymbol{B}) \oplus \boldsymbol{C}_{\text{in}}$, $\boldsymbol{C}_{\text{out}} = (\boldsymbol{A} \oplus \boldsymbol{B})\boldsymbol{C}_{\text{in}} + \boldsymbol{AB}$.

33) So, we have the following strings: $\{\boldsymbol{C}_{\text{out}}\}\boldsymbol{S} = 00$ (for $0 + 0$), 01 (for $0 + 1 = 1 + 0$), and 10 (for $1 + 1$).

Full adders are combined in blocks so as to form *binary adders*. The binary adder shown in Figure 1.71 is an 8-bit (1-byte) device, and we see that it is capable of dealing with any number whose sum does not exceed 255. Otherwise, we get an *overflow* ($C_{fo} = 1$ in Figure 1.71).

Figures 1.70 and 1.71 illustrate why binary processing can take so much more time than "physical" processing. A transistor is a switch with a delay between an input voltage change and the response on its output. This time it is called *gate delay*. Delay times in transistors can be brought down to less than 1 ns nowadays, but let us use 1 ns in a PC as an illustration. This time may be slowed down to about 2 ns for some gates (typically, NAND is one of the fastest and XOR is the slowest). Since a full adder is a cascade of two half adders, the former might yield a result within 4 to 8 ns because the time varies with inputs (1s or 0s). This time increases linearly with the bit length because each full adder within binary adders has to wait for the carry from a previous stage (see Figure 1.71) to output a steady-state result. Thus, solutions of problems for which complexity grow exponentially with time require both an exponential reduction of gate delays and an exponential increase of the number of transistors. We have already seen that further reductions in the size of transistors, and thereby increases in their numbers within processors, will soon reach their limits. Attempts to further reduce gate delays will also hit its physical limits.

We can now go to a classical cryptography primer.

There are two main types of classical cryptographic algorithms:

- Private (secret) key (symmetric) algorithms – single key for both encryption and decryption;
- Public key (asymmetric) algorithms – one key for encryption and another for decryption.

Private key cryptography is *the* classical cryptography known for four thousand years and provides ways of encrypting messages by a sender – *Alice*, with a key and decrypting the messages by a receiver – *Bob* – with the same key. This kind of cryptography is widely used on the Internet today. One example is the Data Encryption Standard (DES), which is not absolutely secure. Another is the *Vernam ciphers* (also called *one-time pad*), which are uncrackable, but too slow for all but the most

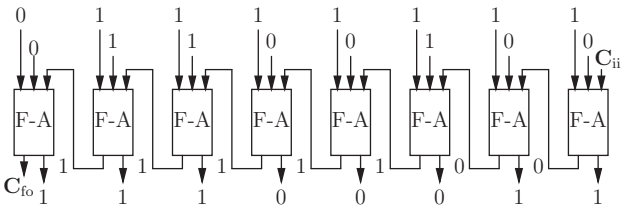


Figure 1.71 Eight-bit binary adder. F-A are full adders (Figure 1.70b). C_{ii} and C_{fo} are initial input and final output carry bits, respectively. The addition 127 + 100 = 227 is shown in the binary representation: 01111111 + 01100100 = 11100011; $C_{ii} = C_{fo} = 0$.

sensitive transmissions. DES makes use a of 56-bit private key operating on 64-bit binary blocks. In Table 1.12, we give an example of a one-time pad cryptosystem using a smaller base-64 representation of basic ASCII characters.

Let us look at the example shown in Table 1.12. Alice wants to encrypt a word of five characters and send it to Bob. Each character in base-64 is represented by a 6-bit string, and therefore any such word is represented by a 30-bit binary sequence. To encrypt it, Alice has to produce a key and a function that can encrypt the word using the key and decrypt it using the same key again. As we will see below, the best key is a completely random 30-bit binary sequence.[34] An appropriate function is XOR, $\oplus$ (see Figure 1.2), since $(A \oplus B) \oplus B = A$. Next, she has to send the key to Bob via a very trusted carrier, if not in person. If she decided to send him the word *qubit*, she would obtain the encrypted message *Y4tb3* and could send it through any public channel to Bob, who can decrypt it using the key and the XOR function as shown in Table 1.12.

The main disadvantages of this system are that Alice and Bob

- have to exchange the key through a reliable channel,
- have to do so for each message anew,
- the key has to be as long as the message.

Public key cryptography solves the first two problems. The last problem is usually solved so as to leave the major part of communication unencrypted and to encrypt just a signature or the most sensitive data.

Classical public key cryptography relies on the assumed (sub)exponential complexity of factoring numbers. The most commonly used version, RSA, was introduced in 1978 by Rivest, Shamir, and Adleman [269] and is widely accepted today in spite of several shortcomings: a classical polynomially complex algorithm might be found; it can be up to 10^4 times slower than private cryptography, for example, DES; and it is software based, as opposed to DES, for which hardware has been developed.

The RSA public key protocol runs as follows:

Table 1.12 An example of *private key cryptography*: one-time pad.

1	original text	q	u	b	i	t
2	binary encoding (of 1)	101010	101110	011011	100010	101101
3	private key (control bits)	110010	010110	110110	111001	011010
4	XOR bits (2 $\oplus$ 3)	011000	111000	101101	011011	110111
5	encrypted text (of 4)	Y	4	t	b	3
6	XOR bits (4 $\oplus$ 3)	101010	101110	011011	100010	101101
7	decrypted text (of 6)	q	u	b	i	t

34) If she wanted to have a genuinely random sequence, she should actually employ a proper quantum process already here since classical processes or algorithms can only provide her with *pseudorandom* sequences.

1. Bob chooses two prime numbers p and q.
2. He calculates $n = pq$.
3. He selects e, which is *relatively prime* to $(p-1)(q-1)$ (two integers are relatively prime if their greatest common divisor is 1).
4. Pair $\{e, n\}$ is the *public key*. Bob sends it to Alice through a public channel.
5. Bob chooses an integer d that would reduce $(ed-1)/[(p-1)(q-1)]$ to an integer.
6. Pair $\{d, n\}$ is Bob's *private key*.
7. Alice uses the public key $\{e, n\}$ to encrypt message m by means of the equation: $c = m^e \bmod n$, where *mod*ulo means the remainder after division. Then, she sends c to Bob.
8. Bob decrypts the cyphertext c by means of the equation $m = c^d \bmod n$.

As an example, let us encrypt *qubit*. For convenience, we will switch to decimal representation. Using (1.305), we get $2^5 + 2^3 + 2 = 42$ for q, and 46, 27, 34, 45 for u, b, i, t, respectively. Thus, *qubit* is represented by 4246273445. In step one of the protocol, we choose (small) $p = \text{prime}[21] = 73$ and $q = \text{prime}[18] = 61$. We get $n = pq = 4453$ (step 2). For step 3, we find prime[606] = 4457 and prime[605] = 4451 and choose $e = 4457$. Finding a d that will reduce $r = (ed-1)/[(p-1)(q-1)]$ to an integer is not as straightforward, but a small program gives, for example, $d = 473$ and $r = 488$, and completes step 5. Now, because we have chosen a small n, we have to break 4246273445 into three parts and encrypt them piecewise. So, step 7 gives $c = 4426^e \bmod n = 4034$, $c = 2734^e \bmod n = 3344$, and $c = 45^e \bmod n = 1513$, and the encrypted message reads: `oi hs PN`. Bob then uses step 8 to decrypt the message: $4034^d \bmod n = 4246 = qu$, $3334^d \bmod n = 2734 = bi$, $1513^d \bmod n = 45 = t$.

More realistic examples can be generated with the help of programs available on the web.[35] They use hexadecimal numbers and extended ASCII (256) characters for encryption. In our case, 64-bit encryption suffices. We get $e =$ `aa934cd8a567932b`, $d =$ `1608a7af02c9c603`, $n =$ `c81f516f71fcb7c9`. This encrypts *qubit* as `1%^KimõE÷`. If we wanted to encrypt *qubit* so as to be unbreakable with today's technology for at least a few years – provided no one comes forward with a polynomial classical factoring algorithm in the meantime – we should use 1024-bit encryption. This gives us a key 8500 hexadecimal digits long, which encrypts *qubit* in a string of over 300 extended ASCII characters. Generation of the keys, encryption, and decryption may take up to a few seconds on today's PCs, which is quite a long time for writing down just a few words[36] at a distance. A realistic application also requires a method of authenticating Alice, so she too has to produce her own public and private keys and combine them with Bob's.

The main advantage of this system is that it is unbreakable, provided that the key is used only once (hence the name *one-time* pad). It is unbreakable because the random key randomizes the encrypted text as well: 0s and 1s in a binary representation of a character are unevenly distributed. Let the probability of having 0 in such

35) For instance, RSA key generation programs are included in almost all Linux distributions.

36) More words than just *qubit* – up to 20 words as long as *qubit* – could be encrypted by one 1024-bit key.

a representation be p. Then, the probability of having 1 is $1 - p$. The probability of having either 0 or 1 in a random key is 1/2. Hence, the probability of having 0 in the encrypted message after applying XOR is given by a sum of products of relevant probabilities: XOR gives 0 when we have either 0s or 1s in both the message and the key; the probability of the former case is $1/2p$ and of the latter $1/2(1 - p)$; their sum is 1/2; and therefore, zeros appear in an encrypted message evenly and ones, equally as so. However, if a key were used more than once, on two or more different messages, then one might be able to determine correlations between 0s and 1s in these encrypted messages and decipher the key.

Of all encryption on the Internet today, 95% is based on the 512-bit RSA, that is, in effect on intractability of factoring 512-bit numbers. Recently, several very efficient classical factoring algorithms have been developed. One of the fastest is the so-called *general number field sieve* or GNFS algorithm [262]. It enables a classical computer to factor a number N within a time window proportional to the following complexity:

$$\exp(1.923(\log N)^{1/3}(\log\log N)^{2/3}) \; . \tag{1.306}$$

In Figure 1.72, the complexity of the GNFS algorithm (c) is compared to the complexity of trial division on a classical digital computer (b) and a classical analog computer (a). We compare numbers in the vicinity of 2^{512}, on which current RSA cryptographic keys are based.

We first determine the complexity and time required for the GNFS algorithm with the algorithms for "brute force" trial division. We have to check the divisibility for at most $\sqrt{N} - 1$ integers (in case the number is a product of two subsequent prime numbers). This means that an algorithm for the division must carry out $\sqrt{N}$ steps. The best division algorithms can carry it out in $O(\log N)$ time [171]. Together, that gives the $O(\sqrt{N}\log N)$ time for trial division. This is the time that an analog computer would require since only in an analog computer can we import

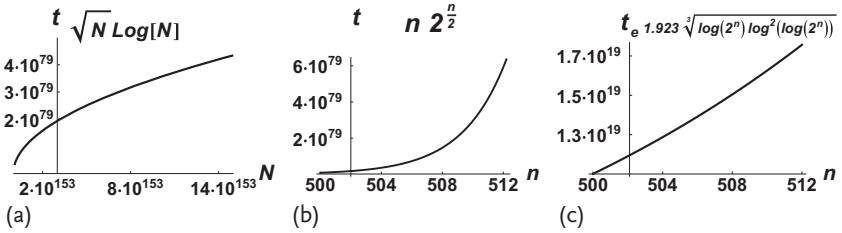


Figure 1.72 Complexities of factoring number N having n bits in the binary input size. (a) Trial division on an analog computer – it is not of exponential complexity; The problem is that there is no such a mighty analog computer around; $2^{512} \approx 13.4 \times 10^{153}$, $2^{508} \approx 0.84 \times 10^{153}$. (b) Trial division on a digital computer is a task of an exponential complexity because the time is expressed as a function of the number of bits (n) needed to write down $N = 2^n$ in a binary form; (c) the complexity of the GNFS algorithm expressed in terms of n; Compare the time range of (c) vs. (a) and (b). However, the range of (a) is given the same values as (b) only to show the complexity curve. Actual time range for (a) is much lower.

a number directly. Therefore, for an analog computer, factoring a number is not a task of exponential complexity. And indeed, for numbers up to 2^{33}, we can factor them within a time proportional to $N^{3/2}$ in the optical "physical computer" we described in Section 1.6. However, to construct such a computer or, say, an electrical analog, one that could factor much bigger numbers and especially 512-bit ones would hardly be feasible.

To be able to carry out trial division by means of a digital computer, we have to write down the number in binary representation. With respect to the number of digits, the time required for trial division *does* grow exponentially[37] because then, $N = 2^n$ and we get $O(\sqrt{N}\log N) = O(n2^{n/2})$, where n is the number of digits in binary representation.[38]

We can see from (1.306) that with the help of the GNFS algorithm, we can factor numbers within time frames that are many orders of magnitude (1.7×10^{19}) smaller than the time frames needed to carry out our trial divisions (5×10^{79}).[39] From Figure 1.72b, we can also see that the time required for a brute force approach – trial division – grows exponentially with size only when carried out on a digital computer.

In 1976, in Martin Gardner's *Scientific American* column, a 129-digit (in decimal representation or 429 in a binary one) RSA key was estimated to be safe for 40 quadrillion years [262]). But, the key (RSA-129) was already cracked in 1994 by K. Leutwyler, "Superhack: Forty quadrillion years early." [167]. For his "hacking," he received \$ 100 prize from the RSA Laboratories. They started to sponsor the RSA Factoring Challenge (with fixed rising prizes for given RSA numbers up to an RSA-2048[40]) to encourage research in computational number theory and the practical difficulty of factoring large integers and to help users of the RSA encryption public-key cryptography algorithm in choosing suitable key lengths for an appropriate level of security. When the prize reached \$ 20 000 in 2005 – collected by E.W. Weisstein for cracking RSA-640 – the RSA Labs withdrew the prizes for the other announced RSA numbers.[41] Nevertheless, in 2009, the group of T. Kleinjung cracked the RSA-768 (with 232 digits in decimal representation) [146].

37) Of course, with respect to decimal digits, it would be exponential too, but the digits are not essential for calculations in an analog computer. For example, the time needed to change the voltage or current for a definite value in an electrical analog computer does not essentially depend on the number of digits of their values.

38) The difference between Figures 1.72a and b with respect to the complexity only appears because of the difference in units in the plots. Thus, equidistant $n = 1, 2, 3, \ldots$ actually represent (2^n) $2, 4, 8, \ldots$ that are not equidistant. The absolute time is here for both representations of order 10^{79}, but only because $O(\sqrt{N}\log N)$ is expressed in the same time units. In a realistic analog computer – for lower values of N – the time would be much shorter because it is practically the same for a whole range of subsequent numbers.

39) The GNFS function given by (1.306) is of *subexponential* complexity. That means that time grows slower – as a function of n – than any exponential function, but faster than any polynomial in n.

40) Since 2003 RSA numbers denote digits in binary representation; so RSA-640 has 193 digits in decimal representation and 640 digits in binary representation, that is, simply 640 bits. RSA-2048 has 617 digits in decimal representation.

41) For cracking RSA-2048, the prize was \$ 200 000.

It is widely believed that a classical polynomial algorithm for computer factoring does not exist, although no one has proved this so far. Therefore, the reaction to the cracking of RSA-768 was, "Well, it seems that a 512-bit RSA is not safe any more. Okay, let's use 1024-bit RSAs." The nonclassical Shor's algorithm, which is polynomial and which surpasses the approach of the GNFS algorithm, could crack any such key on would-be quantum computers in seconds. However, as far as Internet security is concerned, efforts to make quantum computers work are not motivated by a desire to make the Internet unsafe, but rather to provide it with much safer technology – *quantum cryptography*. One night someone could come forward with an ingenious classical polynomial algorithm and make the Internet completely unprotected and unsure the very next morning.

1.22 ... and Why Quantum Cryptography Can?

Quantum cryptography comes as a remedy to an already rising paranoia about possible failures of classical cryptography. The concern is not only that someone may find a polynomially complex classical algorithm for factoring large numbers and thereby, overnight, leave Internet without protection, but also that in ten or twenty years' time one could decipher (using today's algorithms and the assumed computer speedup within this period) all the sensitive and confidential documents someone may have eavesdropped and stored today. Hence, a feasible cryptography that would prevent eavesdropping in principle is a highly sought-after goal. Quantum cryptography promises to fulfill this dream. However, it does not do that by introducing a new encryption algorithm, but by using quantum protocols which make eavesdropping impossible. It enables a transfer of messages from a sender to a recipient unconditionally secure.

Today, quantum cryptography is entering its physical and industrial application stage. A number of new companies develop commercial quantum cryptography systems. For instance, BBN Technologies (Cambridge, MA, USA), idQuantique (Geneva, Switzerland) MagiQ Technologies (New York, NY, USA), SmartQuantum (York, UK), Qutools (Munich, Germany), Optemax (Columbia, MD, USA), Qinetiq (Farnborough, UK), Senetas (Australia), and others. Well established companies like IBM, Toshiba, Siemens, Mitsubishi, Nec, HP, Verizon, Nokia, at&t, and others also develop quantum cryptography systems. The implementation projects are supported by multimillion dollar funding from government as well as private sources [223]. Most of the technologies implemented so far have recently been experimentally shown to be completely insecure [185], but the companies are currently developing patches to such attacks.

Quantum Information Processing and Communication (QIPC) is founded with more than $ 400 million, and according to a report by Global Industry Analysts, global quantum cryptography market will reach $ 850 million by 2015.

The following funding allocations include quantum cryptography: the DARPA of the Department of Defense (DoD) (2002–2007) with $ 50 million (Quantum Infor-

mation Science and Technology, QuIST program); in Europe, Quantum Information Processing and Communications: 5th and 6th EU Framework Programs (FP5 & FP6) (1998–2007), about $ 80 million; the EU Member States have earmarked a total of € 9.1 billion for funding Information and Communication Technologies within EU FP7, making it the largest research theme in the Cooperation program, which is itself the largest specific program of FP7 (with 64% of the total budget); quantum cryptography projects receive a significant percentage of funding within FP7. In Japan, government organizations are sponsoring nanotechnology R&D by $ 2.75 billion/year. Similar funding exists in China, Australia, and Canada, including many private contributions.

The main reason for the aforementioned investments was an increase in breaching the security of the networks; the losses have been estimated to hundreds of millions of dollars a year.

World's first quantum network – the DARPA quantum network – fully operational since October 23, 2003 [93], is a joint project of BBN Technologies, Harvard University, and Boston University. It was technologically implemented by BBN Technologies, and supported by DARPA.[42] The DARPA quantum network spans 29 km and does take an eavesdropper (Eve) into account. Two kinds of error correction schemes are implemented in the network and therefore the network can be considered unconditionally secure against attacks feasible with today's technology, provided the aforementioned patches against the [185] attack are implemented. The same applies to the following undertakings.

In October 2007, Senetas and idQuantique secured the Swiss federal election network by quantum encryption.

Between 2004 and 2008, the project *SEcure COmmunication based on Quantum Cryptography* (SECOQC) was realized in Vienna as a major research effort of 41 research and industrial organizations from the European Union, Switzerland and Russia as an Integrated R&D project within the 6th Framework program of the European Commission [250].

The SECOQC network consisted of six nodes connected by eight quantum key distribution links. The network was deployed in the internal glass fiber communication ring of Siemens (a SECOQC project partner) in Vienna, Austria. A diagram of this QKD given in Figure 1.73 which is taken over from [250]. The nodes SIE, BRT, and so on, were located in the premises of different Siemens dependencies in Vienna (Siemensstrasse, Breitenfurterstrasse, and so on). The design of each node module and implementation was carried out by the Austrian Institute of Technology in collaboration from University of Aarhus, Telecom ParisTech, University of Erlangen-Nuremberg, Bearing Point Infonova, and Siemens Austria.

The biggest quantum key distribution (QKD) network implemented so far is the Tokyo QKD Network [276]. It was made in 2010. Nine organizations from Japan and the EU participated. On the Japanese side, these consisted of the National Institute of Information and Communications Technology (NICT), Koganei, Tokyo,

42) DARPA is the Defense Advanced Research Projects Agency, the central research and development organization for the Department of Defense (DoD) of USA.

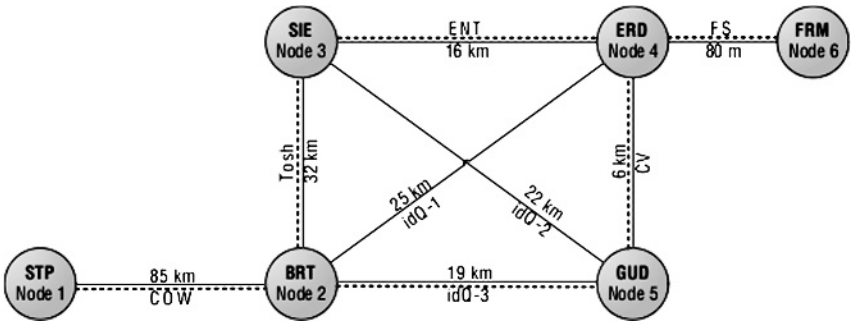


Figure 1.73 Communication based on Quantum Cryptography (SECOQC) Vienna network. The nodes BRT,...,FRM denote different locations in Vienna and STP is outside Vienna. Mutual distances of the nodes are 85, 32, 25, 22 km, and down to 6 and 80 m. Reprinted from [250] with permissions from the authors and from © 2009, Institute of Physics, New Journal of Physics.

Japan, then NEC, Mitsubishi Electric Corporation, and NTT, Toshiba Research Europe Ltd., UK, ID Quantique, Switzerland, the Austrian Institute of Technology, the Institute of Quantum Optics and Quantum Information, and the University of Vienna (the last three are called "All Vienna"). Six QKD links are shown in Figure 1.74. A three-layer architecture based on the key relay via trusted nodes was implemented, as shown in Figure 1.74, similar to the SECOQC network in Vienna.

To appeal to a wider adoption of QKD, applications demanded by potential users of high-end security technology were demonstrated: a secure TV conferencing and secure mobile phone in an area as wide as possible. Since efficient quantum repeaters are still not developed, to expand the QKD distance, one currently needs to rely on key relays via trusted nodes at least at every 80–100 km.

Let us now take a closer look at the quantum cryptography approach to better understand why it receives so much trust and funding. What would solve all the problems with the protocols and algorithms which we have presented in Section 1.21 is a combination of the reliability of a one-time pad and a new, fast, and unbreakable way of exchanging keys. And, this is exactly what the physics of quantum cryptography offers: not a quantum algorithm or "quantum software," but a solution based on the behavior of quantum systems themselves – quantum hardware. Quantum encryption of a document is not something we can keep on our desk and be sure that no intruder can decrypt it. The message – which can be either a random key or a plain text – is "quantum-encrypted" on its way from Alice to Bob. So, the transfer itself is the encryption. However, it is unbreakable by the laws of physics. Thus, so far as the software is concerned, the classical encryption which for its security relies on the available algorithms that require an unmanageable time to crack the encryption is actually and surprisingly more complicated than quantum encryption which does not rely on any quantum or classical computation algorithm.

The possibility of using quantum systems for a new quantum technology depends on whether we can control the particular quantum feature we would like to use. In cryptography, the feature we would like to use is exactly what we al-

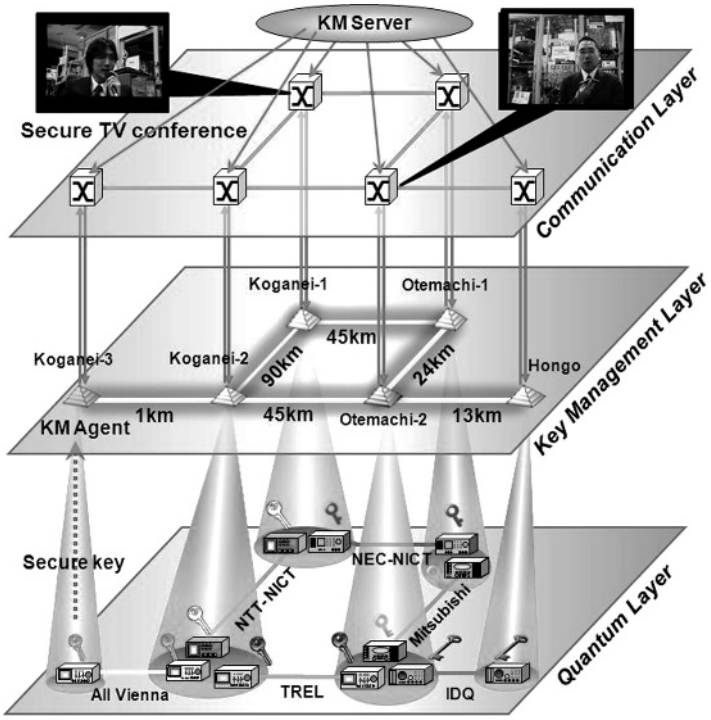


Figure 1.74 Various layers of the Tokyo QKD network. Reprinted from [276] with permissions from the authors and from © 2011, Optical Society of America.

ready have with entanglement. On the one hand, there is a genuine randomness of measurable properties of subsystems that emerges from the complete absence of their predetermined properties, and on the other, there is the perfect correlation of the measurable properties of subsystems when we jointly carry out measurements on them. In other words, Alice communicates with Bob, not by sending him an already encrypted message bit-by-bit, but by their joint recovery of bits from the correlation of their measurement clicks.

There are two main ways, we call them protocols, in which they can establish such a correlation. The ping–pong protocol [36] based on entanglement and the BB84 protocol (named after Bennett and Brassard [22]) based on genuine randomness of quantum measurement of an observable. The former is deterministic and the latter probabilistic. There are several other protocols as well, but they are often closely related to or derived from either the ping–pong protocol, like quantum secure direct communication (QSDC) [18] and QSDC without entanglement [182], or BB84, like B92 [21] (which makes use of two states; BB84 is a four-state protocol) and six-state protocol [48]. We shall present only the ping–pong, QSDC, and BB84 protocols. The former ones, because of their potential development and future implementation and the latter one because all present quantum cryptography implementations are based on it.

1.22.1
Entanglement in Action: Deterministic Communication

Entanglement is one of the main tools of quantum information engineering and therefore we shall first present a deterministic quantum cryptography protocol which is based on entanglement, although it is currently too demanding for a commercial implementation. The latter implementations, some of which we presented above, all utilize BB84 protocol. We shall elaborate on them in Section 1.22.2.

Arguably the best known deterministic protocol is the one proposed by Kim Boström and Timo Felbinger [35] and usually called a *ping–pong* protocol. It makes use of two Bell states $|\Psi^{\pm}\rangle$, given by (1.114) and (1.124), respectively. The qubits are photons. As follows from the property of entangled photons given at the end of Section 1.13, both photons from $|\Psi^{\pm}\rangle$ are unpolarized when measured independently of each other. That means that measuring the polarization of only one of the photons cannot reveal any information as opposed to measurements of both qubits. The latter measurement of both photons reveals whether the photons are in state $|\Psi^{+}\rangle$ or in state $|\Psi^{-}\rangle$.

Bob prepares a pair of photons in the state $|\Psi^{+}\rangle$ as shown in Figure 1.75. He keeps one of the photons – *home qubit* – and sends the other to Alice – *travel qubit*. *Ping!* Alice either directs the photon through a HWP(0°) (see (1.191)) and returns it to Bob, or returns it to Bob as is. *Pong!* In other words, she either applies σ_z (see Table 1.10) to the travel-photon state or I.

Bob then performs a Bell measurement on Alice's and his own photon with photon number detectors (for the "control mode" below). If Alice lets her photon through the HWP, he will detect $|\Psi^{-}\rangle$ and if Alice puts nothing in the path of her photon, then Bob will detect $|\Psi^{+}\rangle$. Thus, Alice sends Bob one bit of information: two messages that Bob recognizes as $|\Psi^{-}\rangle$ and $|\Psi^{+}\rangle$. The procedure is shown in Figure 1.75a. We say that Alice–Bob communication is in a *message mode*.

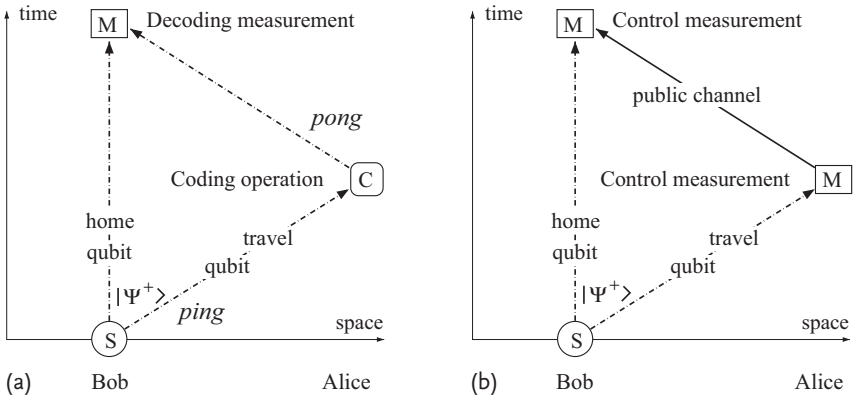


Figure 1.75 Ping–pong protocol. (a) message mode; (b) control mode. The figure is made according to Figures 1 and 2 from [35].

To be sure that they are not eavesdropped, Alice and Bob occasionally go to a *control mode*. In the control mode, Alice directs her photon through a HWP($\pi/8$) to obtain $|\Phi^-\rangle$. After Bob receives Alice's photon and carries out the measurement, Alice informs him about her sending $|\Phi^-\rangle$ via a public channel. If he did not detect both photons by the same detector, then Eve is in the line and they abort the communication [241]. With such a check, they have the probability of 50% to catch Eve, but with repeated checking, the chances go arbitrary close to 100% very fast.

Since 2002, several loopholes in the security of the ping–pong proposal have been found, but also soon patched [36]. It has been proved by A. Wójcik [326] and Z.-J. Zhang, Z.-X. Man, and Y. Li [339, 340] that the original ping–pong is insecure unless the efficiency of transfer exceeds 60 and 80% respectively. Several remedies were proposed already in the above references. Other improvements were proposed most recently [175, 336]. Objections put forward in [337, 338] have been proved wrong [36].

It looked like, the more states we have, the more secure a protocol would be although it might be more difficult to implement. Let us investigate this option.

A four state extension of the ping–pong protocol has been proposed by Q.Y. Cai and B.-W. Li [58]. Hereby, the transfer capacity is increased by enlarging the photon state basis from two states to all four Bell states. However, with today's technology, a realistic discrimination of all four Bell states cannot be implemented (see Section 1.16). For instance, in the Knill–Laflamme–Millburn scheme [147], such a discrimination would require over 200 linear-optics elements like beam splitters and phase shifters.

We can obtain a much simpler four state deterministic protocol by making use of the following result.

Eavesdropping is an issue with QSDC (direct communication) because Eve can snatch some information before Bob detects her. But the limitations Eve imposes on the communication might also be helpful in enabling Alice and Bob to switch to a simpler technology. Let us consider their options.

Bob prepares a photon pair in the $|\Psi^-\rangle$ state, keeps the home photon (h) and sends the travel photon (t) to Alice as shown in Figure 1.76a. Eve lets the photon (while it flies to Alice) through her device shown in Figure 1.76b. She feeds her y mode with a horizontally polarized ancilla photon ($|0\rangle$). She leaves the x mode empty: $|\varnothing\rangle$. The state of the whole system is $|in\rangle_{htxy} = |\Psi^-\rangle_{ht}|\varnothing\rangle_x|0\rangle_y$. The operator which describes the action of the device from Figure 1.76b is: $Q_{txy} = \mathrm{CNOT}_{ty}\mathrm{CNOT}_{tx}\mathrm{PBS}_{xy}\mathrm{CNOT}_{t(x)y}\mathrm{CNOT}_{tx}(H_x \otimes H_y)$, where H is the Hadamard gate and PBS is a simple polarizing beam splitter switch-operator; for example: $\mathrm{PBS}|\varnothing\rangle_x|0\rangle_y = |0\rangle_x|\varnothing\rangle_y$, and so on. Action of Q: $|q\rangle = Q_{txy}|in\rangle_{htxy}$, then Alice's preparing of $|\Psi^\mp\rangle$, $|\Phi^\mp\rangle$, and the final Eve's action by her device in a reverse order, are respectively given as:

$$|q\rangle = \frac{1}{2}[|0\rangle_h(|1\rangle_t|1\rangle_x|\varnothing\rangle_y + |1\rangle_t|\varnothing\rangle_x|0\rangle_y) - |1\rangle_h(|0\rangle_t|0\rangle_x|\varnothing\rangle_y + |0\rangle_t|\varnothing\rangle_x|1\rangle_y)] ,$$

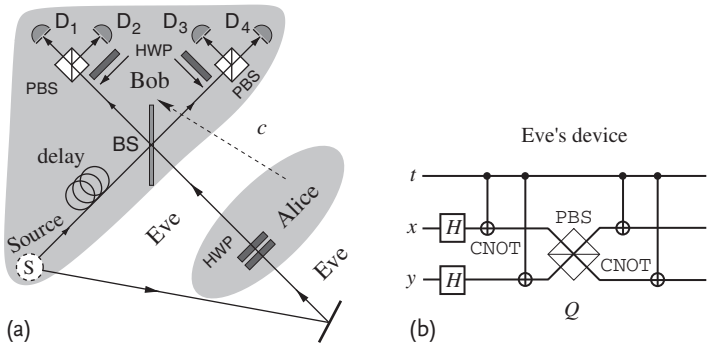


Figure 1.76 *A four-state deterministic communication setup.* (a) Alice puts both, one, or none of her (half-wave plates) HWPs; she let Bob know whether she sent him a Ψ or a Φ state over the classical channel c; (b) Eve's device described by the operator Q see (1.307).

$$Z^{(\Psi^{\pm})} = |1\rangle_{tt}\langle 1| \mp |0\rangle_{tt}\langle 0| \ , \quad Z^{(\Phi^{\pm})} = \pm(|0\rangle_{tt}\langle 1| \mp |1\rangle_{tt}\langle 0|) \ ,$$
$$Q^{\dagger}_{txy} Z^{(\Psi^{\pm})} |q\rangle = |\Psi^{\pm}\rangle_{ht}|\varnothing\rangle_x|0\rangle_y \ , \quad Q^{\dagger}_{txy} Z^{(\Phi^{\pm})} |q\rangle = \mp|\Phi^{\pm}\rangle_{ht}|0\rangle_x|\varnothing\rangle_y \ . \quad (1.307)$$

We see that Eve can distinguish $|\Psi^{\mp}\rangle$ from $|\Phi^{\mp}\rangle$ without altering the travel photon state, that is, being undetectable. Since, Alice and Bob cannot avoid such Eve's undetectable $\Psi - \Phi$ discrimination they decide to make use of it. Alice publicly announces when she sends Ψ and when Φ. That knowledge enables Bob to easily discriminate $|\Psi^{-}\rangle$ from $|\Psi^{+}\rangle$ and $|\Phi^{-}\rangle$ from $|\Phi^{+}\rangle$.

Why would they want to stay with four states? Because it can be shown that the security is higher than with two separate two state protocols. All they have to do is to send uncorrelated messages over Ψ and Φ bundles.

Bob generates $|\Psi^{-}\rangle$ photon pairs by means of spontaneous parametric down-conversion in a BBO crystal [190] and sends one of the photons from each pair to Alice as shown in Figure 1.76a. To send $|\Psi^{-}\rangle$ to Bob she puts nothing in the path of her photon. To send $|\Psi^{+}\rangle$, she puts in HWP(0°) (half-wave plate) in the path. It changes the sign of the vertical polarization. To send $|\Phi^{+}\rangle$, she puts in HWP(45°). HWP(45°) turns $|\Psi^{+}\rangle$ into $|\Phi^{-}\rangle$. To send $|\Phi^{-}\rangle$, Alice takes out HWP(0°).

On his side, Bob identifies $|\Psi^{-}\rangle$ by means of coincidence clicks of detectors D_1 (D_2) and D_3 (D_4) and $|\Psi^{+}\rangle$ by means of coincidence clicks of either detectors D_1 and D_2 or D_3 and D_4. To identify Φ states, upon Alice's notification over c, Bob puts HWP(45°) in the paths of the photons. $|\Phi^{-}\rangle$ photons split on PBSs and D_1 and D_2 or D_3 and D_4 are triggered. $|\Phi^{+}\rangle$ photons trigger either D_1 or D_2 or D_3 or D_4.

Detectors with photon number resolution can recognize two photons in state $|\Phi\rangle$ in one step, and hence their discrimination is deterministic. The highest efficiency of such detectors is currently about 90% [114, 270] and it is expected to be higher in the near future. We can also use single-photon detectors whose highest efficiency is in 99% [173, 213] so as to successfully split photons at a "tree" of beam

splitters (two bunched photons are sent to the first beam splitter where they split with the probability of 50%; each beam is then sent to the next row of two beam splitter where the photons split again, and so on, as shown in Figure 1.43).

Taken together, the coincidence clicks shown in Table 1.13 correspond to a deterministic discrimination of all four Bell states. "Simultaneous clicks" for $|\Phi^+\rangle$ mean that the corresponding detectors detected two photons.

The most efficient way to catch Eve is that Eve sends a c' signal together with $|\Psi^\pm\rangle$ and a c'' signal together with $|\Phi^\pm\rangle$. If Bob finds that messages with c' (c'') really are $|\Psi^\pm\rangle$ ($|\Phi^\pm\rangle$) and not $|\Phi^\pm\rangle$ ($|\Psi^\pm\rangle$), they consider them to be legal messages and they just continue transmission. If not, Eve is in the line and they abort the transmission. In this way, the control and transmission channels are mixed together and now switching between them is required.

Let us now present an explicit algorithm of the above informal procedure. This is a standard way to present cryptographic procedure.

a) Protocol is initialized. $n = 0$. The message to be transmitted is a sequence $x^N = (x_1, \ldots, x_N)$, $x_n \in \{00, 01, 10, 11\}$.
b) $n = n + 1$.
c) Bob prepares two qubits in state $|\Psi^-\rangle$.
d) Bob stores one qubit (*home qubit*) and sends the other (*travel qubit*) to Alice through a *quantum channel*;
e) For $x \in \{00, 01, 10, 11\}$, Alice performs coding operations $|\Psi^-\rangle \to |\Psi^\pm\rangle$, $|\Phi^\pm\rangle$ on the travel qubits and sends them back to Bob; for $|\Psi^-\rangle \to |\Phi^\pm\rangle$, she informs Bob by means of c signal; all signals are sent with a delay to prevent Eve's actions based on receiving them;
   – Bob receives the travel qubit, performs measurements on both qubits and obtains the Bell states; if they are accompanied by a classical signal, Goto (f)– or else, Goto (h);
   – does not receive the travel qubit; error in transmission; Goto (c);
f) If c' (c'') is detected, Bob checks whether he received what Alice sent; if not, Goto (g); If yes, Goto (h); or else, signal c is detected: Goto (h);
g) Eve is in the line; abort transmission.

Table 1.13 Ideal discrimination of all four mixed basis states with photon number resolution detectors. $|\Phi^\pm\rangle$ detections assume that HWP(45°) is inserted (see Figure 1.76a).

	Simultaneous "clicks" at		
$\|\Psi^+\rangle$	D_1 AND D_2	OR	D_3 AND D_4
$\|\Psi^-\rangle$	D_1 AND D_3	OR	D_2 AND D_4
$\|\Phi^-\rangle$	D_1 AND D_2	OR	D_3 AND D_4
$\|\Phi^+\rangle$	D_1 OR D_2	OR	D_3 OR D_4

h) Bob decodes the messages as follows:

$$|\Psi'\rangle = \begin{cases} |\Psi^-\rangle & \Rightarrow x_n = 00 \\ |\Psi^+\rangle & \Rightarrow x_n = 01 \\ c \quad |\Phi^-\rangle & \Rightarrow x_n = 10 \\ c \quad |\Phi^+\rangle & \Rightarrow x_n = 11 \ , \end{cases}$$

where "c-rows" denote Bob's detecting messages after inserting HWP(45°) in the paths of photons, following Alice's announcement on sending Φ photons.

i) – $(n < N)$: Goto a).
   – $(n = N)$: Goto j).

j) Message x^N is transmitted from Alice to Bob. Communication successfully terminated.

The security of the ping–pong protocol with two Bell states requires transfer efficiency of 80%. For the four-state setup, this reduces the required transfer efficiency significantly since the same Eve's gate cannot simultaneously reveal all four mixed states. The transfer efficiency of the latter protocol is not critical because its main protection is a very high number of possible checking via classical signals c' c'' without interrupting the transmission if Eve is not detected. And, Eve can be detected with a sequence of just 20 such signals with a probability higher then 99%.

However, we can still estimate two kinds of the security of the protocol: the maximum information Bob can receive and the maximum probability of detecting Eve during transmission.

The maximum information Bob can receive is given by the Holevo limit [49]

$$\mathcal{E} = \frac{b_s}{q_t + b_t} \ , \tag{1.308}$$

where b_s is the number of bits received by Bob, q_t is the number of qubits, and b_t is the number of bits received via a classical channel. We neglect the bits carried by signals c' and c'' since their number is very small in comparison with the number of messages [49].

For the four-state protocol, the number of qubits in the transfer mode is $b_s = 2$, the number of secret bits exchanged via a quantum channel in the transfer mode is also $b_t = 2$ (four messages), and the number of bits exchanged via a classical channel in the transfer mode is $b_t = 1$ (signal c, carrying 1/2 bits is sent with Φ messages, but Bob takes out HWPs when he does not receive a c message and that amounts to other implicit 1/2 bits). Therefore, we have obtained $\mathcal{E} = 0.67$ and it exceeds all implemented and all realistically feasible protocols. For example, Bennet–Brassard's 1984 protocol (BB84) has efficiency $\mathcal{E} = 0.25$, Bennet 1992 (B92) has $\mathcal{E} = 0.25$, Goldenberg–Vaidman 1995 $\mathcal{E} = 0.33$, Ekert 1991 $\mathcal{E} = 0.5$, and the ping–pong $\mathcal{E} = 0.5$.

We also mention here that Bob can detect a mischievous Eve who wants to disturb the information transmission without being detected. In the ping–pong protocol, Eve can change the phase of Alice's qubit at random and that would randomly

change $|\chi^1\rangle$ into $|\chi^2\rangle$ and vice versa and make the message unintelligible. In our case, such a phase shift would randomize the control messages announced by Alice over c' and c'' (with a delay) and that would reveal Eve.

We estimate the probability of Alice and Bob detecting Eve as follows. Since Eve has to hack four states, the probability of detecting Eve is 1/2 and for N checks, this makes $1 - (1/2)^N$. Thus, already after 10 transmitted messages, we have a probability of detecting Eve equal to 99.9%.

Deterministic protocols are helpful for understanding quantum communication, however, their implementation designs are extremely demanding for the time being. The hardest parts are sources of entangled photon pairs on demand, photon number resolution detectors, storages of photons (Bob has to make use of for his photons while waiting for Alice's photons to arrive), handling the transmission through lousy channels, and achieving satisfactory security in noisy channels. Therefore, a commercial implementation of deterministic protocols is not likely to be realized for the time being. Such an implementation has been done by means of the BB84, though.

1.22.2 No-cloning in Action: Probabilistic BB84 Protocol

The BB84 quantum cryptography protocol is based on the no-cloning Theorem 52 (Section 1.16) and the inherent randomness of the measurement outcomes for a measured observable. This enables us to use single photons for transferring messages and lasers as their sources. Laser sources of individual photons are far easier to implement than sources of entangled photons on demands. Also, in a realistic implementation, phases are used for encoding messages, but in this section, we shall nevertheless use polarizations because they make the protocol BB84 easier to understand. We shall present the phase approach in Section 1.23.

To see how the protocol BB84 works, let us look at Figure 1.77. Alice sends vertically, horizontally, and diagonally polarized photons (only the vertically polarized are shown in the figure) to Bob, who receives them through anisotropic birefringent plates that split incident beams into two beams with two different directions and polarizations. Beams exiting the plate are called *ordinary* and *extraordinary* rays. They are polarized at right angles to each other. A birefringent plate serving as a polarizing beam splitter lets a photon through – either as an ordinary or as an extraordinary ray.

In Figure 1.77, we denote the polarization of the ordinary ray with a bright arrow and of the extraordinary ray with a dark arrow. Both the photons sent by Alice which are entering the plates, and the photons received by Bob which have exited the plates, can be oriented along four different directions: $\updownarrow$, $\nwarrow\!\!\searrow$, $\swarrow\!\!\nearrow$, and $\leftrightarrow$. In Figure 1.77, however, only Alice's $\updownarrow$ photons are shown. Of these vertically polarized photons sent by Alice along path `a`, Bob always receives only the ordinary ray polarized vertically and never along `d`. The chances of a photon appearing as either $\nwarrow\!\!\searrow$ or $\swarrow\!\!\nearrow$ from the diagonally oriented plates on paths `b` and `c` are 50 : 50. As it is

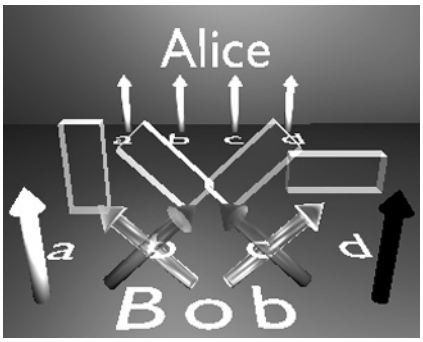


Figure 1.77 *Physical scheme underlying quantum cryptography*. Alice can send photons polarized in two bases: ⊞ and ⊠. Here, only vertically polarized photons, ↑ in basis ⊞ are shown. Incoming ordinary photons pass through a birefringent crystal and exit as ordinary ones (bright arrows) or extraordinary ones (dark arrows). Whether a photon moving along paths b and c will appear in an ordinary (bright arrows) or an extraordinary ray (dark arrows) is completely unpredictable (random). With the horizontally polarized incoming photons, it is the other way round.

obvious from Definition 44 (Section 1.12), vectors ↕, ↔ and ↖↘, ↗↙ can form two bases in either of which one can determine any polarization vector in our example. Let us denote these bases by ⊞ and ⊠, respectively. Of course, if Alice sent photons oriented along ↗↙, then Bob would always have clicks from ordinary photons at c, never from those at b, and on average every second time from ordinary photons at a and d. We denote qubits $|0\rangle$ and $|1\rangle$ in basis ⊞ as $|\updownarrow\rangle$ and $|\leftrightarrow\rangle$, respectively, and in basis ⊠ as $|\nwarrow\searrow\rangle$ and $|\nearrow\swarrow\rangle$.

There are several quantum cryptography protocols that use this method of communication. We shall present BB84, named after Bennett and Brassard [22].

1. Alice chooses random data bits (0 s and 1 s).
2. She chooses bases ⊞ and ⊠ at random.
3. She sends qubits (photons) to Bob.
4. Bob randomly chooses bases, that is, orientations of birefringent plates prior to receiving each photon.
5. He measures the polarization of the photons.
6. He publicly announces the bases he used whenever he detects a photon and Alice (also publicly) says which bases were correct.
7. They discard results corresponding to incorrect choices of bases.
8. To check whether *Eve* has been eavesdropping, Bob publicly reveals some of the results kept after step 7.
9. Alice confirms them. If they find that the results in step 8 differ unacceptably, they abort the protocol.
10. If the results in step 8 do not differ significantly, the remaining bits are Alice and Bob's secret key.

We illustrate the protocol by the example shown in Table 1.14.

There are several points to be emphasized in the above protocol:

- Whenever Bob and Alice use the same bases, the vectors they obtain are (ideally) correlated, and whenever they use different ones, the vectors are uncorrelated and only 50% of obtained bits are correct. Therefore, Bob's error rate would have been 25% if he had taken into account the results obtained in different bases. However, Bob and Alice discard the latter results altogether, that is, they discard 50% (ideally) of all the results, following step 6, and the discarded 50% include the aforementioned 25% errors. The bits they keep we call the *sifted* key. Ideally, this key, would be flawless and unbreakable.
- In a realistic setup, Bob and Alice discard more than 50% of their results. This procedure compensates for one kind of error, such as poor single-photon detection. However, some other errors, like those stemming from nonperfect alignment of Bob's vs. Alice's polarizers, cannot be directly detected since Bob and Alice cannot announce the results. For instance, when both Bob and Alice use basis ⊞, sometimes, when the alignment is not perfect, a photon polarized along ↕ emerges from Bob's polarizer as ↔. To correct such errors, Bob and Alice can apply error correction schemes (see Section 1.19) [59, 297].
- Eve's eavesdropping will appear to Bob as a combination of both kinds of errors. Hence, he will be sure of her eavesdropping only if his error rate is high enough. Let us first consider the simplest eavesdropping method Eve could use: she puts a polarizer – randomly chosen to be oriented along either ↕ or ↔ for ⊞ or along either ↘ or ↙ for ⊠ – in the path of Alice's photons and keeps it there all the time. Then, the information on the chosen bases that Alice and Bob exchange in step 6 of the protocol, together with the orientation of her polarizer, would give her 50% of Alice's bits when the orientation of her basis coincides with Alice's (Alice first throws away those bits that Bob does not have a record of, and thereupon Alice and Bob throw away bits for different bases.) Altogether, Eve can copy 25% of the results. As for Bob, he would not receive 50% of the pho-

Table 1.14 An example of the BB84 protocol. The numbers in the first column correspond to the steps in the protocol according to [22].

1	**1**	**0**	**0**	**1**	**0**	**0**	**1**	**1**	**0**	**1**	**0**	**0**	**1**	**1**	**0**
2	⊠	⊞	⊠	⊞	⊞	⊞	⊞	⊞	⊠	⊠	⊞	⊠	⊠	⊠	⊞
3	↙	↕	↘	↔	↕	↕	↔	↔	↘	↙	↕	↘	↙	↙	↕
4	⊞	⊠	⊠	⊞	⊞	⊠	⊠	⊞	⊠	⊞	⊠	⊠	⊠	⊠	⊞
5	↘		↘		↕	↙	↙	↔		↕	↕	↘		↙	↕
6	⊞		⊠		⊞	⊠	⊠	⊞		⊞	⊠	⊠		⊠	⊞
7			✓		✓			✓				✓		✓	✓
8					↕									↙	
9					✓									✓	
10			0					1				0			0

tons at all, and 50% of the photons passing through Eve's incorrectly oriented polarizers would also be read off "incorrectly" by his birefringent plates. (When Alice uses ⊞ and ↕ and Eve ⊠, Bob gets ↕ photon through ⊞ in half of the measurements. Thus, he recovers 50% of the photons that passed through Eve's incorrectly oriented polarizers.) Therefore, his error rate would be 62.5%, and the protocol is aborted. However, Eve can decide to apply her strategy to only a fraction of Alice's bits, say 5%, and rely on getting more information when the key is applied to a message Alice would send to Bob later on. Then, Eve's information is about 1.2% and Bob's error rate about 3.12%.

Taken together, the BB84 protocol is robust and leaves intruders little chance. It is, however, physically interesting to see whether we can adapt the protocol so as to cancel out an eavesdropper's attempts completely, that is, whether quantum cryptography can be unconditionally secure.

1.23
Why There Can Be No Quantum Eavesdroppers? Unconditional Security

To prove the unconditional security of quantum cryptography, it is, at least for the time being, enough to prove it for BB84. Before we dwell on the actual proof of the unconditional security, we will present variations of BB84 with respect to number of required states, realistic robustness, and underlying physical schemes [27, 178, 192].

Quantum cryptography protocol BB84, presented in the previous section, makes use of basically one single quantum feature: individual system states prepared in one basis (say ⊞) might be totally uncorrelated with individual states of the same system measured in some other basis (say ⊠). This feature suffices for secure distribution of secret keys, which is essentially what quantum cryptography is all about: a secure replacement for insecure classical public key distribution. All other parts of cryptocommunication remain classical. For example, Alice first sends a completely random sequence of 0s and 1s to Bob using the BB84 protocol. Then, Bob throws away all the messages received in the "wrong" basis and informs Alice on all the messages he received – of course, he does not disclose whether he received 0 or 1 (the content of a message), but only which message in a sequence of Alice's attempts he actually received, say, the 31st, then 342nd, and so on. This sequence of successfully transfered 0 s and 1 s form the *key* shown in line 3 of Table 1.12 in Section 1.21. Then, Alice encrypts the text shown in line 1 by making use of this key and XOR (line 4) to obtain the encrypted text shown in line 5 and sends it to Bob through a public channel; Bob, on his side, uses the key he and Alice agreed on, to decrypt the message also by means of XOR (lines 6 and 7 Table 1.12). This is why it is often stressed in the literature that quantum cryptography should actually be called *quantum key distribution* (QKD) [103].

The QKD term refers to the fact that Alice cannot send unencrypted message through a quantum channel by making use of BB84 protocol because 0 s and 1 s of a binary representation of an unencrypted text are not randomly distributed in a transmitted string (and can therefore be cracked via sophisticated classical algorithms) while 0 s and 1 s of the key are. Alice can directly send messages only via deterministic communication presented in Section 1.22.1.

There are actually many varieties of quantum key distribution protocols [103]. First, we need not keep to four states, as in BB84. Two states are enough, as the so-called B92 demonstrates [21], but the security of B92 is lower than the security of BB84. A six-state protocol [48] reduces Eve's information gain for a given error rate [103], though it is more demanding. Hence, BB84 tends to be standard.

Next, polarization is very suitable for understanding and carrying experiments in a laboratory, but it is not robust enough to allow implementation over larger distances (more than a few kilometers).

Figure 1.78 shows a phase-coding scheme that is nothing but an optical fiber version of the Mach–Zehnder interferometer shown in Figure 1.10. Here, beam splitters are substituted by fiber couplers (optical devices that merge two fibers). The probabilities of detectors D_0, D_1 registering a photon are given by (1.8):

$$p_0 = \cos^2 \frac{\phi_A - \phi_B}{2} , \quad p_1 = 1 - p_0 = \sin^2 \frac{\phi_A - \phi_B}{2} . \tag{1.309}$$

Alice makes use of four phase shifts $\phi_A = 0, \pi/2, \pi, 3\pi/2$ and associates bit 0 with $\phi_A = 0$ and $\phi_A = \pi/2$ and bit 1 with $\phi_A = \pi$ and $\phi_A = 3\pi/2$. Bob makes use of two phase shifts $\phi_B = 0, \pi/2$ and associates bit 0 with a click of detector D_0, that is, with $p_0 = 1$, and bit 1 with a click of detector D_1, that is, with $p_1 = 1$. They have to discard cases when there is a 50 : 50 probability of either D_0 or D_1 clicking, that is, when $p_0 = p_1 = 1/2$. This happens, for instance, when Alice chooses $\phi_A = 0$ and Bob $\phi_B = \pi/2$ since then $p_0 = p_1 = \cos^2(-\pi/4) = 1/2$. Hence, they can implement the BB84 protocol as shown in Table 1.15 following [103].

In realistic applications, it is difficult to control the lengths of the two fibers in the above setup up to a fraction of the wavelength of photons. Specifically, for nonequal paths, (1.309) reads

$$p_0 = \cos^2 \frac{\phi_A - \phi_B + k\Delta L}{2} , \tag{1.310}$$

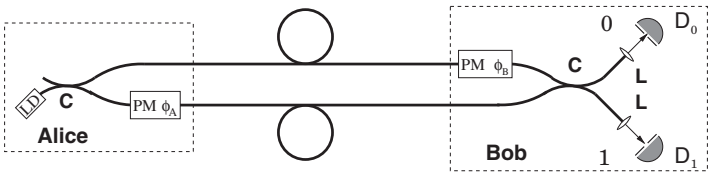


Figure 1.78 *Phase-coding scheme of quantum cryptography*: optical fiber Mach–Zehnder interferometer. LD is a laser diode; PM, phase modulators; C, symmetric fiber couplers – equivalent to beam splitters; D, avalanche detectors; L, lenses.

Table 1.15 Phase-coding implementation of BB84 protocol [103].

Alice's bits	ϕ_A	ϕ_B	p_0	p_1	Bob's bits
0	0	0	1	0	0
0	0	$\pi/2$	1/2	1/2	undetermined
0	$\pi/2$	0	1/2	1/2	undetermined
0	$\pi/2$	$\pi/2$	1	0	0
1	π	0	0	1	1
1	π	$\pi/2$	1/2	1/2	undetermined
1	$3\pi/2$	0	1/2	1/2	undetermined
1	$3\pi/2$	$\pi/2$	0	1	1

where k is the wave number and ΔL is the path-length difference.

Therefore, variations on phase-coding that use only one fiber, with pulses going through it with a delay, have been put forward as the most suitable for a practical implementation. For instance, both Alice and Bob can have their own Mach–Zehnder interferometers with unequal paths at each side, as proposed by Charles Bennett [21]. A photon taking the shorter path in Alice's interferometer and the longer in Bob's cannot be distinguished from a photon taking the longer path in Alice's interferometer and the shorter in Bob's, and so we obtain the desired interference. Successful experiments have been carried out with fibers from 10 km [306] to 150 km long [123]. An even more robust design with two photon pulses traveling from Bob to Alice and back to Bob over the same fiber has been implemented [103] and called *plug-and-play quantum cryptography* since it requires no adjustment prior to usage.

As for the physical schemes underlying possible quantum key distribution protocols, apart from the single photons of BB84, we can also use a pair of photons entangled within an EPR pair, as given by (1.114). That is, instead of the scheme shown in Figures 1.77, we can use the scheme shown in Figure 1.32. By comparing the probability function of single polarized photons[43] with the probability function for two photons entangled in polarization[44], we see that the two schemes are completely equivalent. Let us now dwell on the proof of the unconditional security of BB84, which generally amounts to such proof for quantum cryptography.

As mentioned above, all previous proofs of the unconditional security of quantum cryptography [27, 178, 192] were reduced by Shor and Preskill [288] to a proof for BB84 based on quantum error correction and privacy amplification. The idea behind this reduction is that errors in transmission caused by technical imperfections are indistinguishable from the effects of Eve's eavesdropping. So, the proof of unconditional security of quantum cryptography consists of showing that by error correction, we can reduce both the difference in Alice's and Bob's keys and the percentage of the key Eve can possess to arbitrarily small amounts.

43) The probability of photons coming out from the first polarizer being detected after passing through the second one – given by (1.22) for the first scheme

44) The probability of photons being detected by the two detectors shown in Figure 1.48 – given, for example, by (1.220) for the second scheme

In general, to enable Alice and Bob to exchange a key *unconditionally securely*, we should, in addition to the BB84 protocol from Section 1.22.2, also assume that they have a quantum computer at their disposal to store qubits in its memory and to correct bit-flips and phase shift errors in transmission of qubits. Fortunately, error correction schemes have bit-flip and phase shift correction completely separated, as, for example, the CSS scheme presented in Section 1.19. So, we actually do not need a quantum computer. Using a CSS scheme, we can keep the bit-flip correction and average over random phase vectors obtained by the syndrome measurements and get a mixed state that is equivalent to a randomly chosen code string [288]. A bit-flip correction suffices for a complete recovery of all bits changed in transmission.[45)]

Taken together, in addition to the points of the BB84 protocol from Section 1.22.2, we should, according to Shor and Preskill [288], further assume the following:

11. Alice publicly announces $u + v$ where v is the sifted key block (the remaining bits they agreed upon in point 10) and u is a random codeword in Steane's code C. Bob adds[46)] $u + v$ to his string $v + e$ and corrects the result $u + e$ he received through the quantum channel to the codeword in C.
12. Alice and Bob use the coset $u + C'$, $C' \subset C$ (see below) as their key.

This point deserves some comments: (a) The error correction scheme is reduced to the classical case. For the sake of simplicity, we will use the simplest Hamming code C, although there are other more sophisticated and bigger codes, in particular those that can correct more than one error. (b) The length of the block Alice chooses to send must match the length of the codeword. In our case, both v and u contain seven bits. (c) v might not belong to the code C, that is, Bob cannot correct v directly. (d) Eve cannot make use of $u + v$ to increase her information because Alice picks up bits for v at random, and the probability that Eve already has all these bits is negligible. Thus, Eve will have more than one error and cannot correct these errors.

Let us consider the following example for these points with Alice picking up seven measurements among all those she and Bob sifted in step 10 of the BB84 protocol in Section 1.22.2. She knows she sent, for example, $v = [1110001]$. Bob, however, might have received it with a bit-flip: $v + e = [1111001]$. Neither v nor $v + e$ are Hamming codewords. Alice randomly picks up a codeword, for example, $u = [0011001]$, and publicly announces $u+v = [1101000]$. Bob adds it to $v+e$ and gets $v+e+u+v = [0010001] = u+e$. By using the check matrix for the Hamming code given by (1.286), Bob gets the syndrome $s^{\mathrm{T}} = H \cdot [0010001]^{\mathrm{T}} = [100]^{\mathrm{T}}$. This syndrome s points to the fourth column of h, and he learns that his fourth bit has flipped.

Now, we turn to the code C' and the *coset* of C' – but only to go around it below. C' must be such that its dual code $C'^{\perp}$ has the same minimal distance (in our

45) Here we assume that there is, on average, one error per message block within a key.

46) In the literature one often finds the term "subtracts" here. However, for XOR, these two operations reduce to each other.

case, 3) as C [297]. The coset of C' determined by u is the set of all the words of the form $u + c$ as c ranges over all words in C': $u + C' = \{u + c | c \in C'\}$. The generator G' of the dual code $C'^{\perp}$ gives the word $G'(u)$, which is in one-to-one correspondence with the coset of C': $G'(u) \leftrightarrow u + C$ [179]. Instead of giving the details of computing this one-one correspondence, we would rather present a shortcut that amounts to the same result.

Let us substitute the following step for the step 12 above:

$\widehat{12}$ Bob uses the obtained e to correct his received string $v + e + e = v$, and thus Alice and Bob share the identical key-part v.

After Alice and Bob repeat steps 11 and $\widehat{12}$ enough times, they become almost certain that they have identical copies of the key.

However, computing the coset in step 12 would provide Alice and Bob with *privacy amplification*, and in using $\widehat{12}$, they have to do it separately. This is necessary because Eve can still be in possession of some parts of the key corresponding to the photons she measured and resent to Bob in a correct basis. To reduce the information that Eve can possess, Alice and Bob can apply the following amplification procedure. They pick a pair of bits (informing each other of their choice through a public channel) and substitute it with an XORed value of the pair. Since there is a high probability that Eve will not have information on both bits, this reduces her knowledge. Alice and Bob can iterate the procedure until the information Eve has gained cannot jeopardize the security of their key anymore.

This completes our sketch of the proof of the unconditional security of quantum cryptography. The maximum tolerable error rate is computed to be 11% [179, 288], but we have not elaborated on error and reliability estimates since too many experimental improvements have been achieved recently. For example, last year, the experimentally achieved distances of transmission exceeded the ones of only five years ago by 300%. [74]

What should be added here, though, is that the presented unconditional security does not include attacks supported by technology from not so near future. For instance, Howard E. Brandt [38] and Jeffrey H. Shapiro [283] have considered single CNOT gate attacks on single photons supporting BB84 protocol. Eve makes use of a CNOT whose target photons have polarizations in directions rotated by 22.2° with respect to H,V, that is, in between the orientations of two BB84 bases (H–V and diagonal). In that way she obtains maximal Rényi information equals to 1 for the error probability her eavesdropping creates being equal to 1/3. This significantly increases Eve's chances to snatch information and also imposes severe sacrifice of key bits on Alice and Bob during their privacy amplification.

T.A. Brun, J. Harrington, and M.M. Wilde have shown that, had Eve an access to *closed timelike curves*, she would be able to "learn the basis and bit values of each state [within a BB84 protocol] (and then prepare an identical state) without introducing any loss or disturbance in the quantum transmission." [45] However, this is so only provided the closed timelike curves really exist, but that has not been proved as of yet.

On the other hand, in 2010, it was "demonstrated experimentally that ... it [is] possible to tracelessly acquire the full secret key [with] an eavesdropping apparatus built from off-the-shelf components... The attack is surprisingly general [since a]ll commercial QKD systems and the vast majority of research systems use avalanche photodiodes-based detectors." [185]

All that is related to the fact that single photons in a BB84 protocol have definite polarizations in some basis in which they can be cloned (we can always produce arbitrary many, say, horizontally polarized photons that cannot be distinguished from the original one). It therefore seems likely that the future quantum cryptography would pay more attention to the protocols based on entangled photons (which are genuinely unpolarized) and that experimental and commercial implementations might also switch in that direction in the future.